Advances in Intelligent Systems and Computing 364

Andrzej M.J. Skulimowski
Janusz Kacprzyk *Editors*

Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions

Selected Papers from KICSS'2013—8th International Conference on Knowledge, Information, and Creativity Support Systems, November 7–9, 2013, Kraków, Poland

Springer

Advances in Intelligent Systems and Computing

Volume 364

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

About this Series

The series "Advances in Intelligent Systems and Computing" contains publications on theory, applications, and design methods of Intelligent Systems and Intelligent Computing. Virtually all disciplines such as engineering, natural sciences, computer and information science, ICT, economics, business, e-commerce, environment, healthcare, life science are covered. The list of topics spans all the areas of modern intelligent systems and computing.

The publications within "Advances in Intelligent Systems and Computing" are primarily textbooks and proceedings of important conferences, symposia and congresses. They cover significant recent developments in the field, both of a foundational and applicable character. An important characteristic feature of the series is the short publication time and world-wide distribution. This permits a rapid and broad dissemination of research results.

More information about this series at http://www.springer.com/series/11156

Andrzej M.J. Skulimowski · Janusz Kacprzyk
Editors

Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions

Selected Papers from KICSS'2013—8th International Conference on Knowledge, Information, and Creativity Support Systems, November 7–9, 2013, Kraków, Poland

Springer

Editors
Andrzej M.J. Skulimowski
Decision Sciences Laboratory, Department of Automatic Control and Biomedical Engineering, Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical Engineering
AGH University of Science and Technology
Kraków
Poland

and

International Centre for Decision Sciences and Forecasting
Kraków
Poland

Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
Warsaw
Poland

and

Faculty of Physics and Applied Computer Science
AGH University of Science and Technology
Kraków
Poland

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-3-319-19089-1 ISBN 978-3-319-19090-7 (eBook)
DOI 10.1007/978-3-319-19090-7

Library of Congress Control Number: 2015958542

Printed on acid-free paper

This Springer imprint is published by SpringerNature
The registered company is Springer International Publishing AG Switzerland

Contents

Preface and Highlights of KICSS'2013—the 8th International Conference on Knowledge, Information and Creativity Support Systems 1
Andrzej M.J. Skulimowski and Janusz Kacprzyk

Part I Knowledge

Usefulness of Inconsistency in Collaborative Knowledge Authoring in Semantic Wiki 13
Weronika T. Adrian, Grzegorz J. Nalepa and Antoni Ligęza

Modeling and Empirical Investigation on the Microscopic Social Structure and Global Group Pattern 27
Zhenpeng Li and Xijin Tang

Human-Like Knowledge Engineering, Generalization, and Creativity in Artificial Neural Associative Systems 39
Adrian Horzyk

Improvement of a Web Browser Game Through the Knowledge Extracted from Player Behavior 53
João Alves, José Neves, Sascha Lange and Martin Riedmiller

Using Extracted Symptom-Treatment Relation from Texts to Construct Problem-Solving Map 67
Chaveevan Pechsiri, Onuma Moolwat and Rapepun Piriyakul

Part II Information

An Argument-Dependent Approach to Determining the Weights of IFOWA Operator 85
Cuiping Wei and Xijin Tang

Conceptualizing an User-Centric Approach for Context-Aware Business Applications in the Future Internet Environment 99
Emilian Pascalau and Grzegorz J. Nalepa

Preprocessing Large Data Sets by the Use of Quick Sort Algorithm. . . 111
Marcin Woźniak, Zbigniew Marszałek, Marcin Gabryel and Robert K. Nowicki

ST Method-Based Algorithm for the Supply Routes for Multilocation Companies Problem 123
Lidia Dutkiewicz, Edyta Kucharska, Krzysztof Rączka and Katarzyna Grobler-Dębska

Information Adaptation by an Intelligent Knowledge-Oriented Mechanism 137
Wiesław Pietruszkiewicz and Dorota Dżega

Eval-net: Elements of Evaluation Nets as Extension to Petri Nets. 151
Michał Niedúwiecki, Krzysztof Rzecki and Krzysztof Cetnarowicz

Part III Creativity

A Japanese Problem-Solving Approach: The KJ Ho Method. 165
Susumu Kunifuji

Visual Anonymity in Online Communication: Consequences for Creativity 171
Thomas Köhler

Creativity Effects of Idea-Marathon System (IMS): Torrance Tests of Creative Thinking (TTCT) Figural Tests for College Students. 185
Takeo Higuchi, Takaya Yuizono, Kazunori Miyata, Keizo Sakurai and Takahiro Kawaji

A Color Extraction Method from Text for Use in Creating a Book Cover Image that Reflects Reader Impressions 201
Takuya Iida, Tomoko Kajiyama, Noritomo Ouchi and Isao Echizen

On the Role of Computers in Creativity-Support Systems 213
Bipin Indurkhya

Visualization of N-Gram Input Patterns for Evaluating the Level of Divergent Thinking 229
Taro Tezuka, Shun Yasumasa and Fatemeh Azadi Naghsh

Knowcations—Positioning of a Meme and Cloud-Based Personal Second Generation Knowledge Management System 243
Ulrich Schmitt

Comparing Different Implementations of the Same Fitness Criteria in an Evolutionary Algorithm for the Design of Shapes 259
Andrés Gómez de Silva Garza

Human Smart Cities: A New Vision for Redesigning Urban Community and Citizen's Life 269
Grazia Concilio, Jesse Marsh, Francesco Molinari and Francesca Rizzo

The Role of Creativity in the Development of Future Intelligent Decision Technologies 279
Andrzej M.J. Skulimowski

Part IV Creative Decisions

Solving a Multicriteria Decision Tree Problem Using Interactive Approach 301
Maciej Nowak

Strategic Planning Optimization Using Tabu Search Algorithm 315
Wojciech Chmiel, Piotr Kadłuczka, Joanna Kwiecień, Bogusław Filipowicz and Przemysław Pukocz

Need for Collective Decision When Divergent Thinking Arises in Collaborative Tasks of a Community of Practice. 329
D. Assimakopoulos, M. Tzagarakis and J. Garofalakis

Encoding Clinical Recommendations into Fuzzy DSSs: An Application to COPD Guidelines. 345
Aniello Minutolo, Massimo Esposito and Giuseppe De Pietro

Approximation of Statistical Information with Fuzzy Models for Classification in Medicine. 359
Marco Pota, Massimo Esposito and Giuseppe De Pietro

Formal Specification of Temporal Constraints in Clinical Practice Guidelines. 373
Marco Iannaccone and Massimo Esposito

Part V Feature Engineering and Classification

A Comparative Study on Single and Dual Space Reduction in Multi-label Classification 389
Eakasit Pacharawongsakda and Thanaruk Theeramunkong

Feature Selection Using Cooperative Game Theory and Relief Algorithm. 401
Shounak Gore and Venu Govindaraju

Classification with Rejection: Concepts and Evaluations 413
Wladyslaw Homenda, Marcin Luckner and Witold Pedrycz

Complementarity and Similarity of Complementary Structures in Spaces of Features and Concepts 427
Wladyslaw Homenda and Agnieszka Jastrzebska

Modeling in Feature and Concept Spaces: Exclusion Relations and Similarities of Features Related with Exclusions 441
Agnieszka Jastrzebska and Wojciech Lesinski

***E*-Unification of Feature Structures** 455
Petr Homola

A Clustering-Based Approach to Reduce Feature Redundancy 465
Renato Cordeiro de Amorim and Boris Mirkin

Part VI Music and Video

Building Internal Scene Representation in Cognitive Agents 479
Marek Jaszuk and Janusz A. Starzyk

Optical Music Recognition as the Case of Imbalanced Pattern Recognition: A Study of Single Classifiers. 493
Agnieszka Jastrzebska and Wojciech Lesinski

Modeling and Recognition of Video Events with Fuzzy Semantic Petri Nets 507
Piotr Szwed

Self-localization and Navigation in Dynamic Search Hierarchy for Video Retrieval Interface 519
Tomoko Kajiyama and Shin'ichi Satoh

On the Application of Orthogonal Series Density Estimation for Image Classification Based on Feature Description 529
Piotr Duda, Maciej Jaworski, Lena Pietruczuk, Marcin Korytkowski, Marcin Gabryel and Rafał Scherer

Fast Image Search by Trees of Keypoint Descriptors 541
Patryk Najgebauer and Rafał Scherer

Creative Intersections of Computing and Music Composition 553
Ewa Łukasik

Author Index 565

Preface and Highlights of KICSS'2013—the 8th International Conference on Knowledge, Information and Creativity Support Systems

Andrzej M.J. Skulimowski and Janusz Kacprzyk

Abstract We present, first of all, a broader perspective on the essence of problem solving and decision-making in complex real-world environments. Then, we address a special role of decision support, and decision support systems, as those solutions are generally considered the most promising for solving all kinds of nontrivial problems in our context. Finally, we analyze a vital need for tools and techniques that could involve elements of creativity in problem solving and decision-making, and systems for the support of them. We advocate a need for grasping creativity from many points of view, starting from its role to solve problems in the even more complex present world, and its role as the only means that can yield an added value and hence help attain innovativeness and competitiveness. After those general remarks we present a critical overview of the papers in this volume, peer reviewed and carefully selected, subdivided into six topical parts, and remarks on the scope and an outline of the 8th International Conference on Knowledge, Information and Creativity Support Systems (KICSS'2013) held in Kraków and Wieliczka, Poland, in November 2013, in the context of the historical development of this conference series and an increased interest of a large community of researchers and scholars, but also practitioners that have been decisive for the development of this conference series. The contents and main contributions of accepted papers are briefly outlined in the order as they appear in this volume. Some general remarks and acknowledgements are also included.

A.M.J. Skulimowski (✉)
Decision Sciences Laboratory, Department of Automatic Control and Biomedical Engineering - AGH University of Science and Technology, Kraków, Poland
e-mail: ams@agh.edu.pl

A.M.J. Skulimowski
International Centre for Decision Sciences and Forecasting—Progress and Business Foundation, Kraków, Poland

J. Kacprzyk
Systems Research Institute, Polish Academy of Sciences Warsaw, Warsaw, Poland

J. Kacprzyk
Department of Physics and Applied Computer Science AGH University of Science and Technology, Kraków, Poland

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_1

Keywords Knowledge management • Information processing • Creativity support systems • Computational creativity • Decision support systems • Classification • Artificial intelligence • Arts

This volume should be viewed from various perspectives. First, it is concerned with various issues related to broadly perceived information, knowledge, and creativity, both in their conceptual and theoretical perspective, and that focused on applications. This all is a response to a growing interest in what might be generally described as searching for conceptual frameworks, methodology, and tools and techniques, complemented finally with implementations, for real-world problem solving. This may be, for our purposes, equated with all kinds of approaches and procedures that should end up with finding the best, or maybe at least good or acceptable, decisions in complex real-world problems. Unfortunately, the present environments in which decision processes proceed are extremely complex, and this concerns all levels exemplified by individuals, groups, organizations, nations, and —finally—global alliances or even the world. The difficulty of problem solving, and its related decision-making, at all the above-mentioned levels is clearly visible as we face in the present world growing difficulties of finding good solutions in the context of both human individuals, and all kinds of groups involving multiple individuals, with their various value systems, intentions, preferences, criteria, etc.

For a long time, it has been obvious that the solution of that difficult problems people are facing all over the world, and at all levels, calls for a more concerted approach that could combine conceptual frameworks, methodologies, and tools and techniques of various fields of science, from broadly perceived applied mathematics and computer science through decision sciences, to psychology, sociology, behavioral and cognitive sciences, just to name a few. A synergistic combination of what all those areas of science can provide should open new vistas and provide powerful means for problem solving and decision-making. This is particularly true for a qualitatively new class of problems, which should be solved in the present development level of socioeconomic systems that calls for an extensive use of not information but also knowledge, and also for some novel approaches which can ultimately be viewed as examples of creativity.

It is easy to see that that new multidisciplinary paradigm, with a synergistic combination of insights and capabilities of so many fields, heavily information and knowledge based, and focused on creativity, clearly calls for a pragmatic approach that would make it possible to really solve problems involved. This boils down, as in all similar situations in the present world, to the necessity to support those problem solving and decision-making activities by some computer-based system. Such systems, the so-called decision support systems, have attracted a huge interest and have been considered a necessity in virtually all practical cases, including those pertinent to the topic of this volume.

The scientific community has been aware for a long time of the above "technological change" in problem analyses and problem solving, and their related decision-making and decision analyses, and has tried to come up with new initiatives that have involved the launching of new scientific events and gatherings, and even journals.

One initiative that started very early, about a decade ago, and met with a great interest around the world, was a series of the KICCS (which stands for Knowledge, Information and Creativity Support Systems) conferences that started with the one in Ayutthaya, Thailand (2006), and then followed by those in Ishikawa, Japan (2007), Hanoi, Vietnam (2008), Seoul, Korea (2009), Chiang Mai, Thailand (2010), Beijing, China (2011), and Melbourne, Australia (2012), and finally, in Cracow, Poland (2013). The KICSS conferences had been meant as a forum for the presentation and exchange of ideas and results in the theory and applications of what might be generally described as knowledge, decision and creativity sciences. The KICCS conferences—held annually which is not easy in the present "oversupply'' of conferences—have become popular and have attracted much interest around the world, both due to the topics covered and their scientific level that has always been a top priority of all originators and organizers.

The KICCS'2013—the 8th International Conference on Knowledge, Information and Creativity Support Systems was held in Kraków and Wieliczka, Poland in November 2013 and like all other KICCS conferences—attracted many top participants from all over the world. The choice of the focus theme, "*Looking into the Future of Creativity and Decision Support Systems,*" has clearly indicated that the growing complexity calls for some deeper and insightful discussions about the future but, obviously, complemented with an exposition of modern present developments that have proven their power and usefulness.

The conference was organized by the International Centre for Decision Sciences and Forecasting (CDSF) of the Progress and Business Foundation, Kraków, Poland (www.pbf.pl), a novel and unorthodox initiative that has tried to synergistically combine academic research activities with business, and attain through that an added value. Needless to say that the very philosophy of the CDSF does parallel the essence of problems and their solution that has motivated the launching of the KICCS conferences. The involvement of CDSF has provided an additional momentum to discussions, collaboration initiatives, etc., during the KICCS'2013.

The successful events that occurred during the KICCS'2013 were amplified by suggestions and a widely expressed interest by the participants and had reconfirmed our deep belief that the publishing of a special volume would be a good idea that would serve very well both researchers and practitioners. This had materialized herewith.

This volume contains some carefully selected papers presented at KICCS'2013, in most cases in extended versions with newer results added, representing virtually all topics covered by the conference. Following the already mentioned focus theme of KICSS'2013, that is, "*Looking into the Future of Creativity and Decision Support Systems,*" the list of conference topics included for the first time some future-oriented fields of research, such as anticipatory networks and systems, foresight support systems, relevant and newly emerging applications, exemplified by autonomous creative systems. Special attention was also given to cognitive and collaborative aspects of creativity.

In line with the very essence of the topics and scope of the KICCS'2013 conference, a special session was organized to present results of a Foresight type project "*Scenarios and Development Trends of Selected Information Society Technologies until 2025*" (SCETIST) [4] concerning the future trends and scenarios of selected artificial intelligence and information society technologies until 2025. The importance of this project is difficult to overestimate as it has tried to determine main trends and development patterns of broadly perceived IT/ICT for the next two decades, as well as the road to information society. This all has a decisive importance for the innovativeness and competitiveness of the country, and also the European Union. Interestingly enough, the presentation of results of the SCETIST project and discussions were held in one of top sightseeing attractions, in the 1,000 year old Wieliczka Salt Mine, an UNESCO World Heritage Treasure, more than 100 m underground. An unusual and attractive venue has greatly contributed to an active participation. A photograph of the participants is shown in Fig. 1.

In case of all previous conferences, their proceedings were always published by respected publishing houses, exemplified by Springer and IEEE Press, and were often followed by special post-proceedings including selected papers of a particular quality and interest (cf. [2, 3, 6]). In the case of KICCS'2013, the proceedings were published by the Progress and Business Publishers [5] which sped up the process. Each paper submitted to the conference was peer reviewed by experts in the respective fields, usually two or three experts for short and demo papers, and three or four experts for regular papers. The submissions came from a total of 20 countries from four continents. The truly global nature of the conference was also evident in the composition of the International Program Committee, which comprised 45 high-rank experts from 18 countries. From over 100 submitted papers, 60 papers were accepted for presentation. The full papers were 40, while 20 were short or poster papers. Out of them, by taking into account quality and interest of a broader community, 39 papers, considerably revised and extended, were selected

Fig. 1 A KICSS'2013 session in the Wieliczka Salt Mine

for inclusion into this volume after a peer review process. They present original, previously unpublished research results on theoretical foundations, implementations of decision support and expert systems, creativity support systems, as well as case studies of successful applications of these ideas in various fields.

In addition to the regularly accepted contributions, this volume includes two invited papers by prominent and opinion-making researchers and scholars. The first one is by Prof. Susumu Kunifuji, Vice President of the Japan Institute of Advanced Science and Technology (JAIST), Ishikawa, Japan, and the second by Prof. Thomas Köhler, Director of the Media Center at the Technical University of Dresden, Germany. Specifically, Prof. Kunifuji delivered a speech on a famous Japanese problem-solving approach: the so-called KJ Ho method, which is in Japan the most popular creative problem-solving methodology using creative thinking. It puts unstructured information on a subject matter of interest into order through alternating divergent and convergent thinking steps.

Professor Köhler presented selected issues related to the visual anonymity in online communication and its consequences for creativity. Based upon empirical evidence, he showed how the so-called social self develops creatively in online communication, which in a more general setting is also referred to as computer-mediated communication (Fig. 2).

Following the tradition of previous KICCS conferences, this post-proceedings volume covers some more general issues, here the most relevant aspects of knowledge management, knowledge engineering, decision support systems, intelligent information systems, and creativity in an information technology context.

The papers in this volume are organized in six topical parts, devoted to the main themes of the conference:

Fig. 2 A group of KICSS'2013 participants at the Market Square in Kraków, November 2013

- Knowledge (5 papers),
- Information (6 papers),
- Creativity (10 papers, including 2 invited lectures),
- Creative Decisions (6 papers),
- Classification and Feature Engineering (7 papers), and
- Music and Video (7 papers).

Part I starts with the paper by Adrian, Nalepa and Ligęza who provide a classification of inconsistency in semantic wiki systems in terms of its origin, level, type, and significance and point out a possibility to derive useful information out of such inconsistencies. Li and Tang investigate the microscopic social mechanisms through agent-based Hopfield models to explain the emergence of macroscopic polarization. Horzyk provides the description of artificial neural associative systems that are capable of creating knowledge in a human-like associative way and of triggering artificial associations. Alves et al. describe how a web browser game can be improved through the knowledge obtained from the analysis of the behavior of its players. The last paper by Pechsiri, Moolwat, and Piriyakul studies the methods to extract relations between symptoms and treatments in order to construct a medical problem-solving map.

Part II is devoted primarily to analytic approaches to deal with information in a creative problem-solving context. In the first paper of this part, Wei and Tang define an intuitionistic fuzzy dependent ordered weighted averaging operator (IFDOWA) and apply it to solve multiattribute group decision-making problems. A user-centric approach for context-aware business applications in the future Internet is conceptualized in the paper by Pascalau and Nalepa. Woźniak et al. test two versions of the quick sort algorithm for large data sets and determined their scope of applicability. Dutkiewicz et al. present an algorithm for the optimization of supply routes for multi-location companies problem. This algorithm is based on the substitution tasks method. Pietruszkiewicz and Dżega propose an intelligent auto-adaptive information delivery mechanism to be applied primarily in the distributed e-learning environments. Niedźwiecki, Rzecki, and Cetnarowicz discuss the properties evaluation nets from the point of view of embedding their most useful elements into the Petri Nets.

In Part III, focused on creativity, Higuchi et al. present positive creativity effects of the Idea-Marathon System (IMS) based on the Torrance tests performed for the students of the Ohsuki City College. Iida et al. propose a method to extract colors out of the book contents to automatically create book cover images that can adequately reflect readers' impressions. Indurkhya reviews the design and role of computer-aided creativity support systems in four different application areas. Tezuka, Yasumasa, and Azadi Naghsh propose a new creativity test based on measuring the occurrence of the same subsequences of numbers in a longer sequence input by the subject. Schmitt in his paper on cloud-based personal knowledge management system discusses the ability to create new knowledge and innovation by such systems. Gómez de Silva Garza investigates properties of an evolutionary algorithm applied to shape design. Concilio et al. discuss visions for the redesign of urban communities in smart cities of the future taking into account different quality of life aspects. In the last paper of this part Skulimowski presents the most relevant results of the foresight SCETIST project [4] related to some selected artificial intelligence technologies. The specific topics under investigation included, for instance, the role of creativity in the development of decision and neurocognitive systems.

In the two invited papers, Kunifuji discusses the famous Japanese problem-solving approach, the so-called KJ Ho method, which is in Japan the most popular creative problem-solving methodology using creative thinking, while Köhler presents issues related to the visual anonymity in online communication and its consequences for creativity.

Part IV contains six papers devoted to creative decision support and decision-making. Nowak presents theoretical foundations of an interactive method to solving a multicriteria decision tree problem. Chmiel et al. propose a novel application of a tabu search algorithm to optimize a roadmapping-based strategic planning process. Assimakopoulos, Tzagarakis, and Garofalakis propose an approach to deal with inconsistency of rankings and divergent thinking when a consensus decision is needed in a collaborative expert activity. The next three papers deal with the methodology and applications of medical decision support systems. Minutolo, Esposito, and De Pietro present a fuzzy knowledge management system and its application to codify clinical guidelines in terms of fuzzy if-then rules in the diagnostics and treatment of the chronic obstructive pulmonary disease. Pota, Esposito, and De Pietro discuss the use of statistical versus fuzzy models in medical diagnostics. Iannaccone and Esposito investigate a formal language to specify temporal constraints in a guideline model for clinical decision support.

Part V is concerned with various issues related, on the one hand, to an emerging area that may be called feature engineering and vital to the success of virtually all machine learning problem. Basically, it boils down to a proper choice and design of feature sets which include many factors, notably the understanding of properties of the task and their interactions, and then experiments. On the other hand, classification is dealt with. In the first paper, Pacharawongsakda and Theeramunkong discuss various dimensionality reduction methods that can help attain a higher effectiveness and efficiency of multi-label classification. Basically, they discuss two classes of such methods: based on single space reduction and dual space reduction, and show that the latter yields a better performance. Gore and Govindaraju propose a new approach to feature selection based on the elements of cooperative game theory and the relief algorithm, and show that the results—tested on some NIPS 2003 and UC Irvine data sets—are good, comparable to other state-of-the-art approaches. Homenda, Luckner and Pedrycz discuss issues related to various ideas and approaches to classification with rejection. The idea is that so-called foreign elements are rejected, i.e., not assigned them to any of given classes. Clearly, this should be done with care. The authors discuss various evaluation methods, and show that the models proposed can be used in particular as classifiers for noised data in which the recognized input is not limited to elements of known classes. Then, Homenda and Jastrzębska are concerned with some elements of analyses of relations as models of complexity in real-world phenomena and systems, notably the problem of complementarity. The purpose is to be able to describe dependencies in a deeper way. The authors propose a new definition of complementarity for concepts described by features, and propose similarity relations for complementarity structures in the spaces of features and concepts. Jastrzębska and Lesiński further explore issues related to the modeling of feature and concept spaces by introducing new definitions of three types of exclusion relations: weak, strict, and multiple qualitative are defined and discussed. They focus on similarity analyses. In the next paper, Homola presents an application of e-unification of feature structures to natural language processing (NLP). E-unification is an operation that assigns to a pair of elements of a commutative idempotent monoid M a dominating pair in terms of an equational theory over M. This formalism is then used to describe the automatic translation process. Cordeiro de Amorim and Mirkin propose a new feature weighting clustering algorithm. Basically, their paper introduces an unsupervised feature selection method for data preprocessing step to reduce the number of redundant features, and then to cluster similar features together and finally select a subset of representative features for each cluster. A maximum information compression index between each feature and its respective cluster centroid is used. Examples include three EEG data sets.

Part VI is mainly concerned with some multimedia-related problem, to be more specific, with music and video. Jaszuk and Starzyk discuss some vital problems in robot navigation in real-world environments, specifically related to a continuous observation of a robot's surroundings, and creating an internal representation of a perceived scene. These tasks involve a sequence of cognitive processes exemplified by attention focus, recognition of objects, and building internal scene representation. The authors discuss some relevant elements of a cognitive system which implement mechanisms of scene observation based on visual saccades, followed by creating the scene representation. Using such an internal representation, scene comparison, based on a distance matrix, is performed making it possible to recognize known places, make changes in the environment, or in the structure of complex objects. Jastrzębska and Lesiński consider optical music recognition as some special case of pattern recognition with imbalanced classes. They present a comparative study of classification results of musical symbols using: the k-NNs, k-means, Mahalanobis minimal distance, and decision tree algorithms. The results obtained are promising. Szwed is concerned with the modeling and automatic recognition of video events using, for the event specification, linear temporal logic, and then, for the recognition, the fuzzy semantic Petri nets, that is, some Petri nets with an underlying fuzzy ontology. A prototype detection system consisting of an FSPN interpreter, the fuzzy ontology and a set of predicate evaluators is presented. Initial tests yielding promising results are reported. Kajiyama and Satoh deal with some problems in dynamic search for video retrieval related to self-localization and navigation. They propose an improved graphical search interface, 'Revolving Cube Show,' for multi-faceted metadata, which can handle discrete, continuous, and hierarchical attributes, enabling the users to search flexibly and intuitively by using simple operations to combine attributes. Two functions, for displaying a search hierarchy as a guide tree and for moving to a specific position in the search hierarchy, are added. A video retrieval application for the iPad is shown and tested on 10,352 Japanese TV programs. Duda et al. present a new image classification algorithm called a density-based classifier which puts together the image representation based on keypoints and the estimation of the probability density of descriptors with the application of orthonormal series. Experiments are shown that confirm that the results obtained are good. Najgebauer and Scherer present a method for fast image searching tree-based representation of local interest point descriptors. Due to a decreased number of steps needed, such a representation of image keypoints is more efficient than the standard, frequently used list representation where images are compared in all-to-all manner. The proposed method generates a tree structure from a set of image descriptors, and since the average values of the descriptor components in the nodes are stored, it is possible to quickly compare sets of descriptors by traversing the tree from the root to a leaf. It is also possible to compare whole trees of descriptors what can speed up the process of images comparison. Experimental results confirm advantages of the method. Finally, Łukasik presents various aspects of music composing supported by computer algorithms and methods. Basically, amateur pop composers search databases for music similar to their composition to get stylish parts and aspects their music are stylish or to safeguard themselves against plagiarism. Others are interested in surprising effects of automated composing and put some demands on the produced music, for instance, to attain some emotion or to change musical parameters of the composition, preferably in real time, by *live coding*. Reflections on the creativity in the domain of computer-supported music composition are illustrated by some realistic projects.

As it can easily be noticed, the topics covered, and the scope of their coverage, can serve as a very good source of novel trends, and tools and techniques, that can help many people find solutions to their problems which can generally be stated as a quest to develop means for the representation, analysis and implementation of problems and systems for real-world problem solving. What is relevant in this context is that due to an ever increasing complexity of the world, and a need for innovation and competitive at the level of even individuals and small groups, but mainly branches, nations or even more global alliances, the problem solving needs a creative approach which can only provide an added value needed. We think that the KICCS'2013 conference has been a key event in that quest. Luckily enough, its success as well as the quality of the selected papers included in this volume and also opinion of all the participants has confirmed our belief.

Acknowledgments Sincere thanks are due to the members of the KICSS'2013 local Organizing Committee, the editorial staff, in particular to Ms. Alicja Madura, Dr. Witold Majdak, Ms. Inez Badecka, to the Program Committee members and external reviewers, and to all those who helped with editing this volume. Special thanks and appreciation are also due to Prof. Susumu Kunifuji, the Vice President of JAIST, and the initiator of the KICSS Conference Series. The financial support of the Polish Ministry of Science and Higher Education, under Grant no. 594/P-DUNdem/2013, as well as the contribution from the Mayor of the Wieliczka Municipality are greatly appreciated. Finally, we would like to express our thanks to all the invited speakers,

contributing authors and participants of KICSS'2013 who have played a key role to attain the success and the scientific quality of this Conference.

References

1. Kunifuji, S.: Proceedings of The Third International Conference on Knowledge, Information and Creativity Support Systems. Hanoi National University of Education, Hanoi, Vietnam, 22–23 Dec 2008
2. Kunifuji, S., Tang, X.J., Theeramunkong, T. (eds.): Proceeding of the 6th International Conference on Knowledge, Information and Creativity Support Systems (KICSS 2011) Beijing, China. JAIST Press, Beijing, 22–24 Oct 2011
3. Lee, V.C.S., Ong, K.L. (eds.): KICSS 2012: Seventh International Conference on Knowledge, Information and Creativity Support Systems. IEEE Computer Society, CPS Conference Publishing Services, Melbourne, Victoria, Australia 8–10 Nov 2012
4. Skulimowski, A.M.J. (ed.): "Scenarios and Development Trends of Selected Information Society Technologies until 2025" (SCETIST)—Final Report. Progress & Business Publishers, Kraków, Progress & Business Foundation. http://www.ict.foresight.pl (2013)
5. Skulimowski, A.M.J. (ed.): "Looking into the future of creativity and decision support systems. In: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems (KICSS'2013). Progress & Business Publishers, Kraków (2013) www.kicss2013.ipbf.eu
6. Theeramunkong, T., Kunifuji, S., Nattee, C., Sornlertlamvanich, V. (eds.): Knowledge, Information, and Creativity Support Systems, KICSS 2010 Revised Selected Papers, LNAI, vol. 6746. Springer, Berlin (2011)

Part I
Knowledge

Usefulness of Inconsistency in Collaborative Knowledge Authoring in Semantic Wiki

Weronika T. Adrian, Grzegorz J. Nalepa and Antoni Ligęza

Abstract Inconsistency in knowledge bases traditionally was considered undesired. Systematic eradication of it served to ensure high quality of a system. However, in case of semantic wikis, where distributed, hybrid knowledge bases are developed and maintained collectively, and inconsistency appears to be an intrinsic phenomena. In this paper, we analyze inconsistency in a semantic wiki system in terms of its origin, level, type, and significance. We claim that in some cases inconsistency should be tolerated and show examples where it can be used in a constructive way.

Keywords Semantic wikis · Knowledge engineering · Inconsistency

1 Introduction

Traditional approach to inconsistency in knowledge-based systems considered it an anomaly [25]. One of the main reasons for that is the "principle of explosion" (ECQ, from Latin: *Ex contradictione sequitur quodlibet* which means "from a contradiction, anything follows"). If anything can be entailed from a set of inconsistent statements, then the inconsistent knowledge base becomes unusable. Therefore, numerous methods and techniques have been developed to suppress inconsistency, either by rejecting contradictions (removing, forgetting, etc.) or by searching for a consensus to restore consistency.

With the advent of modern Web-based technologies, there is a growing intensity of collaboration on the Web. Wikipedia-like portals, recommendation systems, or community websites are examples of modern knowledge bases. New challenges are

W.T. Adrian (✉) · G.J. Nalepa · A. Ligęza
AGH University of Science and Technology, al. Mickiewicza 30, 30-059 Krakow, Poland
e-mail: wta@agh.edu.pl

G.J. Nalepa
e-mail: gjn@agh.edu.pl

A. Ligęza
e-mail: ligeza@agh.edu.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_2

posed by distributed knowledge authoring, increased use of mobile devices, dynamic changes of the knowledge, and use of hybrid knowledge representation with mixed levels of formality [6]. Quality of knowledge is often evaluated collectively, by discussion, negotiations, and voting.

It is impractical to treat such knowledge bases same as centralized homogeneous systems, where consistency was an important quality factor. We claim that in semantic wiki systems, struggling for consistency can be ineffective, and can suppress such desirable phenomena as collaborative synergy and fast development of knowledge. Thus, inconsistency should be accepted and incorporated into reasoning rather than removed or ignored. This paper is an enhanced version of the paper presented at the KICSS2013 conference [3].

The paper is organized as follows: In Sect. 2, we introduce the motivation for our work. Then we review selected approaches to handle inconsistency, taking into consideration various aspects and levels of it in Sect. 3. A conceptualization of our semantic wiki environment is proposed in Sect. 4, followed by an analysis of inconsistency in it in terms of origin, types, and significance. This constitutes a starting point for a discussion on tolerating inconsistency in Sect. 5 and presentation of an exemplary use case in Sect. 6. Conclusion and future work is outlined in Sect. 7.

2 Motivation

Verification and validation of knowledge-based systems [13] is a mature field in which numerous solutions, both classic and recent [28], have been proposed. Verification challenges and algorithms depend on selected knowledge representation and reasoning within the considered system. XTT2 [36] is a logic-based representation for rule-based systems, which allows to visually model, refine, and execute modularized rule bases. Moreover, a formal analysis and verification is possible [32]. Such a representation could be adapted for modeling distributed knowledge bases (e.g., with use of semantic wikis [1, 31]).

Specific issues related to collective knowledge engineering [34] as well as verification [5] in such environments have been investigated. It appears that to some extend it would be beneficial to use some of the existing verification solutions. However, inconsistency in collaborative settings appears to be intrinsic phenomena, so the verification methods should be adapted accordingly. When inconsistency arises, the system should automatically and appropriately react, i.e., recognize if it is undesirable, or if it may be accepted and used constructively. Incorporating inconsistency into reasoning and taking advantage of it is not a new idea. In [15], several practical use cases were given and a general framework for including inconsistency was presented, and in [39], usefulness of inconsistency in software development has been discussed.

In order to develop reasonable methods of verification and inconsistency handling for knowledge engineering in semantic wikis, we aim to analyze various aspects, levels, and types of inconsistency. In the following section, we ground our discussion in overview of existing approaches to inconsistency in general, and then proceed to inconsistency analysis in semantic wiki system.

3 Preliminaries

3.1 Basic Concepts

One can find several interpretations of inconsistency that are reflected in various definitions. Intuitively, inconsistency appears when a set of sentences (formulas, theorems, beliefs) cannot be true at the same time. More precisely,

A formal system is inconsistent if there is a formula ϕ such that both ϕ and $\neg\phi$ are theorems [17]. Alternatively,

Axiomatic system is inconsistent if for a given set of axioms Γ (relative to a given logical language $\mathcal{L}$) a formula ϕ can be entailed ($\Gamma \models \phi$) and similarly $\neg\phi$ can be entailed ($\Gamma \models \neg\phi$) [23]. Finally,

A contradiction between two statements is a strong kind of inconsistency between them, such that one must be true and the other must be false.[1]

If model-theoretic semantics is concerned, then the knowledge base or its corresponding theory is inconsistent if it does not have a model [12]. There are also less formal terms in use, not necessarily equivalent to logical inconsistency, for instance incoherent data or incompatible conceptualizations [23].

3.2 Formal Representation of Inconsistency

In order to formalize inconsistency handling, there must be a formal representation of inconsistency itself. In [11], where logics are defined as "formal systems consisting of a language L (in the form of a set of formulas) on which an inference operation C is defined," three approaches are distinguished:

1. *A-scheme*: to pick a subset of the language, and use each element of the subset as a representation of absurdity,
2. *C-scheme*: to relate contradictions to inference, stating that inconsistency arises when all formulas are inferred,
3. *N-scheme*: to capture contradictions through an auxiliary notion of negation ($A, \neg A$ or $A \wedge \neg A$ if conjunction is available).

[1] See http://www.csus.edu/indiv/d/dowdenb/misc/inconsistency.htm.

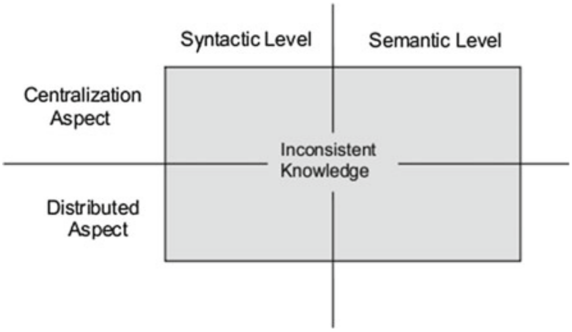


Fig. 1 Aspects and levels of inconsistency [38]

Inconsistency may be also represented in a form of *conflict profiles* as explained in [38]. Finally, inconsistency may be incorporated into formal logic, for instance as in multi-valued logics [9] in which one can represent a statement that is both inferred to be true and false.

3.3 Aspects and Levels of Inconsistency

One can consider different aspects and levels of inconsistency. In [38], two-dimensional classification of inconsistency is given as (1) syntactic vs. semantic level of inconsistency, and (2) centralization vs. distributed aspect of it (see Fig. 1). Inconsistency may be considered in *distribution aspect* where the basic cause of inconsistency is the independence of knowledge agents or knowledge processing mechanisms, or in *centralization aspect* where inconsistency is caused by the dynamic change of the world. It can also be identified and processed on a *syntactic* and *semantic* level. Alternatively, a distinction given in [26] states that inconsistency can be checked for in a purely *logical* way (e.g., p and $\neg p$ are present in the knowledge under discourse), or as *material* inconsistency, when two pieces of knowledge are invalid together due to the assumed interpretation.

3.4 Measuring Inconsistency

Binary distinction between consistent and inconsistent knowledge base is often insufficient. In order to better understand inconsistency and apply appropriate technique to handle it, one should recognize degree of inconsistency measured in some dimensions. Several methods, models, and metrics have been proposed to measure inconsistency [18, 21, 30]. Classifications for inconsistent theories have been proposed in [17]. In [22], different dimensions of measuring inconsistency are explained, selected approaches are compared, and their applications in cases as negotiation between agents or comparing heterogeneous sources are presented. Finally, inconsistency measures can be used to decide, if resolving inconsistency is worthwhile, taking into consideration associated loss of information [19].

3.5 Selected Approaches to Inconsistency

Actual and potential contradictions In [11], two approaches are distinguished, which lay the ground for various methods of inconsistency handling. *Actual contradictions* view assumes that contradictions appear naturally and thus a representation of them within a formalism and reasoning mechanisms in the presence of them should be provided. In *potential contradictions* view, it is claimed that in reality contradictions do not appear, and when contradicted information is given, there are some statements 'responsible' for it. Reasoning then conforms to trying to identify the 'strongest argument,' where all arguments that oppose it are discarded. This approach assumes some mechanism to resolve conflicts and obviate the potential of any contradiction. Practical implementations of this view are, e.g., defeasible reasoning, or modal formulation of default logic.

Methods for Handling Inconsistency Following different aspects and levels of inconsistency, numerous methods for handling it have been developed (survey of them is beyond the scope of this paper, for reference see e.g., [10, 11, 38]). Ordering them from most restrictive to most tolerating inconsistency, one can

1. Discard inconsistency, e.g., by removing/forgetting inconsistent statements.
2. Suppress it, by selecting consistent subsets (removing union of the minimally inconsistent subsets or taking intersection of the maximally consistent subsets, or removing smallest number of assumptions), isolating or repairing.
3. Do not accept it but use to find a consensus, or in argumentation frameworks.
4. Tolerate or accept it, by representing and reasoning with it (paraconsistency).
5. Accept and use constructively, e.g., learn from inconsistency [23].

Amending Classical Logic for Handling Inconsistency Because classical logic collapses in the presence of inconsistency, several amendments have been proposed [11]:

1. Syntactic-based approach: fragment of classical proof theory that is sufficiently weakened so that an arbitrary B fails to be inferred from $A \wedge \neg A$ (numerous paraconsistent logics [41]).
2. Semantic-based approach: conjoining the truth values *false* and *true* can be meaningful and results in a new truth values, namely *contradictory* (three-valued logic, Belnap's four-valued logic [9]).
3. Restricting the use of certain classical proof rules so as to avoid the proof of an arbitrary B following from $A \wedge \neg A$.
4. If premises form contradictory statements, then focusing on consistent parts of this collection and reasoning only from such consistent parts (subsets).

Paraconsistent Logics Main assumption of paraconsistent logics is the rejection of inference explosion (ECQ) in the presence of inconsistency. This shared assumption may be realized by different means. Selected paraconsistent logics include discussive logic, non-adjunctive systems, preservation, adaptive logics, logics of formal inconsistency, many-valued logics, or relevant logics (for a discussion and further reference see [41]). Paraconsistent logic has significant overlap with many-valued logic; however, not all paraconsistent logics are many-valued (and not all many-valued logics are paraconsistent). Dialetheic logics, which are also many-valued, are paraconsistent, but the converse does not hold. Intuitionistic logic allows $A \vee \neg A$ not to be equivalent to true, while paraconsistent logic allows $A \wedge \neg A$ not to be equivalent to false. Thus paraconsistent logic can be regarded as the "dual" of intuitionistic logic.

3.6 Selected Application Areas

Inconsistency has been widely studied in the areas of software engineering [39], legal knowledge engineering, verification of knowledge-based systems [25], diagnostics [27], and artificial intelligence (i.e., robotics, knowledge representation, and reasoning). With the advent of the Semantic Web, numerous proposals have been discussed for semantic knowledge management, both on the levels of semantic annotations in RDF/S [4] as well as ontologies [16, 20, 29, 38] (for a discussion about inconsistency handling in Semantic Web environments see [2]). Recently, inconsistency management has also been studied in multi-context systems [14] and traffic regulations within smart cities [8].

4 Interpretation of Inconsistency in Semantic Wikis

Based on practical experience and previous work [1, 34], we introduce the notion of *Collaborative Knowledge Authoring* in semantic wiki systems. It is a process of eliciting, structuring, formalizing, and operationalizing knowledge performed by conscious agents that collaborate on a knowledge level toward achieving a common goal, can communicate and help each other, as well as change their opinions. One can see that we highlight the following characteristics:

A Collaborative:

- multiple *conscious* agents (authors, knowledge engineers),
- communication, help and dynamics of opinions among agents.

Knowledge:

- consideration on a knowledge level [37], and consequently various methods of knowledge representation and reasoning (hybrid representations, different levels of formality [6])

Authoring:

- set of tasks (knowledge elicitation, structuring, formalization, and operationalization)
- common goal: development of a useful artifact

The developed knowledge base should be *valuable* for its users. This user-centric perspective intuitively alters the quality criteria of the system. Specifically, one should be able to *meaningfully* answer queries posed to the system. This does not necessarily require sustaining consistency.

Inconsistency in such a semantic wiki environment may be considered as driven by the following three factors: distributed knowledge authoring, dynamic change of the system, and hybrid knowledge representation. Analyzing the origins of inconsistency, one can distinguish the following categories:

1. Distributed knowledge authoring
   (a) independent knowledge sources/authors
   (b) independent knowledge processing
   (c) unconscious disagreement
      - in/competence of experts
      - inaccuracy of statements
   (d) conscious disagreement
      - different opinions
      - different conceptualizations
2. Dynamic change of the system
   (a) revisions of knowledge bases
   (b) assertion of facts inconsistent with existing KB
   (c) assertion of rules making the KB inconsistent
   (d) dynamic changes of the world (making the KB outdated)
3. Knowledge representation
   (a) incompatibilities between models expressed with different KR
      - e.g., incorrect combination of disjoint and derives relations
   (b) improper conceptualizations
      - polysemy (missing disambiguation of different word senses)
      - overgeneralized concepts

Moreover, we distinguish the following types of inconsistency:

- Syntactic inconsistency within a model in chosen representation.

- Semantic inconsistency between knowledge represented in different formalisms, possibly of different levels of formality [6, 7].
- Material inconsistency: if a behavior of the system is modeled within the KBS and the model is inconsistent with the actual execution of it.

Impact of the inconsistency depends on a formalization level of the knowledge base (the more the formal representation, the more significant is the inconsistency).

5 On Possible Usefulness of Inconsistency

Inconsistency can be acceptable, or even desirable in a system, as long as it has appropriate mechanisms for acting on it [15]. Sometimes, systematic eradication of contradictions is a mistake on several grounds [11]. First one is pragmatic: they may be hard to detect. Second, they can be more informative than any consistent revision of the theory (for a discussion see [15]).

While inconsistencies are undesired in such situations as specification of a plan or sensor fusion in robotics, they proved to be useful in, e.g., law (inconsistencies in opposition case), income tax database, where contradictions point to fraud investigation (contradictory information should be then kept and reasoned with), or in preliminary stage of software engineering (requirements capture stage), where premature resolution can force an arbitrary decision to be made without the choice being properly considered.

Inconsistency may be a useful trigger for logical actions, e.g., in directing reasoning, instigating the natural processes of argumentation, information seeking, interaction, knowledge acquisition and refinement, adaptation, and learning [23].

Tolerating inconsistency in a semantic wiki system is *pragmatic*, because inconsistency is hard to detect if hybrid representation is considered. Moreover, consistency is *not necessary for the knowledge base to be valuable*, if the quality is ensured collaboratively by employing social mechanisms such as voting, discussions, or "likes." During divergent thinking collaborative tasks, such as brainstorming sessions in research collaboration [10] or early prototyping while developing innovative ideas [40], inconsistency may inspire new associations and lead to more interesting solutions. Furthermore, inconsistency of opinions is thought-provoking and valuable in some applications (e.g., recommendation systems). Finally, inconsistency may be helpful in discovering potential areas of improvement, and be a trigger to acquire more knowledge.

6 Exemplary Use Case

A recommendation system has been implemented in Loki [33–35], a semantic wiki that supports semantic annotations and rules. Knowledge base about movies has been developed collaboratively by editing semantic wiki pages enhanced with semantic annotations (see Fig. 2).

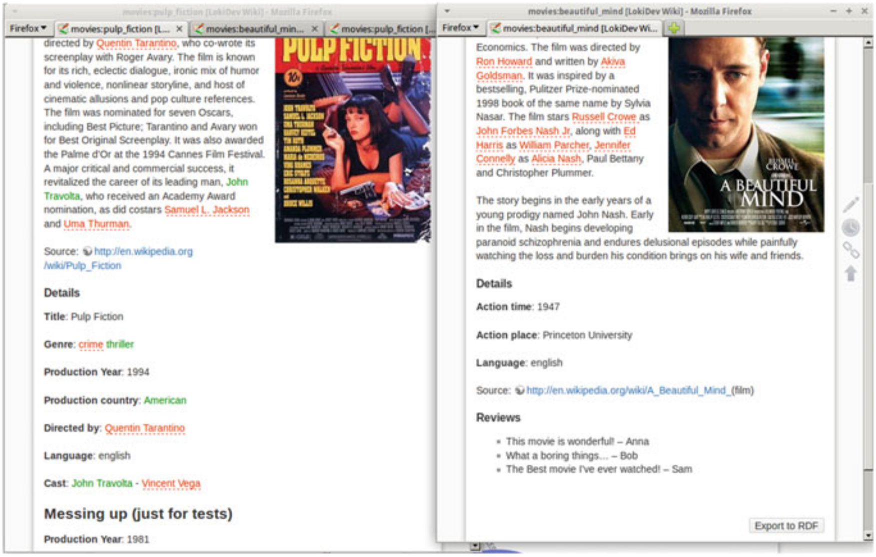


Fig. 2 Loki semantic wiki system: movies use case

Semantic annotations are mapped to the underlying logical representation that allows to reason automatically. Information may be obtained from wiki by posing semantic queries in *ask* and *SPARQL* languages, as well as Prolog goals. The system can also give recommendations based on rules defined for a user [1]. Information may be exported to RDF/XML.

Inconsistency arises when multiple users edit wiki pages and give contradictory information about the same objects. In order to easily locate and asses inconsistency, a visualization plugin has been developed.[2] The plugin highlights inconsistencies for whole namespaces (see Fig. 3) and single pages (see Fig. 4).

Whether a contradiction is acceptable or not depends on the ontology to which the system conforms. For instance, there are *functional* relations and attributes, such as `title` of a movie or its `production year`. For these properties, two different values constitute undesired inconsistency. On the other hand, for properties such as `rating`, several different values may be given and in this case, the inconsistency is natural and represents various opinions of the users. Thus Loki visualization plugin provides a simple configuration mechanism to define what properties are appropriate for given classes of objects and which of them are functional (see Fig. 4). Consequently, not only inconsistencies can be easily identified, but they also may be assessed and treated appropriately.

[2] A prototype implementation of the plugin was carried out by master students Magdalena Chmielewska and Tomasz Szczęśniak.

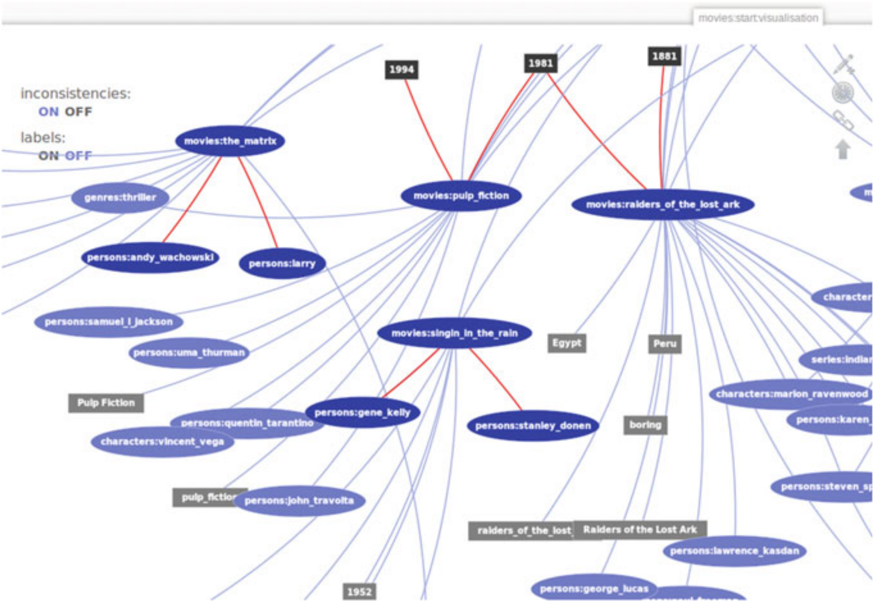


Fig. 3 Loki inconsistency visualization plugin: movies namespace

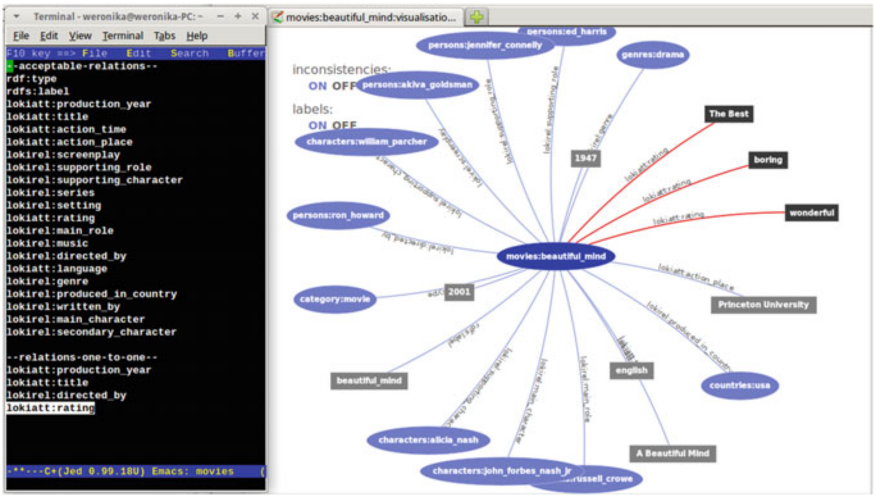


Fig. 4 Identified inconsistency highlighted for analysis

Recognizing more types of inconsistency is planned to be implemented in future. Further refinement of the system may be considered by employing voting or consensus methods to determine appropriate version of information.

7 Summary and Outlook

Inconsistency is an inherent phenomenon in collaborative knowledge authoring. We claim that only after a thorough analysis, one can take a reasonable decision how to handle identified inconsistency. Particularly, it is not obvious that inconsistency must be considered unacceptable. It may be useful and in this paper, we showed an exemplary use case. For future work, we will analyze how to adapt the following techniques for our semantic wiki environment: consensus methods [38], paraconsistent logics [41], 4-valued logics [9, 29], argumentation frameworks [16], belief revision, updating knowledge (e.g., ontologies) [24], inconsistent subsets, and semantic relevance metrics [20]. In fact our environment may be extended by a set of plugins simplifying the collaborative knowledge authoring that considers inconsistency as a useful phenomenon.

References

1. Adrian, W.T., Bobek, S., Nalepa, G.J., Kaczor, K., Kluza, K.: How to reason by HeaRT in a semantic knowledge-based wiki. In: Proceedings of the 23rd IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2011, pp. 438–441. Boca Raton, Florida, USA, Nov 2011. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=6103361&tag=1
2. Adrian, W.T., Ligęza, A., Nalepa, G.J.: Inconsistency handling in collaborative knowledge management. In: Ganzha, M., Maciaszek, L.A., Paprzycki, M. (eds.) Proceedings of the Federated Conference on Computer Science and Information Systems—FedCSIS 2013, pp. 1261–1266. IEEE, Krakow, Poland, 8–11 Sept 2013
3. Adrian, W.T., Nalepa, G.J., Ligęza, A.: On potential usefulness of inconsistency in collaborative knowledge engineering. In: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems (2013)
4. Analyti, A., Antoniou, G., Damásio, C.V., Wagner, G.: Extended rdf as a semantic foundation of rule markup languages. J. Artif. Intell. Res. **32**(1), 37–94 (2008)
5. Baumeister, J., Nalepa, G.J.: Verification of distributed knowledge in semantic knowledge wikis. In: Lane, H.C., Guesgen, H.W. (eds.) FLAIRS-22: Proceedings of the Twenty-Second International Florida Artificial Intelligence Research Society conference: 19–21 May 2009, Sanibel Island, Florida, USA, pp. 384–389. FLAIRS, AAAI Press, Menlo Park, California (2009), to be published
6. Baumeister, J., Reutelshoefer, J., Puppe, F.: Engineering intelligent systems on the knowledge formalization continuum. Int. J. Appl. Math. Comput. Sci. (AMCS) **21**(1) (2011). http://ki.informatik.uni-wuerzburg.de/papers/baumeister/2011/2011-Baumeister-KFC-AMCS.pdf
7. Baumeister, J., Seipel, D.: Anomalies in ontologies with rules. Web Semant. Sci. Serv. Agents World Wide Web **8**(1), 55–68 (2010). http://www.sciencedirect.com/science/article/pii/S1570826809000778
8. Beck, H., Eiter, T., Krennwallner, T.: Inconsistency management for traffic regulations: formalization and complexity results. In: del Cerro, L.F., Herzig, A., Mengin, J. (eds.) Logics in Artificial Intelligence. Lecture Notes in Computer Science, vol. 7519, pp. 80–93. Springer, Heidelberg (2012). http://dx.doi.org/10.1007/978-3-642-33353-8_7
9. Belnap Jr, N.D.: A useful four-valued logic. In: Modern Uses of Multiple-valued Logic, pp. 5–37. Springer (1977)
10. Bertossi, L., Hunter, A., Schaub, T.: Introduction to inconsistency tolerance. In: Bertossi, L., Hunter, A., Schaub, T. (eds.) Inconsistency Tolerance. Lecture Notes in Computer Sci-

ence, vol. 3300, pp. 1–14. Springer, Heidelberg (2005). http://dx.doi.org/10.1007/978-3-540-30597-2_1

11. Besnard, P., Hunter, A.: Introduction to actual and potentialcontradictions.In: Besnard, P., Hunter, A. (eds.) Reasoning with Actual and PotentialContradictions, Handbook of Defeasible Reasoning and Uncertainty ManagementSystems, vol. 2, pp. 1–9. Springer Netherlands (1998). http://dx.doi.org/10.1007/978-94-017-1739-7_1
12. Bienvenu, M., Rosati, R.: New inconsistency-tolerant semantics for robust ontology-based data access. In: Description Logics, pp. 53–64 (2013)
13. Coenen, F., Bench-Capon, T., Boswell, R., Dibie-Barthélemy, J., Eaglestone, B., Gerrits, R., Grégoire, E., Ligęza, A., Laita, L., Owoc, M., Sellini, F., Spreeuwenberg, S., Vanthienen, J., Vermesan, A., Wiratunga, N.: Validation and verification of knowledge-based systems: report on eurovav99. Knowl. Eng. Rev. **15**(2), 187–196, June 2000. http://dx.doi.org/10.1017/S0269888900002010
14. Eiter, T., Fink, M., Weinzierl, A.: Preference-based inconsistency assessment in multi-context systems. In: Janhunen, T., Niemelä, I. (eds.) Logics in Artificial Intelligence. Lecture Notes in Computer Science, vol. 6341, pp. 143–155. Springer, Berlin (2010). http://dx.doi.org/10.1007/978-3-642-15675-5_14
15. Gabbay, D., Hunter, A.: Making inconsistency respectable: a logical framework for inconsistency in reasoning, part i—a position paper. In: Jorrand, P., Kelemen, J. (eds.) Fundamentals of Artificial Intelligence Research, Lecture Notes in Computer Science, vol. 535, pp. 19–32. Springer, Berlin (1991). http://dx.doi.org/10.1007/3-540-54507-7_3
16. Gomez, S.A., Chesnevar, C.I., Simari, G.R.: Reasoning with inconsistent ontologies through argumentation. Appl. Artif. Intell. **24**(1–2), 102–148 (2010)
17. Grant, J.: Classifications for inconsistent theories. Notre Dame J. Form. Log. **19**(3), 435–444 (1978)
18. Grant, J., Hunter, A.: Measuring inconsistency in knowledgebases. J. Intell. Inf. Syst. **27**(2), 159–184 (2006)
19. Grant, J., Hunter, A.: Measuring consistency gain and information loss in stepwise inconsistency resolution. In: Liu, W. (ed.) Symbolic and Quantitative Approaches to Reasoning with Uncertainty. Lecture Notes in Computer Science, vol. 6717, pp. 362–373. Springer, Berlin (2011)
20. Huang, Z., Harmelen, F.: Using semantic distances for reasoning withinconsistent ontologies. In: Proceedings of the 7th International Conferenceon The Semantic Web, pp. 178–194. ISWC '08, Springer-Verlag, Berlin (2008)
21. Hunter, A.: How to act on inconsistent news: ignore, resolve, or reject. Data Knowl. Eng. **57**(3), 221–239 (2006). http://dx.doi.org/10.1016/j.datak.2005.04.005
22. Hunter, A., Konieczny, S.: Approaches to measuring inconsistentinformation. In: Bertossi, L., Hunter, A., Schaub, T. (eds.) Inconsistency Tolerance,Lecture Notes in Computer Science, vol. 3300, pp. 191–236. Springer, Berlin (2005). http://dx.doi.org/10.1007/978-3-540-30597-2_7
23. Kühnberger, K.U., Geibel, P., Gust, H., Krumnack, U., Ovchinnikova, E., Schwering, A., Wandmacher, T.: Learning from inconsistencies in an integrated cognitive architecture. In: Proceedings of the 2008 conference on Artificial General Intelligence 2008, pp. 212–223. IOS Press, Amsterdam, The Netherlands, The Netherlands (2008). http://dl.acm.org/citation.cfm?id=1566174.1566194
24. Lenzerini, M., Savo, D.F.: Updating inconsistent description logic knowledge bases. In: ECAI. pp. 516–521 (2012)
25. Ligęza, A.: Intelligent data and knowledge analysis and verification; towards a taxonomy of specific problems. In: Vermesan, A., Coenen, F. (eds.) Validation and Verification of Knowledge Based Systems, pp. 313–325. Springer, US (1999). http://dx.doi.org/10.1007/978-1-4757-6916-6_21
26. Ligęza, A.: A 3-valued logic for diagnostic applications. In: Yannick, P., Alexander Feldman, A.G. (ed.) Diagnostic REAsoning: Model Analysis and Performance, August, 27th, Montpellier, France (2012). http://dreamap.sciencesconf.org/conference/dreamap/eda_en.pdf

27. Ligęza, A., Kościelny, J.M.: A new approach to multiple fault diagnosis: a combination of diagnostic matrices, graphs, algebraic and rule-based models. the case of two-layer models. Appl. Math. Comput. Sci. **18**(4), 465–476 (2008)
28. Ligęza, A., Nalepa, G.J.: A study of methodological issues in design and development of rule-based systems: proposal of a new approach. Wiley Interdisc. Rev.: Data Min. Knowl. Discovery **1**(2), 117–137 (2011)
29. Ma, Y., Hitzler, P.: Paraconsistent reasoning for OWL 2. In: Polleres, A., Swift, T. (eds.) Web Reasoning and Rule Systems, Lecture Notes in Computer Science, vol. 5837, pp. 197–211. Springer, Berlin (2009)
30. Ma, Y., Qi, G., Hitzler, P.: Computing inconsistency measure based on paraconsistent semantics. J. Logic Comput. **21**(6), 1257–1281 (2011)
31. Nalepa, G., Bobek, S., Ligęza, A., Kaczor, K.: Algorithms forruleinference in modularized rule bases. In: Bassiliades, N., Governatori, G.,Paschke, A. (eds.) Rule-Based Reasoning, Programming, and Applications. Lecture Notes in Computer Science, vol. 6826, pp. 305–312. Springer, Berlin (2011)
32. Nalepa, G., Bobek, S., Ligęza, A., Kaczor, K.: HalVA—ruleanalysisframework for XTT2 rules. In: Bassiliades, N., Governatori, G., Paschke, A.(eds.) Rule-Based Reasoning, Programming, and Applications. Lecture Notes inComputer Science, vol. 6826, pp. 337–344. Springer, Berlin (2011). http://www.springerlink.com/content/c276374nh9682jm6/
33. Nalepa, G.J.: PlWiki—a generic semantic wiki architecture. In: Nguyen, N.T., Kowalczyk, R., Chen, S.M. (eds.) Computational Collective Intelligence. Semantic Web, Social Networks and Multiagent Systems, First InternationalConference, ICCCI 2009, Wroclaw, Poland, October 5–7, 2009. Proceedings. Lecture Notes in Computer Science, vol. 5796, pp. 345–356. Springer (2009)
34. Nalepa, G.J.: Collective knowledge engineering with semantic wikis. J. Univ. Comput. Sci. **16**(7), 1006–1023 (2010). http://www.jucs.org/jucs_16_7/collective_knowledge_engineering_with
35. Nalepa, G.J.: Loki—semantic wiki with logical knowledgerepresentation. In: Nguyen, N.T. (ed.) Transactions on Computational Collective Intelligence III, Lecture Notes in Computer Science, vol. 6560, pp. 96–114. Springer (2011). http://www.springerlink.com/content/y91w134g03344376/
36. Nalepa, G.J., Ligęza, A., Kaczor, K.: Formalization and modeling of rules using the XTT2 method. Int. J. Artif. Intell. Tools **20**(6), 1107–1125 (2011)
37. Newell, A.: The knowledge level. Artif. Intell. **18**(1), 87–127 (1982)
38. Nguyen, N.T.: Advanced Methods for Inconsistent Knowledge Management (Advanced Information and Knowledge Processing). Springer, London (2008)
39. Nuseibeh, B., Easterbrook, S., Russo, A.: Making inconsistency respectable in software development. J. Syst. Softw. **58**(2), 171–180 (2001)
40. Pniewska, J., Adrian, W.T., Czerwoniec, A.: Prototyping: is it a more creative way for shaping ideas. In: Proceedings of the International Conference on Multimedia, Interaction, Design and Innovation. pp. 18:1–18:8. MIDI '13, ACM, New York, NY, USA (2013). http://doi.acm.org/10.1145/2500342.2500361
41. Priest, G., Tanaka, K., Weber, Z.: Paraconsistent logic. In: Zalta, E.N. (ed.) The Stanford Encyclopedia of Philosophy. Summer 2013 edn. (2013)

Modeling and Empirical Investigation on the Microscopic Social Structure and Global Group Pattern

Zhenpeng Li and Xijin Tang

Abstract In this paper, we investigate the microscopic social mechanisms through agent-based modeling and empirical data analysis with the aim to detect the intrinsic link between local structure balance and global pattern. The investigation based on Hopfield model suggest that three types of social influences give rise to the emergence of macroscopic polarization, and the polarization pattern is closely linked with local structure balance. In addition, the corresponding empirical examples are provided to verify the social mechanisms and model simulation results.

Keywords Social influences · Agents-based modeling · Empirical data analysis · Structure balance · Macroscopic polarization

1 Introduction

Global polarization is widely observed in human society. Examples about the group behaviors patterns include culture, social norms, political election, and online public debates on highlighted issues. In politics, polarization refers to the divergence of political attitudes to ideological extremes. Polarization can refer to such divergence in public opinion as a whole, or to such divergence within a particular subset or subgroup. Polarization as one of the interesting social phenomena is

Z. Li
Department of Statistics, School of Mathematics and Computer Science, Dali University, Dali 671003, China
e-mail: lizhenpeng@amss.ac.cn

X. Tang (✉)
Academy of Mathematics and Systems Sciences, Chinese Academy of Sciences, Beijing 100190, China
e-mail: xjtang@amss.ac.cn

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_3

widely studied in many fields, such as social science, economics, and mass communication.

The implications of group polarization are not entirely clear and may include some beneficial as well as detrimental consequences. From decision-making perspective, the global pattern of collective dynamics is rooted from individuals' micro-level decision-making processes, where social influence as one of the important social psychological factors, plays a dominant role. Generally to say, from the social influence point of view, three types of impact run through the whole processes of group decision-making especially in voting. One is positive influence among in-group members; this kind of social force accelerates intra-group opinion convergence. The second one is the negative social impact which may block the formation of consensus among different groups. The third type refers one kind of special individuals' attitudes, or a state that the individuals do not belong to any labeled subgroups; members in the group have no common social identity, no firm stand about some opinions and are in a state of neither fish nor fowl [10–12].

From a bottom-up point of view, in modeling social processes, individuals' local cumulative interacting behaviors would evolve into different global patterns. The emergence of global features comes from the local interconnecting mechanism, e.g., short average path and high clustering coefficients contribute to small world mechanism [17]. However, it is difficult to infer how the global patterns are generated from certain simple local aggregated social processes. In such cases, agent-based simulation technique is a useful tool. The goal of agent-based modeling is to enrich our understanding of fundamental processes that may appear in a variety of applications. This methodology is useful as demonstrated by Reynolds [16] when he modeled the movement of a population of artificial "boids" based on three simple rules:

- *Separation*: Do not get too close to any object, including other boids.
- *Alignment*: Try to match the speed and direction of nearby boids.
- *Cohesion*: Head for the perceived center of mass of the boids in your immediate neighborhood.

From then on, agent-based models of human social interaction based on this same theory-building strategy are becoming acceptable potential and powerful tools for sociologists. Recent developments demonstrate how this technique can provide sociologists with a more rigorous method for specifying the micro foundations of global patterns at the relational level. For example, agent-based models (ABMs) can show how simple and predictable local interactions can generate familiar but enigmatic global patterns, such as the diffusion of information, emergence of norms, coordination of conventions, or participation in collective action [15].

In this study, as well as in its earlier version [13], agent-based modeling is used to investigate how social influences lead to group polarization and the intrinsic link which forms local influence structure balance with the emergence of global bipolarization pattern. Besides we use online empirical data analysis to verify the simulation conclusion.

The rest of the paper is organized as follows: in Sect. 2 we present global balance index (GBI). In Sect. 3 we examine the relationship between social influence and bipolarization through Hopfield attractor model (HAM), especially we focus on the connection between global polarization pattern and triad balance at the micro-level. In Sect. 4 we use empirical data analysis results to show that HAM makes sense to explain the real-world phenomenon. Section 5 presents our concluding remarks.

2 Global Balance Index and Local Structure Balance

Social balance theory provides us an interesting theoretical base to analyze how a social group evolves to the possible balance state. In this study, we investigate how influence signs $(-1, +1, 0)$ change at the dyadic level affects the global (collective) balance state in the whole interpersonal network.

It is assumed that the interpersonal network tends toward higher balance [8], or to be evolved with the probabilities as shown in Eq. (1)

$$\begin{aligned} &\Pr[R_{ij}=0 \to R_{ij}=1 | R_{ij}=0 \to R_{ij}=-1] \approx 1 \\ &\Pr[R_{ij}=1 \to R_{ij}=0 | R_{ij}=-1 \to R_{ij}=0] \approx 0. \end{aligned} \tag{1}$$

Next we introduce the global balance index (GBI), with the aim of application to measure group voting polar patterns for Hopfield network simulation. Given a signed network, a fundamental problem is how to construct a dynamics of sign changes on the edges such that asymptotically the entire network is found on a perfectly balanced state. We use the GBI defined in [6] Eq. (2) to measure the network global balance level.

$$\mathrm{GBI} = \sum_{i,j=1}^{n} (1 - s_i R_{ij} s_j)/2, \tag{2}$$

where the summation runs over all adjacent pairs of nodes, $s_i \in \{\pm 1\}$, $i = 1, \ldots, n$ represents agent i' s opinion, and R_{ij} represents the social influence which can be positive, negative and neutral, i.e., $R_{ij} \in \{+1, -1, 0\}$.

Computing global balance means assigning "+1" or "−1" to each node in the network so as to minimize the GBI. GBI approximates to 0 means that interpersonal network reaches a global structure balance state. We randomly generate a 10 × 10 interpersonal network and 1 × 10 opinion vector. Figure 1 shows the dynamic signed influence network balance processes with using GBI as one measure, we simulate 10 times and observe that each time the GBI reaches 0, i.e., the social network reaches a global network balance state within 1200 steps.

Globally, after the interpersonal network reaches stable balance level, we detect the local triadic interpersonal structure by using *R* package *sna* [2]. Figure 2

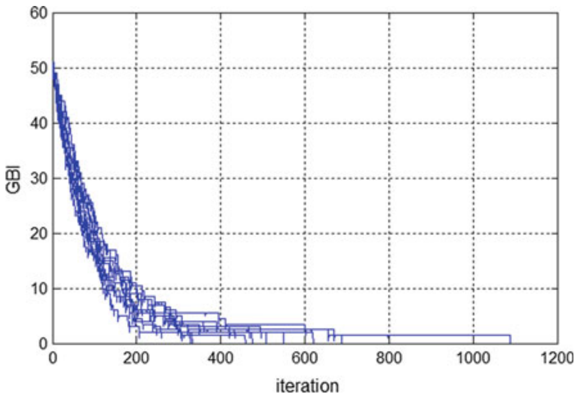


Fig. 1 Simulation results of GBI

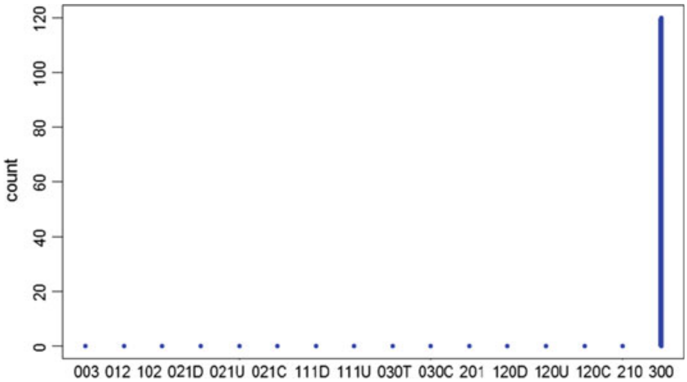


Fig. 2 16 type's triadic distribution after GBI reaching 0

illustrates that code 300 is the only triadic structure remained, which according to [8] suggests that when GBI = 0, local interpersonal relation attains structure stable state.

Next we illustrate the local structure balance and global polarization pattern using the Hopfield attractor model by considering the dyadic ties.

3 Hopfield Attractor Model with Dyadic Influence Ties

3.1 *The Hopfield Attractor Model*

Macy et al. [14] presented a Hopfield model to describe group polarization problems, with considering individual decision-making dimensions, social ties signs, strengths, culture dissemination theories, etc. The modeling mechanism is as follows. The cumulative social pressure of each individual i to choose one opinion s_i is denoted as in Eq. (3),

$$P_{is} = \frac{\sum_{j=1}^{N} I_{ij} s_j}{N-1}, j \neq i \tag{3}$$

where $s_i = \pm 1$ represents binary voting opinions, N is group size, I_{ij} is the weight of the social ties that individual j $(j \neq i)$ connect to i, and the matrix I is termed social influence matrix.

Comparing with Macy et al. [14], with the motivation of investigating the relationship between non-positive social influence and group opinions polarization, instead assigning continuous values between -1 and $+1$ to social ties I_{ij}, we assign three discrete values -1, $+1$ and 0 to I_{ij} to indicate the three types of dyadic social influence as defined in [10–12]. Furthermore, by considering the external intervention $X_i(-0.5 \leq X_i \leq 0.5)$, i.e., the influence to the individual's opinion comes from other out-group's impact; the cumulative social pressure is computed by Eq. (4).

$$\tau_{is} = \frac{v_s}{1 + e^{-KP_{is}}} + (1 - v_s) X_i. \tag{4}$$

where $v_s (0 \leq v_s \leq 1)$ is used to trade off the internal and external group influence to the individual i's opinion, and K is the size of opinions dimension. Given a randomly selected threshold $\pi(0 < \pi < 1)$, if $\tau_{is} \geq \pi$, individual i chooses +1 (support), else chooses −1 (oppose). Equation (5) describes the update of influence processes of individual j to i $(j \neq i)$,

$$I_{ij}(t+1) = (1-\alpha) I_{ij}(t) + \frac{\alpha}{K} \sum_{k=1}^{K} s_{jk}(t) s_{ik}(t), j \neq i, \tag{5}$$

where t is the time step, α is an adjustable parameter between 0 and 1. Based the Hopfield attractor model, [12] extended the dyadic influence structure to triadic scenarios. They also discussed the intrinsic connection between local dyadic and triadic structures balance and global bipolarization pattern.

Next we illustrate the local structure balance and global polarization pattern using the Hopfield attractor model by considering the dyadic ties.

3.2 Analysis of Simulation Results

We take the test by setting $N = 20$, $K = 5$, π belong to [0, 1] uniform distribution and $\alpha = 0.5$. We run 100 times for average. Figure 3a shows the group initial

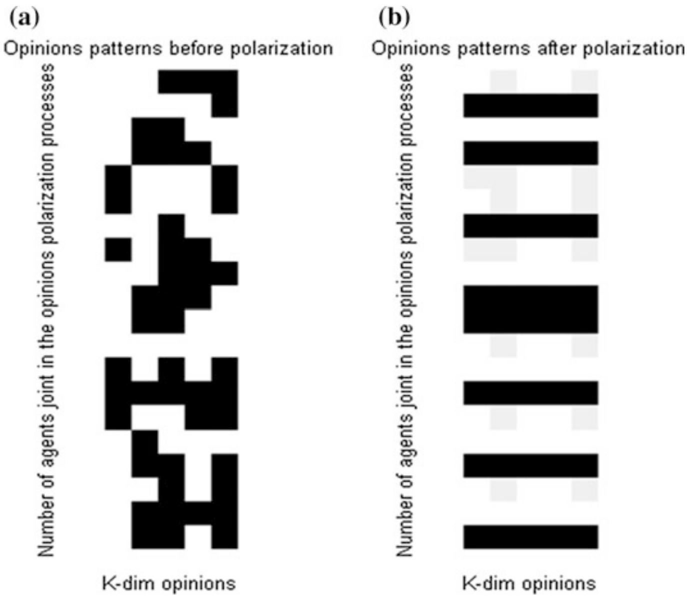


Fig. 3 Group opinion before and after polarization under the condition of imposing three types of social influence. (We generate $N \times K$ matrix which N denotes group size, K denotes the numbers of options, $N = 20$, $K = 5$)

random opinions states when each agent i faces K dimension decision making (before self-organizing polarization). Figure 3b illustrates the group bipolarization state under the condition of no external influence ($v_s = 1$) and with three types of influence. We can observe that two patterns appear after group polarization, i.e., one pattern is (−1, −1, −1, −1, −1), i.e., (black, black, black, black, black) (marked by v_1), the other is (+1, +1, +1, +1, +1), i.e., (white, white, white, white, white) (represented by v_2). The ratio of the 2-pattern size approximates to 1:1.

The relationship between exogenous intervention parameter v_s to group polarization is as shown in Fig. 4. We see that when $v_s = 1$ (no external intervention to the group interaction processes), the ratio of is $v_1/(v_1 + v_2)$ approximate to 0.5. However, the fifty to fifty well-matched equilibrium will be destroyed with a little cut off v_s. In other words, external intervention will lead to the majority pattern. In particular, when $v_s = 0.5$, i.e., group opinion is evenly affected by external and internal factors, we observe the group consensus appears, i.e., $v_1/(v_1 + v_2)$ approximates to 1, the pattern v_2 nearly disappears. It is clearly suggested that, under the condition of imposing external intervention, the group reaches majority or consensus pattern. With no exogenous impact, the group evolves into bipolarization state in the end. We also simulate the case with no negative social influence. The modeling result shows that under the condition with no external intervention, the group reaches the majority or consensus pattern. The result might conclude that non-positive social influence promotes group opinion bipolarization, which is

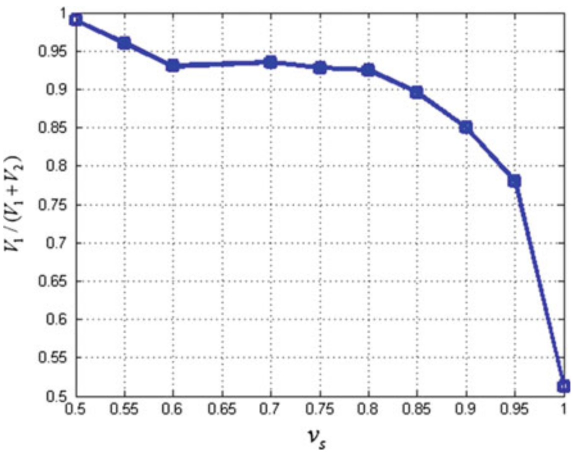


Fig. 4 Exogenous intervention impact on group polarization (In Eq. (4), we adjust v_s from 1 to 0.5, i.e., increasing the out-group intervention from 0 to 0.5, by which we observe impact of the exogenous intervention on group opinions polarization)

consistent with our previous conclusion in [12]. Furthermore, we investigate the triadic relation motifs distribution before and after bipolarization by using R package *sna* [2]. We find that the overwhelming structure balance[1] motifs emerge concurrently with polarization process.

Figure 5 presents the dynamic variation of triadic distributions. The upper plot in Fig. 5 shows the initial local triads distribution as t = 0 according to randomly generated social influence matrix. We can observe that all 16 types of triads exist in the initial triadic relationships. With the social influence matrix updating, at step $t = 19$ some triad motifs disappear, e.g., Code 003, Code 012, and Code 102, while Code 300, Code 210 become dominant (see middle plot in Fig. 5). Finally, at step $t = 29$, other triad motifs disappear except balanced triad motif Code 300 (as shown in the bottom plot of Fig. 5), which means that GBI = 0 and the group achieves the stable state.

The simulation results of Hopfield attractor model suggest that non-positive social influence promotes group bipolarization pattern. In addition, HAM simulation result concludes that opinions polarization in a group is coexisted with local level structure balance, which reveals some interesting internal connection between global collective pattern and local social structure stability.

[1]Holland and Leinhardt [9] addressed that classic balance theory offers a set of simple local rules for relational change and classified local triadic motifs into 16 types, according to mutual reciprocity, asymmetry relation and non-relationship between pairs (or dyadic relations), where Code 300 triad relation corresponding to structure balance under the condition of the triad product signs satisfies "+". More details about structure balance refer to [3, 5, 8, 9].

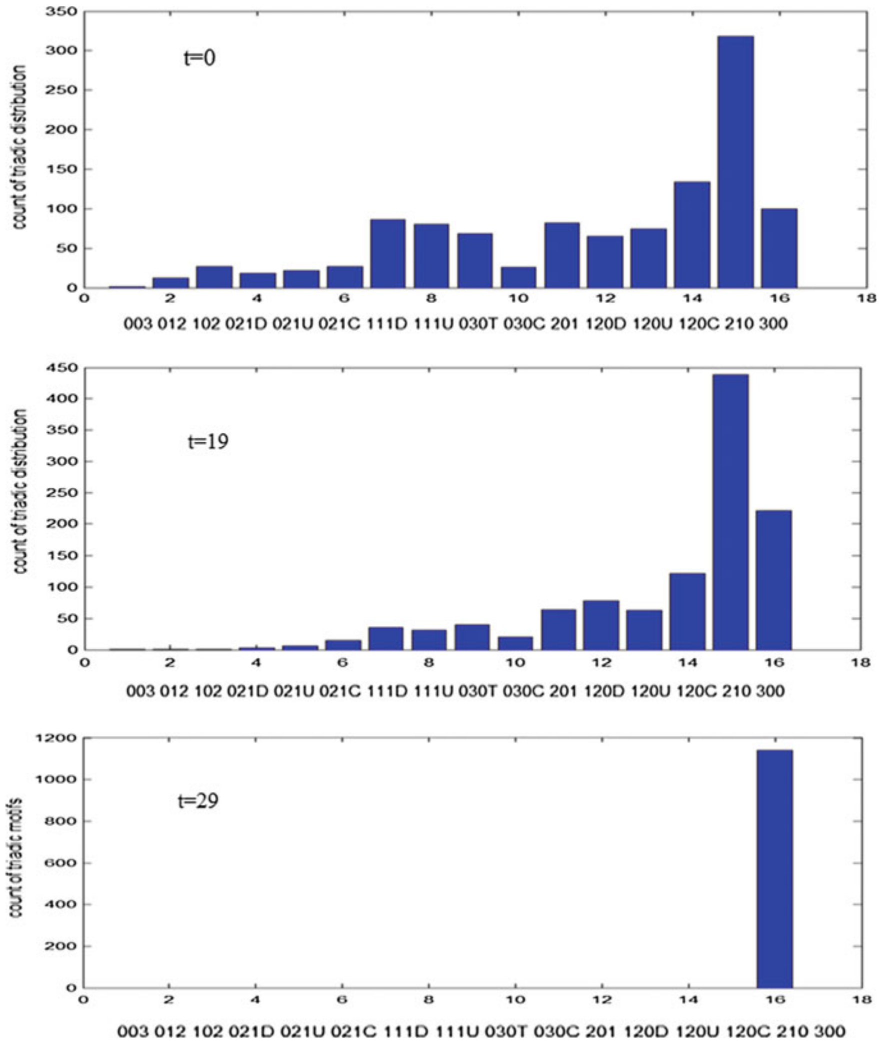


Fig. 5 Dynamic variation of triads distribution (at t = 29, GBI = 0)

4 The Evidences from Real World

In this section, we present two empirical illustrations. One shows that non-positive influence as one important factor that promotes voters' opinions polarization. Another suggests that online social network demonstrates the local structure stable characteristic.

4.1 Political Polarization on Twitter

Social media play an important role in shaping political discourse in the US and around the world. Some empirical evidences reveal that politically active Web 2.0 enabled social media users tend to organize into insular, homogenous communities segregated along partisan lines. For example, Adamic and Glance [1] demonstrated that political blogs preferentially linked to other blogs of the same political ideology. Their finding is also supported by Hargittai et al. [7], which shows that Internet users who share same political point of view tend to behave similarly, such as choose to read blogs that share their political beliefs.

More insightful evidences came from Conover et al. [4] who studied some networks of political communication on the Twitter microblogging service during the 6 weeks prior to the 2010 US midterm elections. That study shows that the retweet network exhibits a highly modular structure, segregating users into two homogeneous communities corresponding to the political left and right, which demonstrate the obvious bipolarization characteristic. A surprising contrast is that the mention network does not exhibit this kind of political polarization pattern, resulting in users being exposed to individuals and information they would not have been likely to choose in advance. Figure 6 displays this kind of group opinions dynamic pattern.

Their findings suggest that the possibility that in–group/out-group differentiation and rejection antagonism, or the non-positive social influence between intra-group members and intergroup members is the emergent cause of social network self-organization, and leads to the bipolarization global dynamic pattern.

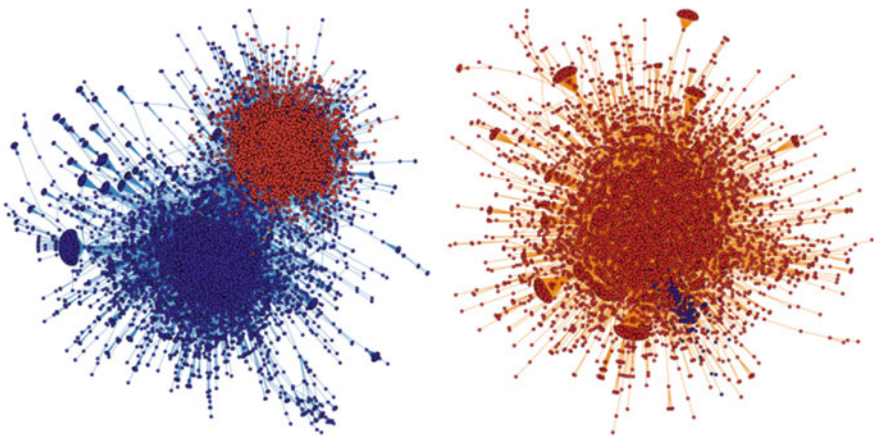


Fig. 6 The political retweet (*left*) and mention (*right*) networks. The color of node reflects cluster assignments. Community structure is evident in the retweet network, but less so in the mention network. In the retweet network, the red cluster *A* is made of 93 % right-leaning users, while the blue cluster *B* is made of 80 % left-leaning users [4]

4.2 Local Structure Balance on the Trust Network of Epinions

In this section, we will detect the local structure balance among mixed relationships, i.e., positive, negative, and neutral. We use the trust network constructed with the product reviews in the Epinions Web site, where people can not only indicate their trust, distrust, or no comments of the reviews of others, but also give both positive and negative ratings either to items or to the other raters. In our study, we analyze the Epinions dataset with 131828 nodes and 841372 edges. The basic statistics about Epinions data set is as listed in Table 1.

In Table 1, "+Δ" stands for the triadic relationship that satisfies structure balance, "−Δ" denotes for the unbalance triadic relation. "+Edges" stands for the trust relationship, "−Edges" denotes for the opposite. Figure 7 shows that most of the edges are embedded into smaller size community (subgroup), and only a few edges are embedded into high density modular structure. However, it is worth noting that in both scenarios the balanced triads distribution is higher than that of the unbalanced counterparts.

We also analyze the 16 types triads distribution for three cases, trust or positive "+," distrust or non-positive "−," mixed or including three scenarios no comments relationships by "0," and "+," "−," correspondingly. Our findings are that the size of local triadic balance structure such as 16-#300, 3-#102, 11-#201 in the network constructed by trust edges (or positive "+" relationship) and mixed relationship (including "+," "0," and "−" links) is more than that in the distrust (or negative "−" relationship) network, as shown in Fig. 8. These observations fit well into Heider's

Table 1 Epinions dataset statistics

Nodes	Edges	+Edges	−Edges	+Δ	−Δ
13, 1828	84, 1372	71, 7129	12, 3670	3368, 2027	299, 4773

http://www.datatang.com/

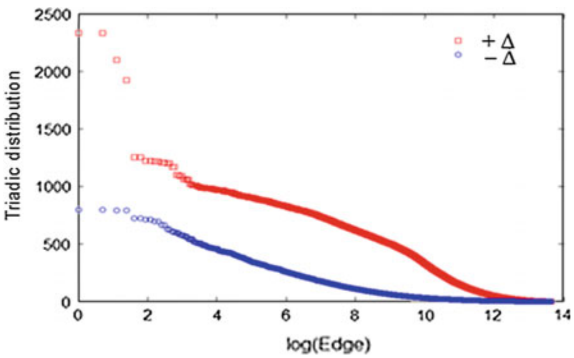


Fig. 7 Balanced triads and unbalanced triads distribution

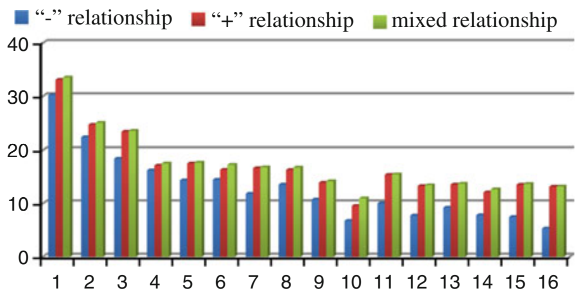


Fig. 8 16 types of triads statistic distribution of three networks

original notion of structural balance, which in its strongest form postulates that when considering the relationships between three people, either only one or all three of the relations should be positive.

5 Conclusions

In this paper, we have investigated the non-positive social impact on group polarization based on agent-based modeling and real-world data analysis. By simulation we find that bipolarization pattern tends to emerge with no external intervention, and consensus may occur among group members if the non-positive influence is neglected.

With dyadic influence considered, by HAM simulation we observe that dyadic influence balance among agents has inherent relation with global bipolarization pattern. This is similar to the macro–micro linkage: sentiment relations among agents (localized as triad and dyad) lead to the collective balance of the group.

Two empirical evidences are consistent with our modeling results. The bipolarization on Twitter shows that no-positive influence as an important factor promotes voters' opinions polarization, while Epinions' trust network demonstrates the local structure stable characteristic existed in the online signed social network. Those data analyzing results are closely connected to the theories of classic social psychology and structure balance which tend to capture certain common patterns of interaction.

Why are some communities, organizations, and social group characterized by cooperation, trust, and solidarity, i.e., consensus pattern, whereas others are fraught with corruption, discord, and fear, i.e., unconformity or polarization pattern? We argue that viewed from the bottom up, the answer centers on global pattern that has its micro-level social mechanisms. By these underlying principles, decision making is distributed and global order self-organizes out of multiplex local interactions among autonomous interdependent actors.

Acknowledgments This research is supported by National Basic Research Program of China under Grant No. 2010CB731405, National Natural Science Foundation of China under Grant No. 71171187, and Research Fund of Dali University (No. KYBKY1219210110).

References

1. Adamic, L., Glance, N.: The political blogosphere and the 2004 U.S. election: divided they blog. In: Proceedings of 3rd International Workshop on Link Discovery (LinkKDD 2005), pp. 36–43 (2005)
2. Butts, C.T.: Social network analysis with SNA. J. Stat. Softw. **24**(6), 1–51 (2008)
3. Cartwright, D., Harary, F.: Structural balance: a generalization of Heider's theory. Psychol. Rev. **63**, 277–292 (1956)
4. Conover, M.D., Ratkiewicz, J., Francisco, M., Goncalves, B., Flammini, A., Menczer, F.: Political polarization on twitter. In: Proceedings of the 5th International Conference on Weblogs and Social Media 2011, Barcelona, Spain, pp. 237–288 (2011)
5. Davis, J.A., Leinhardt, S.: The structure of positive interpersonal relations in small groups. In: Berger, J. (ed.) Sociological Theories in Progress, vol. 2, pp. 218–251. Houghton Mifflin, Boston (1972)
6. Facchetti, G., Iacono, G., Altafini, C.: Computing global structural balance in large-scale signed social networks. PNAS **108**(52), 20953–20958 (2011)
7. Hargittai, E., Gallo, J., Kane, M.: Cross ideological discussions among conservative and liberal bloggers. Public Choice **134**(1), 67–86 (2007)
8. Heider, F.: Attitudes and cognitive organization. J. Psychol. **21**, 107–112 (1946)
9. Holland, P.W., Leinhardt, S.: A method for detecting structure in sociometric data. Am. J. Sociol. **70**, 492–513 (1970)
10. Li, Z.P., Tang, X.J.: Group polarization and non-positive social influence: a revised voter model study. In: Hu, B., Liu, J., Chen, L., Zhong, N. (eds.) Brain Informatics—International Conference, BI 2011, Lanzhou, China, September 7–9, 2011. In: Proceedings Lecture Notes in Computer Science, vol. 6889, pp. 295-303. Springer, Berlin (2011)
11. Li, Z.P., Tang, X.J.: Polarization and non-positive social influence: a hopfield model of emergent structure. Int. J. Knowl. Syst. Sci. **3**(3), 15–25 (2012)
12. Li, Z.P., Tang, X.J.: Group polarization: connecting, influence and balance, a simulation study based on hopfield modeling. In: PRICAI 2012, pp. 710-721 (2012)
13. Li, Z.P., Tang, X.J.: Modeling and empirical investigation on the microscopic social structure and global group pattern. In: Skulimowski A.M.J. (ed.) Looking into the Future of Creativity and Decision Support Systems: In: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems, Kraków, Poland, 7–9 Nov 2013. Advances in decision sciences and future studies, vol. 2, pp. 339–349. Progress & Business Publishers, Kraków (2013)
14. Macy, M.W., Kitts, J.A., Flache, A.: Polarization in dynamic networks a hopfield model of emergent structure. In: Breiger, R., Carley, K., Pattison, P. (eds.) Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers, pp. 162–173. The National Academies Press, Washington DC (2003)
15. Macy, M.W., Willer, R.: From factors to actors: computational sociology and agent-based modeling. Ann. Rev. Sociol. **28**, 143–166 (2002)
16. Reynolds, C.W.: Flocks, herds and schools: a distributed behavioral model. In: ACM SIGGRAPH Computer Graphics ACM 1987, **21**(4), pp. 25–34 (1987)
17. Watts, D.J., Strogatz, S.: Collective dynamics of "small-world" networks. Nature **393**(6684), 440–442 (1998)

Human-Like Knowledge Engineering, Generalization, and Creativity in Artificial Neural Associative Systems

Adrian Horzyk

Abstract This paper presents new approaches to form knowledge and achieve generalization and creativity in a human-like way using artificial associations in artificial neural associative systems. We can assume that knowledge is automatically formed in a human brain that is able to associate many objects, facts, rules, needs, and emotions. The brain structure is very specific and does not reflect architectures of today's computers. This paper describes a neuronal structure that is able to represent various data combinations and their sequences in an aggregated form, which allows to recall consolidated facts and rules by triggering various neuron combinations. New combinations of initially triggered neurons recall even new facts and rules as a result of high-level generalization that can be treated as creative behavior. This paper provides the description of artificial neural associative systems that enable to create knowledge in a human-like associative way and to trigger artificial associations.

Keywords Associative knowledge engineering · As-knowledge · Knowledge representation · Creativity · Artificial associative systems · Emergent cognitive architectures · As-neuron · Semassel

1 Introduction

The aim of this paper is to provide the fundamentals for automatic associative knowledge formation in artificial neural associative systems [3–6] that can use a special kind of neurons (as-neurons) to represent a set of training sequences consisting of objects. The objects are represented by as-neurons and can be defined by vari-

A. Horzyk (✉)
Faculty of Electrical Engineering, Automatics, Computer Science,
and Biomedical Engineering, Department of Automatics and Biomedical Engineering,
AGH University of Science and Technology, Mickiewicza Av. 30,
30-059 Krakow, Poland
e-mail: horzyk@agh.edu.pl
URL: http://home.agh.edu.pl/~horzyk

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_4

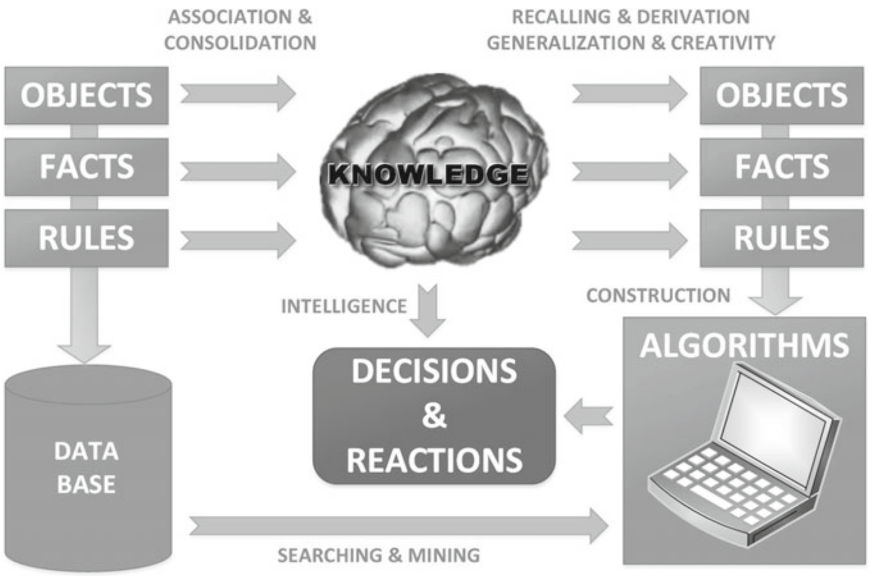


Fig. 1 Diagram of the natural and computational processes using objects, facts, rules, knowledge, and intelligence to reach a decision or to induce a reaction

ous combinations of sensorial input data and other objects. Knowledge is essential for all mental processes of living creatures, especially for their intelligent behavior. Knowledge engineering is also key to solving some of the fundamental challenges of today's computer science. It facilitates more efficient and intelligent computer systems and further development of artificial intelligence. Knowledge can be developed, expanded, verified, changed, and specified in a brain during one's lifetime. Facts, rules, and algorithms are the external products of intelligence and knowledge (Fig. 1). They are recalled as sequences of elements that have been represented and associated in a brain. Nowadays, knowledge is often treated as a set of facts and rules, but in reality they are only derived from knowledge and cannot represent the entirety of it [9–11]. Knowledge is also defined as an inner configuration of a neural network that has been created in a training process for a given set of training samples [19].

Knowledge is used in cognition and decision processes that are based on rules that often result from various multicriteria optimization problems [16]. The freedom of choice in decision-making problems determines the intelligence of the system and can manifest creativity related to the hidden internal states of the decision-making process [11, 16–19]. Computational creativity has been developing for many years in many areas: music, vision, art, linguistic, and for problem solving [15, 20, 22]. Creativity is usually modeled as an effect of approximation, extrapolation, generalization, addition of noise, damage, and disordering effects [2, 11, 15, 19, 21, 23]. Researchers usually define creativity as the development of ideas, products, or solutions that are perceived as unique, novel, relevant, and useful [2, 15, 16, 23]. The definitions of creativity define external manifestations of creativity but do not explain where creativity comes from, and how it is produced in brain structures. The following sections try to answer to these questions in terms of associative and generalization processes that take place in the biological and artificial associative neural systems.

Human knowledge fails to work as a relational database because it requires fast and direct neuron communication to consolidate fact and rules and quickly recall associations. This is because relational databases function by looping and searching for specific data that are passively stored [4]. Moreover, brains of living creatures have no suitable structures for storing data tables that are currently commonly used in today's computer systems. Various kinds of artificial neural networks currently used in computational intelligence can be trained and used to formulate answers and even generalize about the trained objects. This kind of generalization is usually based on multiparametric approximation functions that are represented by neural networks and a special kind of nonlinear neurons. Furthermore, taking into account the usual frequencies of biological neuron activations (that occur between 33 and 83 ms [12]) and the usual measured intervals between presentations of various stimuli and reactions to them (usually between 300 and 1100 ms [7]), we can notice that only from 4 to 25 sequential interneuronal computational steps can be processed in a human brain. Thus, each associative reaction to presented stimuli takes constant time in a human brain. There is no place and no time for looping or searching through large data tables. The classical computer system stores various collections of data, and when necessary it tries to search them through many times to find data that are useful for some given reasons or tasks (Fig. 1). In the classical computer methodology hardly any task has constant computational complexity [14], but a brain performs computations in a quite different way. This would not be possible without knowledge that affects and steers associative processes and can indicate possible ways to solutions. The solutions can be found within the gained knowledge, especially due to the ability to generalize and create new variations of known objects, facts, rules, processes, methods, or algorithms that are represented and associated in an associative system.

'Associate' means to consolidate and actively connect neural representations of objects, rules, etc., to the existing ones. Such consolidations of objects and rules do not form knowledge that can be used for reasoning. This paper introduces 'active association' that not only add and consolidate objects and rules but represent them actively in the associative neural system that consists of as-neurons that can recall consolidated objects and rules if being triggered by external or internal excitations. Moreover, some new external combinations of neuron excitations can recall even new behaviors (e.g., rules) that have not been trained. Such generalization process is a result of consolidation and aggregation of learned facts and rules that have constituted the knowledge. This kind of generalization is very similar to human creativity [2, 15, 16, 23] that usually is based on previous experiences, knowledge, new inspiring surroundings, contexts of thinking, etc. The associations work without rules, conditions, and high-complexity algorithms of contemporary computer science. They are very fast, though they cannot always satisfy us. This is why we need to constantly expand our knowledge to improve the quality, precision, and efficiency of our thinking processes and conclusions. Brains and nervous systems of living creatures have a different architecture and way of data processing from today's computers.

Artificial associative systems (called aas-systems or artificial as-systems) carry out associative neurocomputations (called as-neurocomputing or as-computing) on graphs that are mainly built of neurons that are able to associate (called as-neurons) and sense receptors embedded in interneuronal space [3–6]. They can associate and consolidate various kinds of information and form knowledge in the associative human-like way that enables contextual recalling of associated data, their combinations, and sequences. Aas-systems can be trained using sequences of data combinations and contexts of previous associations. The time lapse and plasticity of these systems are indispensable for reproducing associative processes, using context of previous states, form knowledge, and achieve generalization and creativity on the sequence and rule levels.

2 Associative Model of Neurons

The associative model of neurons (called as-neurons) [3] reflects important features and functions of biological neurons [1, 7, 8, 12]. The as-neurons expand the artificial McCulloch-Pitts models and spiking models of neurons [19, 24]. This model tries to represent frequent subcombinations of input data and to connect to the other neurons that are frequently simultaneously activated with the other neurons or their activations come in close periods of time one after another. The as-neuron gradually, conditionally, and dynamically changes its size (influencing sensitivity), transmission parameters (synaptic weights), and connections to other neurons and sense receptors. It automatically and gradually relaxes in time and is switched into the refraction process after achieving its spiking threshold and activation. When the as-neuron achieves its spiking threshold, it is automatically activated and quickly starts to influence other connected as-neurons, change parameters of presynaptic and postsynaptic connections, and conditionally create or change its synaptic connections. In the refraction state, the as-neuron gradually returns to its resting state. During this time, it is less sensitive to next input stimuli until it exceeds its resting state again. The as-neurons interact not only through synapses but also through interneuronal space that allows them to connect and create new associative relations (as-relations), reproducing frequent real relations between objects. Such connections are usually created for the as-neurons that are frequently activated in similar periods of time. The relaxation of an as-neuron means that it gradually returns to its resting state after excitation stimuli that no longer activate the neuron. The associative context for each as-neuron expires when it returns to its resting state. This gradual returning plays a very important role in contextual data processing because the context of the previous stimuli decreases in time according to the time that has passed since the as-neuron was excited, not activated. The time lapse determines relaxation of as-neurons and thus also their grade of contextual influence. It means that the actual state of an associative system is determined not only by activated as-neurons in a given moment but also by the excited and refracted as-neurons that are not in their resting states. The previously excited, but not yet activated, as-neurons can be activated faster in the future if other excitation stimuli occur before they return back to their resting states. The refracted as-neurons can be harder and slower to activate once again if other excitation stimuli occur before they return back to their resting states. These non-resting states of as-neurons significantly change the associative processing of next incoming data. The associations change and can run otherwise, thanks to these temporal non-resting states. It enables associative systems (as-systems) to change associations, activate alternative weaker associations, be creative, process sequences of data combinations spread over time (time-spread combinations), learn, adapt, and store them in their neuronal structures. Each non-resting state of each as-neuron describes a potential temporal context for further associative data processing (as-processing) and enables contextual associative reasoning (as-reasoning). To conclude, the as-neurons supply new abilities and ways to process data unlike the ones used in contemporary computer science.

The refraction period of neurons is also functionally very important [7]. This period begins after neuron activation. It temporarily makes neurons be not or less sensitive and reactive on next excitations coming from the other neurons during a short future period of time. It also enables other competing neurons to achieve their activations before currently refracted neurons. The change in activation order of neurons changes the associations and a way of data processing. It means that the inner dynamic associative program change and the associative system can perform other alternative reactions. Moreover, the neurons can directly or through special inhibition neurons actively block other neurons using inhibition neurotransmitters. In case of lateral inhibition, first activated competing neurons block or move away in time activations of other suppressible connected neurons. This kind of cooperative competition allows neurons to automatically divide and represent an input data space according to similarity and a frequency of repetition of input data combinations [3].

Each as-neuron represents a set of input combinations that are usually spread over time. This spread period of time begins in a moment when the neuron leaves its resting state at last before its activation and finishes in a moment of being activated. All time-spread combinations of input stimuli that can activate a given neuron define semantic associative elements (called ‘semassels’) that are fundamental for associative data processing and knowledge representation in the associative systems [3]. Each semassel can simultaneously represent many similar as well as completely different combinations of input data that can be additionally spread over time. The only condition to qualify the combination for being in a set of combinations defining the semassel is its ability to activate an as-neuron. This feature is very important in view of generalization and enables to create memory tracks that can alternatively activate neurons instead of real input stimuli coming from sense receptors. The semassels are not stable, but they change over time because they are represented by neurons that adapt themselves and optimize their activation frequencies to survive. Changing sets of combinations defining semassels change associations and semantic relations between them. The biological neurons cannot be activated too often nor too rarely. Too frequent activations of biological neurons deplete their stores of neurotransmitters, energy, and proteins and make them gather too many Ca^{2+} and Zn^{2+} ions, which can lead to their death (apoptosis) [1, 7, 8, 12]. Too rare activations of biological neurons or too weak combinations of input stimuli do not allow appropriate metabolism and efficient energy use, so they try to block ineffective frequent stimuli and spread combinations. It decreases synaptic transmissions for engaged synapses. Moreover, such neurons do not demand the constant sets of inputs. They can use various subcombinations of them.

The biological neurons can change their activation frequencies using a few built in plasticity mechanisms. The neuron sensitivity for input stimuli is dependent on the size of its perikaryon, the number of multiplied connections, and the number of postsynaptic neurotransmitter receptors that open ion gates. All these parameters can change in time. If a given neuron is activated too often it has to increase production of cellular elements that makes it growing. When the size of the neuron increases its sensitivity for input stimuli automatically decreases because there are required more ions to change its potential and achieve its threshold of activity. In result, only the strongest combinations will be represented in neurons [3]. All these input combinations can include excitation as well as inhibition stimuli that are spread over time. Moreover, if some competing neurons grow, and their sensitivities decrease, a subgroup of weaker combinations is rejected. Then the other competing neurons are relatively more sensitive and can start to represent these rejected combinations. The cooperatively competing neurons automatically divide the data combinations between them. The lateral inhibition additionally supports this mechanism.

The synaptic transmission can change in time according to the correlation of the synaptic activity and the postsynaptic as-neuron activity. If the excitation of the synapse affects the activation of the neuron in a short period of time, then the synapse is strengthen. On the other hand, if the excitation of the synapse does not trigger activation of the neuron in short period of time then the synapse is weaken [8]. These mechanisms are more forceful when the neuron is small and has not achieved its optimal frequency of activation. If the given neuron achieves its optimal frequency of activation the size changes of its perikaryon, and even changes in synaptic strength become weaker and slower. This natural process of activation frequency optimization enables neurons to automatically specialize over time and stabilize represented semassels [3]. If the input stimuli change over time, the neuron can start again the plasticity processes of its adaptation, frequency optimization, and tuning the represented semassel. At last, the biological non-optimized neurons usually plastically interconnect through the interneuronal space especially when they are often activated in close periods of time. The above-mentioned features and functions are modeled in the as-neurons [3]; however, it is not easy to simulate them on today's computers because they work, react, and change their parameters and states continuously and simultaneously in time. This model can be simplified for some groups of tasks by adapting its relaxation function (1), so that the state, synaptic weights, and other transmission parameters of the simplified as-neurons (2) can be updated in discrete moments of time, expressed in natural numbers:

$$exc_t^{SN} = \mathrm{NRF}_{\alpha,\beta}^{SQR}\left(exc_{t-1}^{SN}, \theta^{SN}, x_1^t, \ldots, x_K^t\right) = \sum_{k=1}^{K} w_k x_k^t + \\ + \begin{cases} \alpha\ exc_{t-1}^{SN} + \frac{(\alpha-1)\beta(exc_{t-1}^{SN})^2}{\theta^{SN}} & \text{for } 0 > exc_{t-1}^{SN} \\ \alpha\ exc_{t-1}^{SN} + \frac{(1-\alpha)\beta(exc_{t-1}^{SN})^2}{\theta^{SN}} & \text{for } 0 \leqq exc_{t-1}^{SN} < \theta^{SN} \\ -\beta\ \theta^{SN} & \text{for } exc_{t-1}^{SN} \geqq \theta^{SN} \end{cases} \tag{1}$$

where

α controls the total neuron relaxation speed, i.e., its returning to the resting state, $0 \leqq \alpha < 1$,
β controls the neuron relaxation speed in under-threshold and after-activation states, $0.5 \leqq \beta < 1$,
θ is the threshold above which the *SN* neuron is activated,
exc_{t-1}^{SN} is the previous excitation state of the *SN* neuron,
x_k^t is the *k*th input stimulus that can externally excite or inhibit (suppress) the *SN* neuron in *t*th moment of time,
w_k is the current weight value of the *k*th synapse.

The biological neurons do not significantly change their activation thresholds but can change their perikaryon sizes in order to well-tune their sensitivity to input stimuli. The as-neurons change their thresholds θ for the same reasons [3]. In this model the optimal frequency is not always possible to determine when neurons work and adapt to a given training sequence set. There can be measured the quotient of the number of the neuron activations to the number of all presented sequences of a given training sequence set. The optimal quotient can be determined individually for each data set and available as-neurons. They react on the presented combinations stretching unevenly their representation on a whole presented input space according to the presentation frequency of each training sequence and the repetitions of the existing subsequences (Fig. 2).

The above-described neuron relaxation function $\mathrm{NRF}_{\alpha,\beta}^{SQR}$ enables discrete changes of excitation in synchronized periods of time for all neurons for simplified artificial as-systems. Such time flow can be easy simulated on CPU or GPU, but such simplified discrete as-neurons lose some features. They cannot accurately compete for first activation moment and discrete updating of neurons allowing them to simultaneously overcome the activation threshold θ, i.e., $exc_t^{SN} > \theta$. This limits or even disables simulation of lateral inhibition, natural process of competing, and automatic covering of input data space by such as-neurons. This simplified model also limits ability to trigger alternative associations in proper moments. Despite these limitations, this model can be successfully applied to many interesting tasks and easy simulated on today's computers. It enables development and training discrete artificial associative systems using training sequences of input data combinations, and not only simple data sets consisting of separate data combinations. It allows these systems to form knowledge about the trained tasks and trigger sequences of associations for various time-spread combinations of input data. The trained associations are automatically generalized, triggering new sequences that differ from the trained ones. It enables knowledge-based creativity of the as-systems that is context-dependent and defined by initial excitations of neurons that trigger following associations.

The synaptic transmission in biological neurons dynamically changes over time. These changes can be short and temporal reproducing the phosphorylation of ion channels or can be durable after the physical changes in the number of neurotransmitter receptors [8, 12]. Moreover, the synaptic transmission between neurons can be multiplied making the transmission very solid and strong. Thanks to the temporal and stable mechanisms for changing synaptic transmission, each synapse can be used twice: first, for solid and durable memory, and second, for temporal and fast remembering that can pass off or be consolidated through repetitions that establish durable changes in it [3]. The simplified model of associative neuron described in this paper uses a single weight value for expressing the synaptic transmission. The temporal and stable changes of synaptic transmission are necessary for exploiting diversified memory functions of as-systems. The synaptic weight for the simplified model of as-neuron can be determined after the number η_{SN} of activation of the SN neuron for a given training sequence set $\mathbb{S}$ after the following formula (2):

$$w_{SN,\widehat{SN}} = \frac{2\delta_{SN,\widehat{SN}}}{\eta_{SN} + \delta_{SN,\widehat{SN}}} \tag{2}$$

$$\delta_{SN,\widehat{SN}} = \sum_{\{\rightsquigarrow ACON_{\tau}:SN\rightsquigarrow\cdots\rightsquigarrow\widehat{SN}\in AAT\}} \frac{1}{\tau} \tag{3}$$

The time lapse τ depends on the grade of the associative contextual $SN \rightsquigarrow \widehat{SN}$ connection that does not activate the $\widehat{SN}$ neuron immediately after exciting its synapse but only takes a part in excitating this as-neuron. This neuron will be activated in the future only when the sufficient number of other input stimuli will be added to this excitation. This forms a special kind of spread combination that activates an as-neuron (1) and partially defines its semassel.

3 Associative Knowledge Representation

Artificial associative systems (aas-systems) are self-adapting plastic neural graphs, which elements actively react on repeatability and frequency of various spread over time combinations of input stimuli and their sequences [3]. The neurons of aas-systems are able to dynamically create and modify connections and their parameters to represent these combinations and their sequences after given training samples or training sequences. The diversity of their elements and connections allows them to cover an input data space according to the repeatability and various frequencies of their presentations. The neural representation of frequent combinations is not duplicated but automatically consolidated. New neurons try to represent only those frequent combinations and subcombinations that are already not represented. Such aggregations make aas-systems capable to generalize and be even creative. The aas-systems consist of associative neurons, sense receptors, and effectors that are embedded in interneuronal space.

All their elements work in time that is crucial for their inner associative processes and context consideration. The sense receptors transform various data into their inner representation. The neurons represent and associate various combinations of input data. The effectors affect surroundings, computer systems, or other associative systems. Their inner graph structure is self-created after some defined plasticity mechanisms. The associative rules link each represented object to many other objects. The appropriate weights (2), the sensitiveness of neurons, and their possible active reactions (1) enable to trigger sequences of associations upon activating neurons that represent various objects. The sequences of objects can represent whatever, e.g., various facts, rules, and even algorithms.

This paper describes the construction only on high-level representation of already created objects in order to demonstrate the generalization properties and creativity of aas-systems (Fig. 2f). We assume that semassels representing objects are already formed, established, and represented in as-neurons of the active neuroassociative knowledge graph (ANAKG). This graph is the subgraph of the aas-system that represents the sequences of objects taking into account their sequences and previous contexts. The ANAKG can be quickly constructed after training sequences and the information about repetitions of individual sequences as follows: For each training sequence $S \in \mathbb{S}$, its repetitions, and for each combination $c \in S$ in subsequent points of time check if this combination activates any existing as-neuron. If it does not then create a new as-neuron representing this combination and activate it. Next, connect the currently activated neuron to all recently activated neurons by the currently processed training sequence S if such connections do not yet exist. Compute or update the weights of these connections after formula (2) in accordance to the flow of time between activations of each pair of connected neurons (3). Each neuron counts up its activations η_{SN} for the trained set $\mathbb{S}$ and automatically updates weights of all outcoming connections (2). Each synapse (connection) updates its activation coefficient δ if presynaptic and postsynaptic neurons have been activated for the same processed training sequence according to the flow of time between their activations (3). If the computed or updated weights provide ambiguous or incorrect activations of the following neurons determined by the currently trained sequence, then the previously activated neurons create suppressive (inhibition) connections to the incorrectly activated neurons or to their excited synapses.

The biological as-systems form knowledge in nervous systems during their lifetime. In computer science, we need to form knowledge much faster and usually for given data that are stored in databases. The artificial as-systems are equipped with ability to quickly adapt to any set of training sequences of data combinations, e.g., sentences consisting of words. The as-neurons of such systems represent such combinations and appropriately link them together to reproduce their real sequences or subsequences. Such links are represented by weighted neural connections. The weights reflect the frequencies of each of the two subsequent combinations

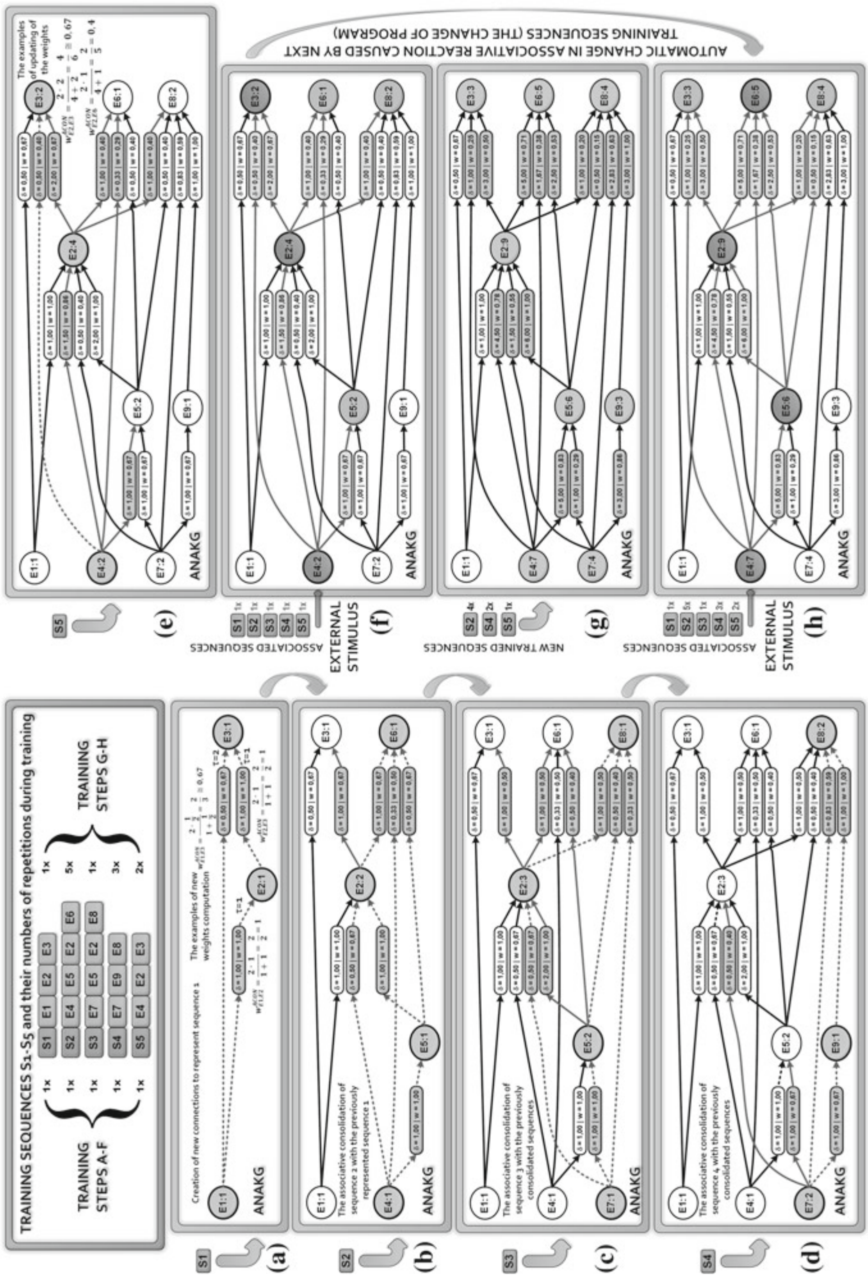


Fig. 2 The construction of the ANAKG that is a part of an aas-system for the given set $\mathbb{S}$ consisting of five training sequences that frequencies of repetition differ

represented in the connected neurons and the flow of time between presentations of these connected combinations (2). Moreover, there are created some extra connections coming from more previous combinations in order to make the reaction of each as-neuron conditional on a few previous activations of the neurons representing the context of the trained sequence, which is especially important since when the same combinations in various training sequences are represented by the same neurons, so associative activation tracks of neurons reflecting following combinations of training sequences cross many times in the aas-system (Fig. 2). Yet, not all existing contextual connections are sufficient to retrace each training sequence. This apparent weakness is a great advantage of aas-systems because it allows them for creativity and activation of new sequences. The insignificant or imprecise contexts recall the most frequent and strong activations or their sequences that have been learned. When the context is known and very precise, there is no place for creativity, and the brain recalls the learned associations that suit this context. If the context is partial or differs from the trained contexts, then the recalled associations can be new and creative, thanks to the consolidation of the same and similar combinations in the same as-neurons and their connections to other as-neurons. As a result, less-frequent sequences can be achieved only if the external context will be more precise.

The knowledge represented in aas-systems and ANAKG subgraphs can be easily developed and extended. All future associations are dependent on all presented samples, their sequences, and their frequencies of presentations. The repetition of previously trained sequences also can change the way of future associations and answers of the aas-system. It means that the inner associative programs of data processing can automatically change according to the repeated or new training sequences. In conclusion, the aas-systems are reprogramming themselves after training input data combinations and their frequencies alike a human mind.

4 Final Remarks and Conclusions

The presented ANAKG and artificial neural associative systems are able to form and develop knowledge in an associative way using the as-neurons for any set of training sequences. These emergent cognitive architectures can actively connect representations of aggregated data groups. The construction, training process, and recalling of associations are fast and contextual. The associations of similarity and the associations of sequence can be triggered using various internal or external activations of as-neurons or sense receptors. The accuracy and novelty of the used context for excitation of as-neurons determine if the achieved reaction (answer) will be the one of the previously trained or will be creative. The creative answers are usually produced for new contexts, i.e., new time-spread combinations of input stimuli that activate as-neurons in some new ways.

Concluding, knowledge can be automatically formed in the artificial associative systems as a result of weighted consolidation of various facts and rules in accordance to their similarities and subsequent occurrences. Such knowledge can be contextually evaluated and various facts and rules can be retrieved by recalling them as the result of activations of the following as-neurons which represent various pieces of information. Creativity of artificial associative systems comes into being as a result of a high-level generalization on the sequence level.

References

1. Alberts, B., Bray, D., Hopkin, K., Johnson, A., Lewis, J., Raff, M., Roberts, K., Walter, P.: Basics of Cell Biology. PWN, Warsaw (2007)
2. Duch, W.: Creativity and the Brain, In: A Handbook of Creativity for Teachers. Ed. Ai-Girl Tan, pp. 507–530. World Scientific Publishing, Singapore (2007)
3. Horzyk, A.: Artificial Associative Systems and Associative Artificial Intelligence. pp. 1–276. EXIT, Warsaw (2013)
4. Horzyk, A.: Information Freedom and Associative Artificial Intelligence. LNAI, vol. 7267, pp. 81–89. Springer, Berlin (2012)
5. Horzyk, A.: How Does Human-Like Knowledge Come into Being in Artificial Associative Systems, pp. 189–200. Proceedings of International Conference on Knowledge, Information and Creativity Support Systems (2013)
6. Horzyk, A.: How Does Generalization and Creativity Come into Being in Neural Associative Systems and How Does It Form Human-Like Knowledge, pp. 238–257. Neurocomputing (2014)
7. Kalat, J.W.: Biological Grounds of Psychology. PWN, Warsaw (2006)
8. Kossut, M.: The Mechanisms of Brain Plasticity. PWN, Warsaw (1993)
9. Lachwa, A.: Fuzzy World Harvest, Numbers, Relationships, Facts, Rules, and Decisions. EXIT, Warsaw (2001)
10. Larose, D.T.: Discovering Knowledge From Data. Introduction to Data Mining. PWN, Warsaw (2006)
11. Ligeza, A., Kluza, K., Nalepa, G.J., Potempa, T.: Artificial intelligence for knowledge management with BPMN and rules. Proc. AI4KM (2012)
12. Longstaff, A.: Neurobiology. PWN, Warsaw (2006)
13. Maass, W.: Networks of spiking neurons: the third generation of neural network models. Neural Netw. **10**, 1659–1671 (1997)
14. Papadimitriou, C.H.: Computational Complexity. WNT, Warsaw (2002)
15. Schmidhuber, J.: Developmental robotics, optimal artificial curiosity, creativity, music, and the fine arts. Connect. Sci. **18**(2), 173–187 (2006)
16. Skulimowski, A.M.J.: Freedom of Choice and Creativity in Multicriteria Decision Making. LNAI, vol. 6746, pp. 190–203. Springer, Berlin (2011)
17. Skulimowski, A.M.J.: Anticipatory network models of multicriteria decision-making processes. Int. J. Syst. Sci. **45**(1), 39–59 (2014)
18. Skulimowski, A.M.J.: Hybrid anticipatory networks. In: Proceedings of 11th International Conference ICAISC 2012, LNAI, vol. 7268, pp. 706–715. Springer, Berlin (2012)
19. Tadeusiewicz, R., Korbicz, J., Rutkowski, L., Duch, W. (eds.): Neural networks in biomedical engineering. In: Monograph: Biomedical Engineering—Basics and Applications, vol. 9. Exit, Warsaw (2013)
20. Tadeusiewicz, R.: New trends in neurocybernetics. Comput. Methods Mater. Sci. **10**(1), 1–7 (2010)
21. Tadeusiewicz, R., Figura, I.: Phenomenon of tolerance to damage in artificial neural networks. Comput. Methods Mater. Sci. **11**(4), 501–513 (2011)
22. Tadeusiewicz, R.: Neural Networks as Computational Tool with Interesting Psychological Applications, Chapter in book: Computer Science and Psychology in Information Society, pp. 49–101. AGH Printing House, Krakow (2011)

23. Thaler, S.L.: The Creativity Machine Paradigm, In: Carayannis, E.G. (ed.) Encyclopedia of Creativity, Invention, Innovation, and Entrepreneurship. Springer (2013)
24. Wulfram, G., Werner, M.K.: Spiking Neuron Models. Single Neurons, Populations, Plasticity. Cambridge University Press, Cambridge (2002)

Improvement of a Web Browser Game Through the Knowledge Extracted from Player Behavior

João Alves, José Neves, Sascha Lange and Martin Riedmiller

Abstract In this article, we describe how a web browser game can be improved through the knowledge obtained from the analysis of the behavior of its players. The analysis of player behavior is a technique that has been used with success in traditional computer games. These kind of analyses have been proven to help the developers in creating games which provide a more engaging and enjoyable experience. Currently, there is an interest in trying to replicate this success in a less-conventional genre of games, normally called browser games. The defining characteristic of browser games is the fact that they are computer games which are played directly on the web browser. Due to the increased ease of Internet access and the growth of the smartphone market, this game genre has a promising future. Browser games regarding the area of game mining have an advantage over traditional computer games in that their characteristics player behavior is relatively easy to collect.

Keywords Knowledge extraction · Data mining · Machine learning · Browser games · MMOG

J. Alves (✉) · J. Neves
Department of Informatics, University of Minho, Braga, Portugal
e-mail: joao.mpereira.alves@gmail.com

J. Neves
e-mail: jneves@di.uminho.pt

S. Lange
5DLab, Freiburg, Germany
e-mail: sascha@5dlab.com

M. Riedmiller
Department of Computer Science, University of Freiburg, Freiburg, Germany
e-mail: riedmiller@informatik.uni-freiburg.de

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_5

1 Introduction

Browser games are computer games that have the unique characteristic of being played directly on the web browser without the need of additional software fittings. The web browser is then responsible to provide the player interface and the access to the game world.

Typically, games in this genre offer a multiplayer environment where users can interact with each other. Attending to the way this genre works, the current game status of the players needs to be stored by the game provider. This fact makes it easy to collect information about the players, since companies already need to store it in order for their games to work.

There is an increasing awareness in the game developers community about browser games with the rise of smartphones with web access [3] and the growing popularity of social platforms [9].

Traditionally, the business model for computer games involves selling a game package to the customer that enables him to enjoy the game. However, by definition, in browser games there is no package to be sold. To be able to gain popularity and to attract new players, due to fierce competition, these games are available free of charge. Because of this constraint, developers have to resort to other sources of revenue. Commonly these sources are advertising, which may be displayed easily within the player interface, or selling in-game features, which typically enhance the player's game experience [16].

In this paper, we will focus on the behaviorial analysis of the players registered in a browser game developed by 5DLab[1] called Wack-a-Doo[2] and the improvements made in result of this analysis. Wack-a-Doo is a strategy browser game which requires players to develop economical and research structures so that they can maintain an army able to resolve military conflicts with other players and to secure in-game points of strategic interest. Regarding this game, some work has already been published; however, the focus was more on enumerating important behaviorial characteristics and less on improvements that can be created taking into account the information obtained from those characteristics [1].

In browser games, the playing mechanics involve the player making a limited set of actions, for example moving armies or founding cities that have an associated time cost for completion. As a result, the player can leave the game and return only when the time associated to these actions is over or he can perform more in-game actions. Because of these playing mechanics, the typical user performs short playing sessions various times a day [10]. The mechanics of Wack-a-Doo, while having their own original flavor, follow also this generic formula.

[1] http://www.5dlab.com.

[2] http://www.wack-a-doo.com.

2 Related Works

The growth in popularity of free-to-play games like Wack-a-Doo has created a new research perspective on behavior analysis. Because the major revenue stream for the developers of these kind of games is selling in-game content through microtransactions, the analysis of the players preferences and their development is truly crucial for the game's financial success [16]. The games developed under this game model require a constant analysis of the player behavior in order to maintain their financial sustainability [12].

The dependency that games have on obtaining knowledge from the player behavior lead to the creation of many fairly recent companies that provide solutions which enable the game developers to perform all kinds of in-depth analysis of the data collected by the game. Some companies present these characteristics namely Honeytracks[3] or GameAnalytics.[4]

Unfortunately, the amount of studies publicly known about knowledge extraction in MMOGs is limited, mainly due to confidentiality issues. The available knowledge for free-to-play games is more comprehensive; however, it generally comes from articles or blog posts [6].

One of the techniques used for analyzing player data is calculating metrics that serve as key performance indicators (KPIs); examples of KPIs in MMOGs could be the session times, the churn rate or if applicable even tutorial completion information [7]. Because of the more mature state of the field of web analytics, there are some techniques that were adapted from this field and used in the context of MMOGs. Examples of this adaptation could be conversion rates analysis, user acquisition cost analysis, or cohort analysis [8].

There are also data mining techniques used on more traditional computer games that have objectives such as behavior prediction [14], classification of user behavior [2, 5], and retention modeling [8] that are interesting and can potentially provide very useful information to the MMOGs developers [6].

3 Motivation

The sources of revenue in Wack-a-Doo are the selling of extra features to the players and the selling of bonus that give a temporary game advantage to the players who buy them. There is then, of course, a huge incentive for the developers to create a game that incentives the consumption of those products. To improve their game experience it was decided to analyze the player's behavior.

The company behind Wack-a-Doo divides the players in categories according to a hierarchy of states. This classification is made in such a way that when a player advances to the next conversion state, he can never posteriorly be classified at a

[3] www.honeytracks.com.

[4] http://www.gameanalytics.com.

lower position in this hierarchy. From the lowest to the highest hierarchic level, the conversion states found in this study are *registered*, *logged in once*, *played 10 min*, *logged in 2 days*, *active player*, and *paying player*.

Aside from the conversion state, there are other indicators that the Wack-a-Doo developers suggest as important, such as the time played or of course the very common churn ratio. Our objective in this study is to analyze the player behavior and extract knowledge that can help developers improve or understand these indicators. The indicators used in this study will be primarily the time played and the conversion state.

There could be multiple ways to make the analysis of the player behavior, the approach chosen for this study was to extract as much knowledge using as source only the first playing days of each player. As such we decided to use the information from the first 3 days of each player and make a short-term prediction. The day used for the prediction value was the 10th day.

Given the long-term characteristic of the games in this genre [16], the study focus on only some days can be confusing. The fact is that the analysis is not only on some days of the game history; what happens is that we used not all the days of the game history but only the first 3 days of the player history. This short period is relevant due to three factors. The former relies in the fact that it is in the first days of playing that we can find the information about all the users experimenting the game for the first time. The second point is that the first days of each player are very important in defining their behavior for the rest of the game; for example, in Wack-a-Doo after the 10th day and not counting inactive users 81 % of the players maintain their conversion status. The latest is a consequence of the high short-term churn ratio that is verified on Wack-a-Doo.

This short-term analysis can also be very useful to help the companies quickly decide on the effectiveness of a marketing campaign. For example, they can study the players that registered from a certain marketing campaign in their first few days, and predict whether or not there is a good percentage of them that are going to bring any added value to the game.

4 Data Collection and Preprocessing

For this study to be relevant, the data needs be extracted automatically from the deployed Wack-a-Doo database. As such, the first task was to analyze the database and decide which of the stored data could be used to describe the player's behavior. Fortunately, the developers had a very-well-designed database and stored various player attributes useful for this study.

The approach taken while selecting the attributes to collect from the database was a comprehensive one. The final list contained 33 features for each player that varied from the gameplay related such as the number of armies or the number of settlements founded, to the more social oriented such as number of messages received/sent or number of likes received/sent, to the obligatory more financial oriented such as amount of money spent.

One of the characteristics of the Wack-a-Doo database is that it does not have a time-related attribute associated with its data. In practise, only the current state of the player is registered and there is no information about when certain changes in the state of the player occurred. Because of this it would not be possible to study the evolution of the player using Wack-a-Doo's database alone.

To surpass this shortcoming we developed a system for collection and preprocessing of player status that we call SysCOPPE. This system is capable of giving us the necessary knowledge about player's evolution. With SysCOPPE we are easily able to collect all of the selected attributes from the Wack-a-Doo database at regular intervals. For the reasons of this study, we decided to opt to use an interval of 24 h between each data collection (Fig. 1).

SysCOPPE runs every day and stores its daily data collection in a text comma-separated values (csv) file. After that, the data in the csv file is preprocessed and then inserted in the SysCOPPE database. The data used for this study was collected during a period of 108 days with the first collection data being 23 of March the first data collection date.

During the preprocessing of the csv's SysCOPPE performs data filtering and cleansing tasks. For example, sometimes players use stolen credit cards to make purchases and when that happens their registered money gross will revert back but the conversion state will not. In this case SysCOPPE will update the conversion state to the value it was before. Another example more related to the developer Wack-a-Doo implementation could be the negative values for time played when the player has registered but never logged in, when this happens SysCOOPE changes the value to zero.

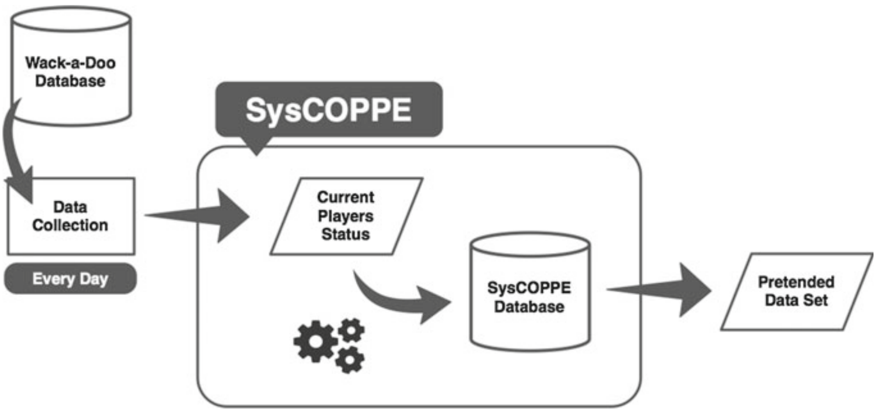


Fig. 1 SysCOPPE

Another of the features of SysCOPPE is the ability to generate datasets according to user specified requirements. The user can specify e.g., the time interval, the filtering inactive users or of course the target feature of his pretended dataset.

5 Methodology

According to the definition, a time series is a series of timely sequential observations $X_i(t)$; $[i = 2, \ldots, N; t = 1, \ldots, T]$, where T refers to each time in point and i refers to the number of variables in the time series. In our case, the number of observation by each point in time is equal to the number of players and the variables registered are the features that we record for each player. Because our number of observations per point in time is bigger than one, we can say that the data registered on the SysCOPPE database allows the construction of a multivariate time series (MTS).

However, when creating this time series we need to be careful when choosing what values do we want to use for our time axis. We could make a time series using the collection data as the time value, but from an analysis point of view this would be highly debatable. What makes sense is to position the time axis in relation to the days since the player registered in the game; with this when we select the time position number 1, we will only see players in identical game development stages. With the necessity cut our time series to only three points in time, the traditional MTS mining techniques are inadequate [15]. Because of this constraint imposed by our objectives and to avoid confusions, we will stop treating our data as a time series.

The first step to prepare for our experiments was to create the two datasets with the 33 collected features during the first 3 played days with one having as target the conversion state and the other the time played. Because the datasets comprise of data collected during three days plus the prediction target value on the tenth, the number of features for each recorded player was not 33 plus the prediction target but 33 for each day (e.g., for each player we have recorded the number of logins on day 1, on day 2, and on day 3. The number of players used for this study were 4212.

The number of attributes collected are already considerable; however, we were not totally satisfied because our datasets consisted only of absolute values that did not revel directly the player evolution. To overcome this problem, we decided to do some data transformation adding new attributes.

One of the measured features where we made some data transformation was the number of armies (where we calculated the difference of the number of armies between each of the 3 days). This gives a more direct measure of how fast the player is creating armies. Another example could be the information that we store each day if the player has joined or not an alliance under the form of a Boolean and we added a new attribute saying specifically in which day he joined an alliance. We also added some attributes that are not game related but could prove to be relevant such as the day of the week when he registered. With this new approach we added 59 new attributes, having this new dataset version a total of 158 attributes. This addition should make the task of the prediction algorithms easier.

During our experiments we used the Repeated Incremental Pruning to Produce Error Reduction algorithm [4] for rule learning and the C4.5 algorithm [13] for decision tree generation. This particular decision to use these kinds of algorithms is justified by the fact that we do not want to create the best prediction model possible but to extract knowledge from the data. Decision tree and rules are one of the most direct ways to accomplish this objective [11].

6 Experiments and Results

6.1 Conversion State

For the analysis focused on the conversion state, we used the RIPPER and the C4.5 algorithms. For this classification problem, the decision tree model had a general accuracy of 90.2 %, really close to the 91.0 % general accuracy obtained with the model created by RIPPER (Table 1).

The confusion matrix associated to the model created by the RIPPER algorithm shows that there is some difficulty distinguishing between the *active player* and the *logged 2 days* conversion states. Actually this observation also holds when analyzing the model resulting from the C4.5. According to the RIPPER model, the precision percentages of the *active player* conversion state and the *logged 2 days* states are, respectively, 77.9 and 89.25 %, being the former the worst class precision of both models. The recall for these two states is, in the same order, 82.25 and 55.77 %.

These results show that there is a very big difficulty predicting the *logged 2 days* state, and the class most affected by this is the *active player* state. In fact this is one of the reasons why we can say that the investment in adding new attributes to the dataset paid of. With these new attributes, the confusion between those two classes decreased around 20 % in both of the algorithms used. It was interesting to verify also that the attributes that made most difference in decreasing this confusion were the ones directly measuring player evolution, e.g., the difference between the time played in one day and the time played in the next one.

Table 1 Conversion state classification RIPPER confusion matrix

Pred. →	Ten minutes	Logged once	Active	Logged 2 days	Registered	Paying
Ten minutes	1317	0	4	0	0	0
Logged once	0	911	0	0	0	0
Active	14	1	292	46	0	2
Logged 2 days	174	26	72	382	31	0
Registered	1	0	0	0	896	0
Paying	0	0	7	0	0	36

This difficulty to distinguish between active players and players who logged in twice in a row can probably be explained by the nature of our analysis. When using only the information about the first 3 days the difference of a player who has logged in twice in a row and an active player should not be very noticeable. However the active players are crucial to the prosperity of Wack-a-Doo, this is why we dedicated some time to the analysis of how the models behaved in distinguishing these two classes.

From the learned rules, the one that is most helpful in distinguishing between the *active player* and the *logged in twice* states is the one that dictates that on day 3 the player played the game less then 24.3 min. This value is now useful for the developers as a goal to reach for each player and a performance indicator.

Another useful finding that helps distinguish between players from these two states is that if by day 2 the player has not raised an army, then he will never be an active player. What was concluded from this finding was that the military aspect of the game was an engaging factor for the players and as such it should be advertised. As a result from this finding, a slight change in the tutorial tasks of the game was introduced. In this new version of tutorial, the military mechanics are explained not only earlier but also more extensively.

It is also curious to observe again that the victory ratio of a player is positively related to the time he will be playing in the future. This means that the players who have a bigger win/loss ratio will end up playing the game more. This is a somewhat expected because it is only natural that a player finds a game more enjoyable when he is winning. However, boosting the victory ratio of the players is tricky because if one player wins another has to loose. The solution that was created for this ratio problem was the introduction of computer-controlled armies in the game. This new kind of armies not only helps boosting the players victory ratio but also, especially in the early game, provides more action for the players.

In the RIPPER model, there was only one rule that was used to classify the players belonging to the paying state. This rule can be interpreted as stating that a customer who was a paying customer in the first 3 days is classified as a paying customer in the 10th day. This is a trivial rule; however, what surprises is that it is able to classify almost 84 % of all the paying customers. This high percentage probably means that the game developers are doing a good job on encouraging the new players to spend money in their very first days.

From the models created it was evident that a higher number of logins per day indicates a likely higher retention of the player. It was that if a player only logged in once in the second day, then by the 10th day he would not login any more. To try to lure the players into the game and motivate them to come back, a temporary bonus feature was added. This feature consists of small “bubbles” of resources that appear around the player’s villages and when popped give the resources to the player. These bubbles are temporary and need to be popped within a certain time.

6.2 Time Played

Every time a player is playing the game, Wack-a-Doo keeps track of the time they spend and when their session is over it updates the amount of registered playtime. We can say that Wack-a-Doo keeps a cumulative counter of each players playtime. The metric unit used to register the time played is seconds. Because this data is stored in a continuous scale we had the need to discretize it.

When we tried to create bins of the same range size, we faced a problem because the number of users in each of the bins is very unbalanced. The most common bin has the range $(-\infty, 63385]$ and contains 98.5 % of players. This happens for two reasons the first is that it is expected for the cluster with low values to contain more players because of the users who register and never play or just try the game for one unique short session. The second is that there are some few players who spend a big amount of time playing, for example there are 18 players that are distributed between 126769 and 316923 s of playtime. And because of this distribution of players the range size for the bins needs to be big in order to each bin to have at least one player and to accommodate all players. Because of all this, the upper boundary of the first bin is really high which again contributes to unbalance of the player distribution across the bins.

With this last failure to apply an automatic discretization algorithm, we learned that it would make more sense for us to create the bins according to our analysis of the Wack-a-Doo data. After our analysis, we propose five profiles of players according to time playing patterns found in the Wack-a-Doo data. We should note that while the specific times used in the ranges of each profile are very specific to the Wack-a-Doo game, the theoretical notion behind each of these profiles is probably useful in every game of this genre (Tables 2 and 3).

The frequency of each of the classes still varies greatly from a minimum of 68 players to a maximum of 1773; however, the class imbalance is not as huge as the one observed in the previous discretization attempt. But more important than that, the clusters created make sense from a game point of view and should allow us to extract much more relevant knowledge from the Wack-a-Doo data.

After this preprocessing of the dataset, we used the rule learning algorithm RIPPER and the C4.5 decision tree generator algorithm. Both of these algorithms

Table 2 Profiles for players according to their time played

Name	Time range	Frequency
Not playing	[0, 1]	552
Experimenting players	(1, 100]	330
Casual players	(100, 10000]	1489
Interested players	(10000, 60000]	1773
Hardcore players	$(60000, +\infty)$	68

Table 3 Time classification C4.5 confusion matrix

Pred. →	Not playing	Experimenting	Casual	Interested	Hardcore
Not playing	552	0	0	0	0
Experimenting	8	319	2	1	0
Casual	3	3	1482	1	0
Interested	5	2	59	1687	20
Hardcore	0	0	0	29	39

performed very well with the decision tree having a general accuracy with 96.84 % and the rules a general accuracy of 95.75 %.

Because of the slight imbalance of our classification, general accuracy can be a very misleading measure; however, after studying both of the confusion matrices, we can conclude that the models created are relevant. In both of the created models, the worst precision results are obtained for the casual and hardcore players classes. In the particular case of the decision tree, the class precision of casual players is 96.05 % and of hardcore players 66.10 % with a respective recall of 99.53 % and 57.35 %. With this analysis, we can also conclude that this model does not perform well when differentiating between hardcore players and interested players when they are hardcore players.

The *completed tutorial* attribute in the decision tree model is one of the major factors distinguishing between the interested and hardcore players. This means that a more appealing game tutorial is very likely to bring more players to classes related to higher playtime. This led to a bigger focus on the development of a better tutorial and resulted in a rework of the tutorial design and functioning. Among the new tutorial features are an automatic pop-up system that details every assignment with pictures and themed graphics or an helping arrow that guides the players throughout the assignments. This helping arrow at first points to every location where the player needs to click but as the tutorial is getting to an end it starts to get less common and only indicating how to access recently introduced game features. Another change was the inclusion of relatively advanced game features in the tutorial that were previously left out.

Other important factor distinguishing between the interested and the hardcore players is the already mentioned success in battle. For the players who played more then 355 min, the ones who had a number of defeats bigger than zero in the first day are likely to be interested players and the ones who had more victories than defeats are likely to be classified as hardcore players. The solution created for this was already mentioned in the previous section where an artificial victory ratio boosting mechanism is discussed.

Example of the positive effect of the social features

```
(diff_time_d3d1 >= 10302) and (time_played_day1 >= 6680)
and (overlal_score_day3 >= 315) and (diff_ovrscore_d2d1 >= 75)
and (day_join_alliance <= 1) => result=hardcore players
```

(Example rule created with the RIPPER algorithm)

In the above example, *diff_time_d3d1* and *diff_ovrscore_d2d1* represent respectively the difference of the cumulative values of time played and overall score between days 3 and 2 and between days 2 and 1. If the player has joined an alliance the *day_join_alliance* attribute has a numerical value indicating in which of the 3 days that happened. This particular rule shows how the fact that a player has joined an alliance on day 1 positively influences his time spent on the game.

The creation of these two models led to the discovery that the social aspect of the game has a positive effect on the time played. The players who have sent messages or likes to other players or joined an alliance are always classified as one of the classes who spend more time playing. With this knowledge some effort was made in order to not only make the social part of the game more appealing but also to make the game itself more social. Some of the features that resulted from this effort are the addition of more channels for the in-game chat system and the introduction of better bonus for the players who brought their friends to the game.

It was added to Wack-a-Doo an interesting characteristic that makes it a more social game. What was changed was that the game when deciding the positioning of the player in the Wack-a-Doo map will take into account the real geographic location of the players. In practice, this means that players from the neighboring cities will be located in close proximity to each other. This change increases the probability that the player's motivation to play will come from real-life social factors. An example of such factors could be a rivalry between two cities that motivates the development of a war inside the game.

7 Conclusions

The future of browser games is predicted to be very promising with the rise of smartphones with web access and the growing popularity of social platforms. This creates an increased awareness in the game developers community, which in turns raises the competition in the browser games market. The usage of data mining techniques to extract knowledge about their game can be the deciding factor to achieve financial success.

In this paper it was described SysCOPPE, a system that was created to collect and preprocess data from a browser game. This type of solution is very versatile and can be easily adapted to a browser game created by a different developer. Future work on SysCOPPE should pass by adding the capability to make an automatic analysis of predefined features every time the SysCOPPE database is updated.

It was presented a case study using the data from a browser game currently in the market. In the experiments section it was suggested five profiles according to playtime patterns that were identified in the Wack-a-Doo data. These profiles are a contribution that may help classify players from other games of this genre.

This article focused on making very short-term predictions of features that the developers considered relevant. It is expected that this article will make a contribution to comprehend which characteristics of the player's behavior inside the game are important to be taken in consideration, and which parts of the game should be object of a more careful elaboration.

An example of an important player's characteristic with impact on the players behavior is the *victory ratio*. Another example could be the discovery that social-related game features, such as alliances or a like system, have also a positive effect inside the game.

There was also an indication of some changes or features in Wack-a-Doo that focused each of these important player characteristics. These improvements to the game can also be reproduced or at least used as guidelines by other companies that work on the web browser game market.

5DLab has recently released a mobile application of Wack-a-Doo for smartphones. This version of the game is not focused on this study. However, a study of the differences between the mobile players and the web browser players could yield interesting results.

Although it has been concluded that there was useful knowledge extracted using the current approach based only on the player behavior in their first days, we should note that to identify complex behavior patterns it is necessary to use longer time spans.

References

1. Alves, J., Neves, J., Lange, S., Riedmiller, M.: Knowledge extraction from the behaviour of players in a web browser game. In: Proceedings of the International Conference on Knowledge, Information and Creativity Support Systems, pp. 519–530. Progress & Business Publishers, Krakow, Poland (2013)
2. Anagnostou, K., Maragoudakis, M.: Data mining for player modeling in videogames. In: 2009 13th Panhellenic Conference on Informatics, pp. 30–34 (2009)
3. Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 20102015 (1 February 2011) by Cisco Systems Inc, http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white_paper_c11-520862.html, last access September 2014
4. Cohen, W.: Fast effective rule induction. In: Proceedings of the Twelfth International Conference on Machine Learning, pp. 115–123 (1995)
5. Drachen, A., Sifa, R., Bauckhage, C., Thurau, C.: Guns, swords and data: clustering of player behavior in computer games in the wild. In: Computational Intelligence and Games, 2012 IEEE Conference on Computational Intelligence and Games, pp. 163–170 (2012)
6. Drachen, A., Thurau, C., Togelius, J., Yannakakis, N., Bauckhage, C.: Game Analytics, Game Data Mining, pp. 205–253. Springer, London (2013)
7. Fields, T.V.: Game Analytics, Game Industry Metrics Terminology and Analytics Case Study, pp. 53–71. Springer, London (2013)
8. Fields, T., Cotton, B.: Social Game Design: Monetization Methods and Mechanics. Morgan Kauffman Publishers, Waltham (2012)
9. Kirman, B., Lawson, S., Linehan, C.: Gaming on and off the social graph: the social structure of facebook games. In: 2009 International Conference on Computational Science and Engineering, pp. 627–632 (2009)

10. Klimmt, C., Schmid, H., Orthmann, J.: Exploring the enjoyment of playing browser games. Cyberpsychology Behav. **12**, 231–234 (2009)
11. Langley, P., Simon, H.A.: Applications of machine learning and rule induction. Commun. ACM **38**, 54–64 (1995)
12. Mellon, L.: Applying Metrics Driven Development to MMO Costs and Risks. Versant Corporation (2009)
13. Ross, Q.: C4.5: Programs for Machine Learning, vol. 1. Morgan Kaufmann Publishers, California (1993)
14. Shim, K., Srivastava, J.: Sequence alignment based analysis of player behavior in massively multiplayer online role-playing games (MMORPGs). In: Data Mining Workshops ICDMW 2010 IEEE International Conference, pp. 997–1004 (2010)
15. Swift, S., Kok, J., Liu, X.: Learning short multivariate time series models through evolutionary and sparse matrix computation. Nat. Comput. **5**, 387–426 (2009)
16. Vanhatupa, J.: Business model of long-term browser-based games—Income without game packages. In: 2011 7th International Conference on Next Generation Web Services Practices (NWeSP), pp. 369–372 (2011)

Using Extracted Symptom-Treatment Relation from Texts to Construct Problem-Solving Map

Chaveevan Pechsiri, Onuma Moolwat and Rapepun Piriyakul

Abstract This paper aims to extract the relation between the disease symptoms and the treatments (called the symptom-treatment relation), from hospital-web-board documents to construct the problem-solving map which benefits inexpert people to solve their health problems in preliminary. Both symptoms and treatments expressed on documents are based on several EDUs (elementary discourse units). Our research contains three problems: first, how to identify a symptom-concept-EDU and a treatment-concept EDU. Second, how to determine a symptom-concept-EDU boundary and a treatment-concept-EDU boundary. Third, how to determine the symptom-treatment relation from documents. Therefore, we apply a word co-occurrence to identify a disease-symptom-concept/treatment-concept EDU and Naïve Bayes to determine a disease-symptom-concept boundary and a treatment-concept boundary. We propose using k-mean and Naïve Bayes to determine the symptom-treatment relation from documents with two feature sets, a symptom-concept-EDU group and a treatment-concept-EDU group. Finally, the research achieves 87.5 % precision and 75.4 % recall of the symptom-treatment relation extraction along with the problem-solving map construction.

Keywords Word order pair · Elementary discourse unit · Symptom-treatment relation

C. Pechsiri (✉) · O. Moolwat
Department of Information Technology, Dhurakij Pundit University, Bangkok, Thailand
e-mail: itdpu@hotmail.com

O. Moolwat
e-mail: moolwat@hotmail.com

R. Piriyakul
Department of Computer Science, Ramkhamhaeng University, Bangkok, Thailand
e-mail: rapepunnight@yahoo.com

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_6

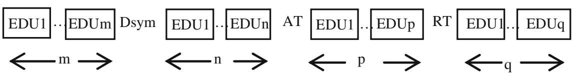
1 Introduction

The objective of this research is to develop a system of automatically extracting relation between the group of disease's symptoms and the treatment/treatment procedure (called the symptom-treatment relation) from the medical-care-consulting documents on the hospital's web-board of the nongovernmental organization (NGO) website edited by patients and professional medical practitioners. The extracted symptom-treatment relation is also conducted for the construction of problem-solving map (PSM), which is a map representing how to solve problems, especially disease treatments. PSM benefits general people to understand how to solve their health problems in the preliminary stage. Each medical-care-consulting document contains the disease symptoms and the treatments are expressed in the form of EDUs (elementary discourse units), which is a simple sentence/clause defined by [3]. Each EDU is expressed by the following linguistic expression:

EDU → NP1 VP
VP → verb NP2 | verb adv
verb → verb_{weak}-noun1 | verb_{weak}-noun2 |verb_{strong}
NP1 → pronoun | noun1 | noun1 AdjPhrase1
NP2 → noun2 | noun2 AdjPhrase2
noun1 → ' ', 'ผู้ป่วย/patient', 'อวัยวะ/human organ', 'แผล/ scar',…
noun2 → ' ', 'อวัยวะ/human organ', 'อาการ/symptom', 'สี../..color', 'ยา/ medicine',…
verb_{weak} → 'เป็นแผล/have scar', 'มีอาการ/have symptom',
verb_{strong} → 'รู้สึกปวด/pain', 'ใช้/apply', 'ทา/apply', …

where NP1 and NP2 are noun phrases, VP is a verb phrase, adv is an adverb, and AdjPhrase1 and AdjPhrase2 are adjective phrases.

Moreover, there are two kinds of treatments existing on the web-board documents; the actual treatment notified by patients/users from their experiences and the recommended treatment edited by professional medical practitioners. Thus, each medical-care-consulting document contains several EDUs of the disease-symptom-concepts along with the actual-treatment-concept EDUs and the recommended-treatment-concept EDUs as shown in the following form:



where

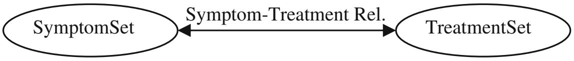
- Dsym, AT, and RT are a group of disease-symptom-concept EDUs, a group of actual-treatment-concept EDUs, and a group of recommended-treatment-concept EDUs, respectively, as follows:
  $\text{Dsym} = (\text{EDU}_{sym-1}\text{EDU}_{sym-2} \ldots \text{EDU}_{sym-a})$ where a is an integer number and is > 0,

AT = (EDU_{at-1} EDU_{at-2} … EDU_{at-b}) where b is an integer number and is ≥ 0,
RT = (EDU_{rt-1} EDU_{rt-2} … EDU_{rt-c}) where c is an integer number and is ≥ 0

- m, n, p, and q are the number of EDUs and are ≥ 0
- Therefore, the symptom-treatment relation can be expressed as follows:

$$\text{Dsym} \rightarrow \text{AT}$$
$$\text{Dsym} \rightarrow \text{RT}$$

From Fig. 1, Dsym is EDU1-EDU3, AT is EDU5, and RT is EDU8-EDU10. Thus, the extracted symptom-treatment relations (as shown in the following) from medical-care-consulting documents with several problem-topic names, e.g., dis-



eases, are collected for constructing the general PSM representation (see Fig. 5). where SymptomSet = {$Dsym_1$, $Dsym_2$, …, $Dsym_n$}, TreatmentSet = At ∪ RT, and $Dsym_1 \neq Dsym_2 \neq \ldots \neq Dsym_n$.

There are several techniques [1, 8, 9] being used to extract the symptom-treatment relation or the disease treatment relation from texts (see Sect. 2). However, Thai documents have several specific characteristics, such as zero anaphora or the implicit noun phrase, without word and sentence delimiters, etc. All of these characteristics are involved in three main problems of extracting the symptom-treatment relation

Problem-Topic: Stomachache

EDU1(symptom):*[ผู้ป่วย]ปวดท้องอย่างหนัก*; *[ผู้ป่วย]/[A patient] ปวดท้อง/has a stomachache อย่างหนัก/heavily*
([A patient] has a stomachache heavily.)
EDU2 (symptom): *[ผู้ป่วย]มีแก๊สในกระเพาะมาก*; *[ผู้ป่วย]/[The patient] มี/has แก๊ส/gas ในกระเพาะ/ inside stomach มาก* /a lots
([The patient] has lots of gas in the stomach)
EDU3 (symptom): *อาการมักจะเป็นหลังทานข้าวเย็นและตอนกลางคืน*; *อาการ/symptom มักจะเป็น/mostly occurs หลัง/after ทานข้าวเย็น/having dinner และ/and ตอนกลางคืน/at night*
(The symptom mostly occurs after having dinner and at night)
EDU4: *[ผู้ป่วย]สงสัยเป็นโรคกระเพาะ*; *[ผู้ป่วย]/[The patient] สงสัย/doubts เป็นโรคกระเพาะ/to get a gastropathy*
([The patient] doubts to get a gastropathy)
EDU5 (treatment): *[ผู้ป่วย]กินยาลดกรดเพื่อแก้ปวดท้อง*; *[ผู้ป่วย]/[The patient] กินยาลดกรด/takes an antacid เพื่อแก้ปวดท้อง /to solve the stomach ache*
([The patient] takes an antacid to solve the stomach ache)
EDU6: *แต่ก็ไม่หายปวด*; *แต่/But ก็ไม่หายปวด /it cannot work.* **(But it cannot work)**

Physician Suggestion

EDU7 *ไปหาหมอหรือยัง /Have you seen the doctor?*
EDU8 (recommendation): *ถ้า[ผู้ป่วย]เป็นโรคกระเพาะ*; *ถ้า /If [the patient] เป็นโรคกระเพาะ/ get a gastropathy*
(If [the user] gets a gastropathy)
EDU9 (recommendation): *[ผู้ป่วย]ก็อาจต้องกินยาลดการหลั่งกรดในกระเพาะอาหาร*; *[ผู้ป่วย]/[The patient] ก็อาจต้องกินยา/may take a medicine ลด/to reduce การหลั่งกรดในกระเพาะอาหาร/ gastric acid secretion*
([the user] may take a medicine to reduce the gastric acid secretion)
EDU10 (recommendation): *หลีกเลี่ยงอาหารที่ทำให้เกิดแก๊สในกระเพาะ*; *หลีกเลี่ยง/avoid อาหาร/food ที่ทำให้เกิด/ causing แก๊สในกระเพาะ/stomach gassy* **(Avoid food causing stomach gassy.)**

Fig. 1 Shows an example of a web-board document containing the symptom-treatment relations (where the […] symbol means ellipsis.)

from the NGO web-board documents (see Sect. 3): the first problem is how to identify the disease-symptom-concept EDU and the treatment-concept EDU. The second problem is how to identify the symptom-concept-EDU boundary as Dsym and the treatment-concept-EDU boundary as AT/RT. The third problem is how to determine the symptom-treatment relation from documents. From all of these problems, we need to develop a framework that combines the machine learning technique and the linguistic phenomena to learn the several EDU expressions of the symptom-treatment relations. Therefore, we apply learning relatedness value [6] of a word co-occurrence (called "Word-CO") to determine a symptom-concept EDU or a treatment-concept EDU. Word-CO in our research is the expression of two adjacent words as a word order pair (where the first word is a verb expression) existing in one EDU and having either a disease symptom-concept or a treatment concept. The Naïve Bayes classifier [7] is also applied to solve the disease-symptom-concept-EDU boundary and the treatment-concept-EDU boundary from the consecutive EDUs. We also propose using the Naïve Bayes classifier to determine the symptom-treatment relation from documents after clustering objects of posted problems (Dsym) on the web-board and clustering treatment features.

Our research is separated into five sections. In Sect. 2, the related work is summarized. Problems in extracting symptom-treatment relations from Thai documents are described in Sect. 3 and Sect. 4 shows our framework for extracting the symptom-treatment relation. In Sect. 5, we evaluate and conclude our proposed model.

2 Related Work

Several strategies [1, 8, 9] have been proposed to extract the disease treatment relation (or the symptom-treatment relation as in our research) from the textual data.

In 2005, Rosario [8] extracted the semantic relations from bioscience text. In general, the entities are often realized as noun phrases, the relationships often correspond to grammatical functional relations. For example:

> Therefore administration of TJ-135 may be useful in patients with severe acute hepatitis accompanying cholestasis or in those with autoimmune hepatitis.

where the disease *hepatitis* and the treatment *TJ-135* are entities and the semantic relation is: *hepatitis* is treated or cured by *TJ-135*. The goals of her work are to identify the semantic roles DIS (Disease) and TREAT (Treament), and to identify the semantic relation between DIS and TREAT from bioscience abstracts. She identified the entities (DIS and TREAT) by using MeSH and the relationships between the entities by using a neural network based on five graphical models with lexical, syntactic, and semantic features. Her results were of 79.6 % accuracy in the relation classification when the entities were hidden and 96.9 % when the entities were given.

Abacha and Zweigenbaum [1] extracted semantic relations between medical entities (as the treatment relations between a medical treatment and a problem, e.g., disease) by using the linguistic-based pattern to extract the relation from the selective MEDLINE articles.

Linguistic Pattern: … *E*1 … be effective for *E*2… | … *E*1 was found to reduce *E*2 …where *E*1, *E*2, or *Ei* is the medical entity (as well as UMLS concepts and semantic types) identified by MetaMap.

Their treatment relation extraction was based on a couple of medical entities or noun phrases occurring within a single sentence as shown in the following example:

> Fosfomycin (*E*1) and amoxicillin-clavulanate (*E*2) appear to be effective for cystitis (*E*3) caused by susceptible isolates

Finally, their results showed 75.72 % precision and 60.46 % recall.

In 2011, Song et al. [9] extracted the procedural knowledge from MEDLINE abstracts as shown in the following by using supporting vector machine (SVM) and comparing to conditional random field (CRF), along with natural language processing.

> …[In a total gastrectomy] (Target), [clamps are placed on the end of the esophagus and the end of the small intestine] (*P*1). [The stomach is removed] (*P*2) and [the esophagus is joined to the intestine] (*P*3)…

where *P*1, *P*2, and *P*3 are the solution procedures. They defined procedural knowledge as a combination of a target and a corresponding solution consisting of one or more related procedures/methods. SVM and CRF were utilized with four feature types: content feature (after word stemming and stop-word elimination) with a unigram and bi-grams in a target sentence, position feature, neighbor feature, and ontological feature to classify target. The other features, word feature, context feature, predicate-argument structure, and ontological feature, were utilized to classify procedures from several sentences. Their results are 0.7279 and 0.8369 precision of CRF and SVM, respectively, with 0.7326 and 0.7957 recall of CRF and SVM, respectively.

In most of the previous works, i.e., [1, 8], the treatment relation between the medical treatment and the problem (as a disease) occur within one sentence, whereas our symptom-treatment relation occurs within several sentences/EDUs on both the treatments and the problem. However, the work of [9] has several sentences of the treatment method but there is one sentence of problem as the target disease or symptom. Therefore, we propose using the Naïve Bayes classifier to learn the symptom-treatment relation with features from clustering objects of posted problems (Dsym) on the web-board and clustering treatment concepts from AT/RT.

3 Problems of Symptom-Treatment Relation Extraction

To extract the symptom-treatment relation, there are three problems that must be solved: how to identify a symptom-concept EDU and a treatment-concept EDU, how to determine a symptom-concept-EDU boundary and a treatment-concept-EDU boundary, and how to determine symptom-treatment relations from documents.

3.1 How to Identify Symptom-Concept EDU and Treatment Concept EDU

According to the corpus behavior study of the medical care domain, most of the symptom-concept EDUs and the treatment concept EDUs are expressed by verb phrase. For example:

Symptom Concept

(a) EDU: “ผู้ป่วยรู้สึกเวียนศีรษะ”; “ผู้ป่วย/*A patient* รู้สึก/*feels* เวียนศีรษะ/*dizzy*”(A patient feels dizzy.)

(b) EDU: “ฉัน[มีอาการ]ปวดศีรษะ”; “ฉัน/*I* [มีอาการ/*have symptom*] ปวดศีรษะ/*headache*” (I have a headache symptom.) where […] means ellipsis

Treatment Concept

(c) EDU: “กินยาลดกรด”; “กิน/*consume* ยาลดกรด/*antacid*” (Take an antacid.)

However, some verb phrase expressions of the symptom-concepts are ambiguities. For example:

(e) EDU: “[คนไข้]ถ่ายยาก”; “[คนไข้/*patient*] ถ่าย/*defecate* ยาก/*difficultly*” ([A patient] defecates difficultly.)

(f) EDU1: “ห้องน้ำสกปรกมาก”; “ห้องน้ำ/toilet สกปรกมาก/is very dirty.” (A toilet is very dirty.)

EDU2: ฉันจึงถ่ายยาก”; “ฉัน/*I* จึง/*then* ถ่าย/*defecate* ยาก/*difficultly*” (Then, I defecate difficultly.)

From (e) and (f) examples, the verb phrase expression of the symptom-concept occurs only in (e) with the concept of “ท้องผูก/be constipated.”

This problem can be solved by learning the relatedness from two consecutive words of Word-CO with the symptom-concept or treatment concept. Where the first word of Word-CO is a verb expression, v_{co}, approaching to the symptom-concept or the treatment concept (where $v_{co} \in V_{co}$, $V_{co} = V_{co1} \cup V_{co2}$, V_{co1} is a set of verbs approaching the symptom concepts and V_{co2} is the treatment-verb concept set). The second word of Word-CO is a co-occurred word, w_{co} ($w_{co} \in W_{co}$; $W_{co} = W_{co1} \cup$

W_{co2}). W_{co1} and W_{co2} are co-occurred word sets inducing the v_{co1} w_{co1} co-occurrence and the v_{co2} w_{co2} co-occurrence to have the symptom concept and treatment concept, respectively, where $v_{co1} \in V_{co1}$, $w_{co1} \in W_{co1}$, $v_{co2} \in V_{co2}$, and $w_{co2} \in W_{co2}$. All concepts of V_{co1}, V_{co2}, W_{co1}, and W_{co2} from the annotated corpus are obtained from Wordnet and MeSH.

V_{co1} = {'ถ่าย/ *defecate*','เบ่ง/*push*', 'ปวดท้อง/*have an abdomen pain*','ปวด/*pain*', 'อึดอัด/*be uncomfortable*','รู้สึกไม่สบาย/*be uncomfortable*','มี[อาการ]/*have* [*symptom*]',}
V_{co2} = 'กิน/*consume*', 'ทา/*apply*', 'ใช้/*apply*' 'รักษา/*remedy*','บำรุง/*nourish*', 'ลด/*reduce*',…}
W_{co1} = {' ', 'ยาก/*difficultly*', 'ถ่าย/*stools*', 'เชื้อ/*germ*', 'เหลว/*liquid'*, 'ประจำเดือน/*period*', 'แน่นท้อง/*fullness*', 'ท้องเฟ้อ/ *flatulence*', 'ไข้/*fever*', …}
W_{co2} = {'ยา/*medicine*', 'อาหาร/*food*', 'อาหารเสริม/*supplement*',…}

3.2 How to Solve Symptom-Concept-EDU Boundary (Dsym) and Treatment-Concept-EDU Boundary (AT, RT)

According to Fig. 1, there is no clue (i.e. 'และ/and', 'หรือ/or',…) on both EDU3 to identify the disease-symptom boundary (EDU1-EDU3) and EDU10 to identify the treatment boundary (EDU8-EDU10). After the symptom-concept EDU and the treatment-concept EDU have been identified by using Word-CO from the previous step, we then solve the symptom-concept-EDU boundary and the treatment-concept-EDU boundary by applying Naïve Bayes to learn a Word-CO pair from a window size of two consecutive EDUs with one sliding EDU distance.

3.3 How to Determine Symptom-Treatment Relation

The relations between symptoms and treatments are varied among patients, environments, times, etc., even though they have the same disease. For example:

(a) EDU1symtom:	[ผู้ป่วย]ปวดท้อง/*have a stomach ache* อย่างหนัก/*heavily* **([A patient] has a stomachache heavily**.)
EDU2symtom:	[ผู้ป่วย] มี/*has* แก๊ส/*gas* ในกระเพาะ/ *inside stomach* มาก/a lots **([The patient] has lots of gas in the stomach)**
EDU3 treatment:	[ผู้ป่วย] กิน/*takes* ยาลดกรด/ *an antacid* **([The patient] takes an antacid)**
EDU4:	แต่/*But* ก็ไม่หายปวด/*it cannot work.* **(But it cannot work)**

(b) EDU1symtom: [ผู้ป่วย] ปวดท้อง//*have a stomachache* **([A patient] has a stomachache.)**

EDU2symtom: [ผู้ป่วย] มี/*has* แก๊ส/*gas* ในกระเพาะ/*inside stomach* **([The patient] has gassy in the stomach)**

EDU3 treatment: [ผู้ป่วย] กินยาลดกรด/*Take an antacid* **([The patient] takes an antacid)**

EDU4: [ผู้ป่วย] รู้สึกดีขึ้น/ *Feel better* (**The patient] feels better**)

According to examples (a) and (b), the symptom-treatment relation occurs only on (b) because EDU4 of (b) contains "รู้สึกดีขึ้น/ *feel better*" as Class-cue-word of the symptom-treatment relation. Therefore, we propose automatically learning the symptom-treatment relation on documents by using the Naïve Bayes classifier, with the features from clustering objects with Dsym as $\langle v_{co1-1}w_{co1-1}, v_{co1-2}w_{co1-2}, \ldots, v_{co1-a}w_{co1-a}\rangle$ (where each symptom-concept EDU is determined by Word-CO, v_{co1} w_{co1}), and clustering the treatment features from AT and RT as $\langle v_{co2-1}w_{co2-1}, v_{co2-2}w_{co2-2}, \ldots, v_{co2-b/c}w_{co2-b/c}\rangle$ (where (each treatment-concept-EDU is determined by Word-CO, v_{co-2} w_{co-2})).

4 A Framework for Symptom-Treatment Relation Extraction

There are five steps in our framework. The first step is the corpus preparation step followed by the step of Word-CO concept learning, especially symptom concepts and treatment concepts. Then, the feature extraction step for symptom-treatment relation learning step which is followed by the symptom-treatment relation extraction steps are operated as shown in Fig. 2.

4.1 Corpus Preparation

This step is the preparation of a corpus in the form of EDU from the medical-care-consulting documents on the hospital's web-board of the nongovernmental-organization (NGO) website. The step involves using Thai word segmentation tools [10], including name entity [4]. After the word segmentation is achieved, EDU segmentation is then to be dealt with [5]. These annotated EDUs will be kept as an EDU corpus. This corpus contains 6000 EDUs of several diseases and is separated into two parts; one part is 4000 EDUs for learning the Word-CO concepts, boundaries of the symptom feature group and the treatment feature group, and the symptom-treatment relations, based on tenfold cross-validation. The other part of

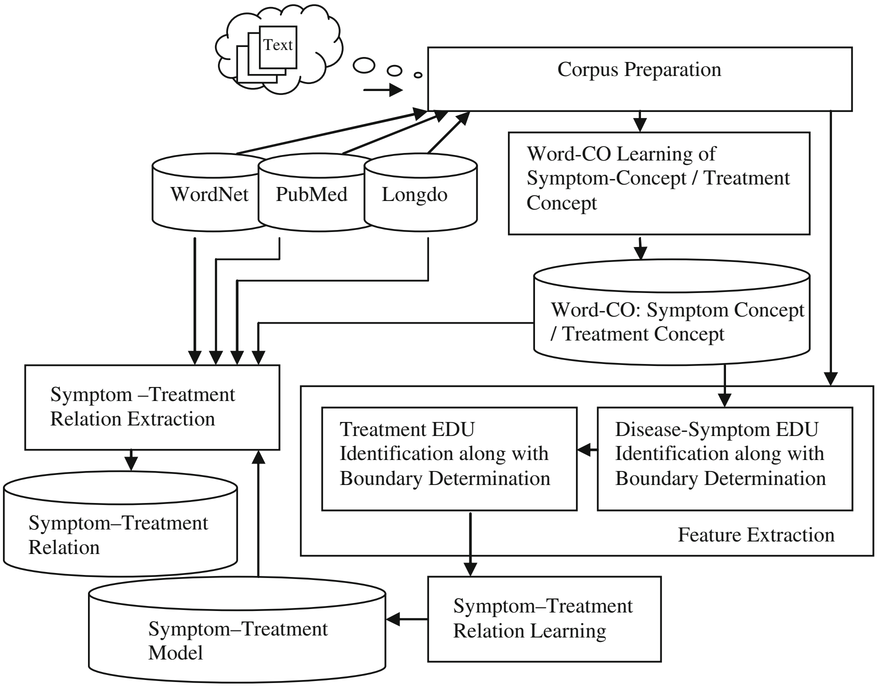


Fig. 2 System overview

2000 EDUs is for determining the boundaries and the symptom-treatment relation extraction. In addition to this step of corpus preparation, we semiautomatically annotate the Word-CO concepts of symptoms and treatments along with the Class-cue-word annotation to specify the cue word with the Class-type set {“yes”, “no”} of the symptom-treatment relation as shown in Fig. 3. All concepts of Word-CO are referred to Wordnet (http://word-net.princeton.edu/obtain) and MeSH after translating from Thai to English, by using Lexitron (the Thai-English dictionary) (http://lexitron.nectec.or.th/).

4.2 Word-CO Concept Learning

According to [6], the relatedness, r, has been applied in this research for the relatedness between two consecutive word in Word-CO with either the symptom concept, $v_{co1}\ w_{co1}$, or the treatment concept, $v_{co2}\ w_{co2}$ (as shown in Eq. 1), where each Word-CO, $v_{coi}\ w_{coi}$, existing on an EDU contains two relatedness $r(v_{coi}, w_{coi})$ values. If v_{coi} is v_{co1}, one relatedness value is the symptom concept and the other is

ปวดท้อง/ Stomachache

<**Symptom Boundary**>

<EDU>*น้องชาย*<**verb-co:** class=symptom ; concept='has a symptom'>*มีอาการ*</verb-co><**word-co**: class=symptom ; concept='stomachache'>*ปวดท้อง*</word-co>*อย่างหนัก* </EDU>

(A user brother has a stomachache heavily.)

<EDU>[*น้องชาย*]<**verb-co**: class=symptom ; concept='has'>*มี* </verb-co><**word-co**: class=symptom ; concept='gassy'>*แก๊ส*</word-co>*ในกระเพาะมาก* </EDU>

([The user brother] has lots of gas in the stomach.)

<EDU>[*น้องชาย*]*มักจะ*<**verb-co**: class=symptom ; concept='has a symptom'>*มีอาการ*</verb-co><**word-co**: class=symptom ; concept= 'appearance'> *เป็น*</word-co>*ตอนหลังทานข้าวเย็นและตอนกลางคืน* </EDU> **([The user brother] mostlyhas symptoms occurring after having dinner and at night.)**

</**Symptom Boundary**>

<EDU>[*ผู้ใช้*]*สงสัยเป็นโรคกระเพาะ* </EDU> **([The user] doubts to get a gastropathy.)**

<**Treatment Boundary** >

<EDU>*เลย*<**verb-co**: class=treatment; concept='consume'>*กิน*</verb-co><**word-co**: class=treatment ;concept='antacid'>*ยาลดกรด*</word-co>*เพื่อแก้ปวดท้อง* </EDU> **(Then [The user brother] takes an antacid to solve the stomach ache.)**

</**Treatment Boundary**>

<EDU>*แต่ก็*<**Class-cue-word: class**=no>*ไม่หาย*</Class-cue-word>*ปวด* </EDU> **(But it cannot work.)**
<EDU>*และปวดเพิ่มขึ้น*</EDU> **(And, [The user brother] has more pain)**

Fig. 3 Symptom-treatment relation annotation

the non-symptom concept. If v_{coi} is v_{co2}, one relatedness value is the treatment concept and the other is the non-treatment concept. The only v_{coi} w_{coi} co-occurrence with a higher $r(v_{coi}, w_{coi})$ value of the symptom concept or the treatment concept than that of the non-symptom concept or the non-treatment concept, respectively, is collected as an element of $VW_{symptom}$ or $VW_{treatment}$ (v_{co1} w_{co1} $\in VW_{symptom}$ where $VW_{symptom}$ is a set of Word-CO with the symptom concepts, and v_{co2} w_{co2} $\in VW_{treatment}$ where $VW_{treatment}$ is a set of Word-CO with the treatment concepts). $VW_{symptom}$ and $VM_{treatment}$ are used for identifying the disease-symptom concept EDU and the treatment concept EDU respectively.

$$r(v_{coi}, w_{coi}) = \frac{fv_{coi}w_{coi}}{fv_{coi} + fw_{coi} - fv_{coi}w_{coi}}. \quad (1)$$

where $r(v_{coi}, w_{co2})$ is the relatedness of Word-CO with a symptom of concept if coi = col1 or a treatment concept if $coi = co2$.

$v_{coi} \in V_{coi}$, $w_{coi} \in W_{coi}$ V_{co1} is a set of verbs with the symptom concepts.

V_{co2} is a set of verbs with the treatment concepts.

W_{co1} is the co-occurred word set having the symptom concept in the v_{co1} w_{co1} co-occurrence

W_{co2} is the co-occurred word set having the symptom concept in the v_{co2} w_{co2} co-occurrence

fv_{coi} is the numbers of v_{coi} occurences. fw_{coi} is the numbers of w_{coi} occurences

fv_{coi} w_{coi} is the numbers of v_{coi} and w_{coi} occurences

4.3 Feature Extraction

This step is to extract two feature groups used for classifying the symptom-treatment relation in the next step, the symptom feature group (which is Dsym) and the treatment feature group (which is AT/RT). Therefore, the symptom feature group and the treatment feature group can be extracted from the consecutive EDUs by using v_{co1} w_{co1} and $v_{\text{co}2}$ $w_{\text{co}2}$ to identify the starting EDU of Dsym and the starting EDU of AT/RT respectively. Then, we learn the probability of a Word-CO pair, v_{coi-j} w_{coi-j} $v_{coi\text{j}+1}$ $w_{coi-j+1}$, with the symptom concept class (where $coi = co1$) and the treatment concept class (where coi = co2) from the learning corpus with a window size of two consecutive EDUs with one sliding EDU distance (where $i = \{1, 2\}, j = \{1, 2, \ldots, \text{endOfboundary}\}$). The testing corpus of 2000EDUs is used to determine the boundary of the symptom feature group and the boundary of the treatment feature group by Naïve Bayes as shown in Eq. (2)

$$\begin{aligned} \text{EDUBoundaryClass} &= \underset{\text{class} \in \text{Class}}{\arg\max}\ P(\text{class}|v_{coi\_j}w_{coi\_j}\ \ v_{coi\_j+1}w_{coi\_j+1}) \\ &= \underset{\text{class} \in \text{Class}}{\arg\max}\ P(v_{coi\_j}w_{coi\_j}|\text{class})P(v_{coi\_j+1}w_{coi\_j+1}|\text{class})P(\text{class} \end{aligned} \tag{2}$$

where

v_{coi_j} $w_{coi\_j} \in \text{VW}_{\text{symptom}}$, $v_{coi\_j+1} \in \text{VW}_{\text{symptom}}$
$coi = co1$ $\text{VM}_{\text{symptom}}$ is a set of Word-CO with the symptom concepts
$v_{coi\_j}w_{coi\_j} \in \text{VW}_{\text{treatment}}$, $v_{coi\_j+1}w_{coi\_j+1} \in \text{VW}_{\text{treatment}}$
$coi = co2$ and $\text{VM}_{\text{treatment}}$ is a set of Word-CO with the treatment concepts
$j = 1, 2, \ldots$, endOfboundary Class = {“yes”, “no”)

4.4 Symptom-Treatment Relation Learning

It is necessary to cluster objects (or patients posting problems on the web-board) for enhancing the efficiency of learning the symptom-treatment relation because there is high symptom diversity depending on patients, diseases, environment, etc. We cluster the n samples of posted problems on the web-board by using k-mean as shown in Eq. (3) [2].

$$\text{Cluser}(x_j) = \underset{1 \le k \le K}{\arg\min}\ \left\|x_j - \mu_k\right\|^2 \tag{3}$$

where x_j is a disease-symptom vector, Dsym, of an object $\langle v_{co1-1}w_{co1-1}, v_{co1-2}w_{co1-2}, \ldots, v_{co1-a}w_{co1-a}\rangle$ and $j = 1, 2, \ldots, n$ posted problems. μ_k is the mean vector of the kth cluster. The highest number of $v_{co1-i}w_{co1-i}$ occurrences in each cluster is selected to its cluster representative. Thus, we have a symptom cluster set (Y) {rhinorrhoea-based-cluster, abdominalPain-based-cluster, brainSymptom-based-cluster,…nSymptom-based-cluster}.

From Eq. (3), we replace x_j with x_j to cluster the treatment features where x_j is a Word-CO element, v_{co2-i} w_{co2-i}, of AT ∪ RT and $j = 1, 2, \ldots, m$ Word-COs, v_{co2} w_{co2}. After clustering the treatment features, the highest number of the general concept (based on WordNet and MesH) of $v_{co2-i}w_{co2-i}$ occurrences in each cluster is selected to its cluster representative. Then we have a treatment cluster set (Z) {relax-based-cluster, foodControl-based-cluster, injectionControl-based-cluster, …mTreatment-based-cluster}.

According to clustering the extracted feature vectors from Sect. 4.3, we learn the symptom-treatment relation by using Weka (http://www.cs.wakato.ac.nz/ml/weka/) to determine probabilities of y, z_1, …, z_h with the Class-type set of the symptom-treatment relation, {'yes' 'no'} where $y \in$ Y, $z_1, \ldots, z_h \in$ Z, and h is max (b,c) from AT and RT. The Class-type set is specified on any five EDUs right after AT or RT. An element of the Class-type set is determined from the following set of Class-cue-word pattern.

Class-cue-word pattern = {'**cue**:หาย/disappear = **class**: yes', '**cue**:รู้สึกดีขึ้น/feel better = **class**: yes', 'cue:ไม่ปวด/do not pain = **class**: yes', '**cue:**"" = **class**: yes', '**cue**:ไม่หาย/appear = **class**: no','**cue**:ยังปวดอยู่/still pain = **class**: no', '**cue**:ปวดมากขึ้น/have more pain = **class**: no',…}

4.5 Symptom-Treatment Relation Extraction

The objective of this step is to recognize and extract the symptom-treatment relation from the testing EDU corpus by using Naïve Bayes in Eq. (4) with probabilities of y, z_1, …, z_h from the previous step with the algorithm shown in Fig. 4.

$$\begin{aligned} \text{SymTreat\_RelClass} &= \underset{\text{class} \in \text{Class}}{\arg\max}\; P(\text{class}|y, z_1, z_2, .., z_h) \\ &= \underset{\text{class} \in \text{Class}}{\arg\max}\; P(y|\text{class})P(z_1|\text{class})P(z_2|\text{class})..P(z_h|\text{class})P(\text{class}) \end{aligned} \tag{4}$$

where

$y \in Y$ Y is a symptom clusterset.
$Z_1, Z_2, \ldots, Z_H \in Z$ Z is a treatmentcluster set.
Class = {"yes" "no")

The extracted symptom-treatment relation of this step can be used for constructing PSM as shown in Fig. 5.

```
Assume that each EDU is represented by (NP VP).      L is a list of
EDU.  VW_symptom is a set of word-order-pairs having the symptom concepts
and VW_treatment is a set of word-order-pairs having the treatment
concepts (see section4.2).  v_co1∈V_co1,  v_co2∈V_co2,  w_co1∈W_co1,  w_co2∈W_co2 (see
section 3.1 )
MEDICINAL_PROPERTY_EXTRACTION( L, V_co1, V_co2, W_co1, W_co2 )

 i← 1;j←1; R←∅; flag←0; SymptomVector ← ∅;
while i ≤ length[L] do
  { while flag = 0  /*findSymptomConceptEDU
      if  v_s-i w_s-i∈ VW_symptom  then  flag=1
      else  i++ ;
    While notEndofBoundary and v_co1-i w_co1-i∈ VW_symptom
                                /*findSymptomFeatureVector
    { equation2, SymptomVector← SymptomVector ∪ v_co1-i w_co1-I;
          i ++ };
    cluster SymptomFeatureVector /*equation 3
    Flag← 0 ; j←1; treatmentVector ←∅;
    while flag = 0  /*findTreatmentConceptEDU
       if  v_co2-j w_co2-j∈ VW_treatment   then  flag=1
       else  {i++ ;j++};
    While notEndofBoundary and v_co2-j w_co2-j∈ VW_treatment
                               /*findTreatmentFeatureVector
    {equation2, treatmentVector←treatmentVector ∪ v_co2-j w_co2-j;
                j++; i++};
  cluster TreatmentFeatureVector /*equation 3
  SymptomTreatmentRelationExtraction by equation 4
   if SymptomTreatmentRelation = yes   then
     {R ← R ∪{⟨SymptomVector⟩+⟨TreatmentVector⟩};
       i++ };
```

Fig. 4 Symptom-treatment relation extraction algorithm

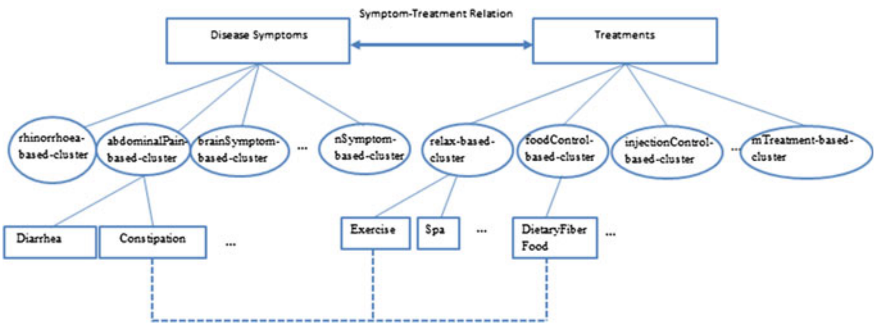


Fig. 5 Shows the PSM representation of the symptom-treatment relation

5 Evaluation and Conclusion

The Thai corpora used to evaluate the proposed symptom-treatment relation extraction algorithm consist of about 2,000 EDUs collected from the hospital's web-board documents of medical-care-consulting. The evaluation of the symptom-treatment relation extraction performance of this research methodology is expressed in terms of the precision and recall. The results of precision and recall are evaluated by three expert judgments with max win voting. The precision of the extracted symptom-treatment relation after clustering is 87.5 and 75.4 % recall. These research results, especially the low recall, can be increased if the interrupt occurrences on either a symptom boundary or a treatment boundary, as shown in the following, are solved.

EDU1: หนูมีอาการท้องผูกค่ะ (**I have a constipation symptom.**)
EDU2: [หนู]พยายามฝึกถ่ายทุกวัน (**[I] try to train excretion every day.**)
EDU3: ได้ผล (**It can work**)
EDU4: แต่หนูต้องกินโยเกิร์ตด้วย: (**But I must have yogurt too**)

where EDU3 is an interrupt to the treatment-concept-EDU boundary (EDU2 and EDU4). Moreover, our extracted symptom-treatment relation can be represented by PSM (Fig. 5) which is beneficial for patients to understand the disease symptoms and their treatment. However, the extracted symptoms and the extracted treatments are various to the patient characteristics, environment, time, etc. Therefore, the generalized symptoms and the generalized treatments have to be solved before constructing PSM.

Acknowledgment This work has been supported by the Thai Research Fund grant MRG5580030.

References

1. Abacha, A.B., Zweigenbaum, P.: Automatic extraction of semantic relations between medical entities: a rule based approach. J. Biomed. Semant. **2**(Suppl 5): S4 (2011). http://www.jbiomedsem.com/content/2/S5/S4
2. Aloise, D., Deshpande, A., Hansen, P., Popat, P.: NP-hardness of Euclidean sum-of-squares clustering. Mach. Learn. **75**, 245–249 (2009)
3. Carlson, L., Marcu, D., Okurowski, M.E.: Building a discourse-tagged corpus in the framework of rhetorical structure theory. In: Current Directions in Discourse and Dialogue, pp. 85−112 (2003)
4. Chanlekha, H., Kawtrakul, A.: Thai named entity extraction by incorporating maximum entropy model with simple heuristic information. In: Proceedings of IJCNLP' 2004 (2004)
5. Chareonsuk, J ., Sukvakree, T., Kawtrakul, A.: Elementary discourse unit segmentation for Thai using discourse cue and syntactic information. In: Proceedings of NCSEC 2005 (2005)
6. Guthrie, J. A., Guthrie, L., Wilks, Y., Aidinejad, H.: Subject-dependent co-occurrence and word sense disambiguation. In: Proceedings of the 29th Annual Meeting on Association for Computational Linguistics (1991)

7. Mitchell, T.M.: Machine Learning. The McGraw-Hill Companies Inc. and MIT Press, Singapore (1997)
8. Rosario, B.: Extraction of semantic relations from bioscience text. A dissertation submitted in partial satisfaction of the requirements for the degree of Doctor of Philosophy in Information Management and Systems. University of California, Berkeley (2005)
9. Song, S.-K., Oh, H.-S., Myaeng, S.H., Choi, S.-P., Chun, H.-W., Choi, Y.-S., Jeong, C.-H.: Procedural knowledge extraction on MEDLINE. In: AMT 2011, LNCS 6890, pp. 345–354 (2011)
10. Sudprasert, S., Kawtrakul, A.: Thai word segmentation based on global and local unsupervised learning. In: Proceedings of NCSEC'2003 (2003)

Part II
Information

An Argument-Dependent Approach to Determining the Weights of IFOWA Operator

Cuiping Wei and Xijin Tang

Abstract Based on entropy and similarity measure of intuitionistic fuzzy sets, a novel approach is proposed to determine weights of the IFOWA operator in this paper. Then, an intuitionistic fuzzy dependent OWA (IFDOWA) operator is defined and applied to handling multi-attribute group decision making problem with intuitionistic fuzzy information. Finally, an example is given to demonstrate the rationality and validity of the proposed approach.

Keywords Multi-attribute group decision-making · Intuitionistic fuzzy sets · Intuitionistic fuzzy-dependent OWA operator · Entropy · Similarity

1 Introduction

The ordered weighted aggregating (OWA) operator [26], as an important tool for aggregating information, has been investigated and applied in many documents [1, 9, 12, 20, 25, 32]. One critical issue of the OWA operator is to determine its associated weights. Up to now, a lot of methods have been proposed to determine the OWA weights. Xu [21] classified all those weight-determining approaches into two categories: argument-independent approaches [6, 12, 15, 20, 26, 28] and argument-dependent approaches [1, 7, 21, 23, 27, 29]. For the first category, Yager [26] suggested an approach to compute the OWA weights based on linguistic quantifiers provided by Zadeh [30, 31]. O'Hagan [12] defined degree of orness and

C. Wei
College of Mathematical Sciences, Yangzhou University, Yangzhou 225002, China
e-mail: cpwei@yzu.edu.cn

C. Wei
Management School, Qufu Normal University, Rizhao 276826, China

X. Tang (✉)
Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China
e-mail: xjtang@iss.ac.cn

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_7

constructed a nonlinear programming to obtain the weights of OWA operator. Xu [20] made an overview of methods for obtaining OWA weights and developed a novel weight-determining method using the idea of normal distribution. For the second category, Filev and Yager [7] developed two procedures to determine the weights of OWA operator. Xu and Da [23] established a linear objective-programming model to obtain the OWA weights. Xu [21] proposed a new dependent OWA operator which can relieve the influence of unfair arguments on the aggregated results. In [27, 29], Yager and Filev developed an argument-dependent method to generate the OWA weights with power function of the input arguments.

With the growing research of intuitionistic fuzzy set theory [2, 3] and the expansion of its application, it is more and more important to aggregate intuitionistic fuzzy information effectively. Xu [22, 24] proposed some intuitionistic fuzzy aggregation operators to aggregate the intuitionistic fuzzy information. In [22], Xu pointed out that the intuitionistic fuzzy OWA (IFOWA) weights can be obtained similar to the OWA weights, such as the normal distribution-based method. However, the characteristics of the input arguments are not considered in these methods.

In this paper, we investigate the IFOWA operator, and establish a new argument-dependent method to determine the IFOWA weights. To do that, this paper is organized as follows. Section 2 reviews the basic concepts about intuitionistic fuzzy information. In Sect. 3, a new argument-dependent approach to obtain the IFOWA weights is proposed based on entropy and similarity measure. A intuitionistic fuzzy dependent OWA (IFDOWA) operator is developed and its properties are studied. Section 4 provides a practical approach to solve multi-attribute group decision making problem with intuitionistic fuzzy information based on IFDOWA operator. The concluding remarks are given in Sect. 5.

2 Preliminaries

Some basic concepts of intuitionistic fuzzy sets, some operators, entropy and similarity measures are reviewed.

2.1 The OWA Operator and Intuitionistic Fuzzy Sets

Definition 2.1 [26] Let $(a_1, a_2, \ldots, a_n)$ be a collection of numbers. An ordered weighted averaging (OWA) operator is a mapping: $R^n \rightarrow R$, such that

$$\mathrm{OWA}(a_1, a_2, \ldots, a_n) = w_1 a_{\sigma(1)} + w_2 a_{\sigma(2)} + \cdots + w_n a_{\sigma(n)}, \tag{1}$$

where $a_{\sigma(j)}$ is the jth largest of $a_j (j = 1, 2, \ldots, n)$, and $w = (w_1, w_2, \ldots, w_n)^T$ is an associated vector of the operator with $w_j \in [0, 1]$ and $\sum_{j=1}^{n} w_j = 1$.

Definition 2.2 [2, 3] Let X be a universe of discourse. An intuitionistic fuzzy set (IFS) in X is an object with the form

$$A = \{\langle x, \mu_A(x), \nu_A(x)\rangle | x \in X\} \quad (2)$$

where $\mu_A : X \rightarrow [0,1], \nu_A : X \rightarrow [0,1]$ with the condition $0 \le \mu_A(x) + \nu_A(x) \le 1$, $\forall x \in X$. The numbers $\mu_A(x)$ and $\nu_A(x)$ denote the degree of membership and non-membership of x to A, respectively.

For each IFS A in X, we call $\pi_A(x) = 1 - \mu_A(x) - \nu_A(x)$ the intuitionistic index of x in A, which denotes the hesitancy degree of x to A.

For convenience, we call $\alpha = (\mu_\alpha, \nu_\alpha)$ an intuitionistic fuzzy value (IFV) [24], where $\mu_\alpha \in [0,1], \nu_\alpha \in [0,1]$, and $\mu_\alpha + \nu_\alpha \le 1$. Let Θ be the universal set of IFVs.

For comparison of IFVs, Chen and Tan [5] defined a score function while Hong and Choi [8] defined an accuracy function. Based on the two functions, Xu [24] provided a method to compare two intuitionistic fuzzy values (IFVs).

Definition 2.3 [24] Let $\alpha = (\mu_\alpha, \nu_\alpha)$ and $\beta = (\mu_\beta, \nu_\beta)$ be two IFVs, $s(\alpha) = \mu_\alpha - \nu_\alpha$ and $s(\beta) = \mu_\beta - \nu_\beta$ be the score degrees of α and β, respectively; $h(\alpha) = \mu_\alpha + \nu_\alpha$ and $h(\beta) = \mu_\beta + \nu_\beta$ be the accuracy degrees of α and β, respectively. Then

(1) If $s(\alpha) < s(\beta)$, then α is smaller than β, denoted by $\alpha < \beta$;
(2) If $s(\alpha) = s(\beta)$, then

  (1) If $h(\alpha) = h(\beta)$, then α and β represent the same information, i.e., $\mu_\alpha = \mu_\beta$ and $\nu_\alpha = \nu_\beta$, denoted by $\alpha = \beta$;
  (2) If $h(\alpha) < h(\beta)$, then α is smaller than β, denoted by $\alpha < \beta$;
  (3) If $h(\alpha) > h(\beta)$, then α is bigger than β, denoted by $\alpha > \beta$.

Definition 2.4 [22, 24] Let $\alpha = (\mu_\alpha, \nu_\alpha)$ and $\beta = (\mu_\beta, \nu_\beta)$ be two IFVs. Then, two operational laws of IFVs are given as follows:

(1) $\overline{\alpha} = (\nu_\alpha, \mu_\alpha)$;
(2) $\alpha \oplus \beta = (\mu_\alpha + \mu_\beta - \mu_\alpha\mu_\beta, \nu_\alpha\nu_\beta)$;
(3) $\lambda\alpha = (1 - (1 - \mu_\alpha)^\lambda, \nu_\alpha^\lambda), \lambda \ge 0$;
(4) $\lambda(\alpha_1 + \alpha_2) = \lambda\alpha_1 + \lambda\alpha_2$;
(5) $\lambda_1\alpha + \lambda_2\alpha = (\lambda_1 + \lambda_2)\alpha$.

With the thorough research of intuitionistic fuzzy set theory and the continuous expansion of its application scope, it is more and more important to aggregate intuitionistic fuzzy information effectively. Xu [22, 24] proposed some intuitionistic fuzzy aggregation operators to aggregate the intuitionistic fuzzy information.

Definition 2.5 [22] Let $\alpha_i = (\mu_{\alpha_i}, \nu_{\alpha_i})(i = 1, 2, \ldots, n)$ be a collection of IFVs. An intuitionistic fuzzy weighted averaging (IFWA) operator is a mapping: $\Theta^n \rightarrow \Theta$, such that

$$\text{IFWA}(\alpha_1, \alpha_2, \ldots, \alpha_n) = w_1\alpha_1 \oplus w_2\alpha_2 \oplus \cdots \oplus w_n\alpha_n = \left(1 - \prod_{j=1}^{n}(1 - \mu_{\alpha_j})^{w_j}, \prod_{j=1}^{n}\nu_{\alpha_j}^{w_j}\right) \quad (3)$$

where $w = (w_1, w_2, \ldots, w_n)^T$ is the weighting vector of $\alpha_i (i = 1, 2, \ldots, n)$ with $w_j \in [0, 1]$ and $\sum_{j=1}^{n} w_j = 1$.

Definition 2.6 [22] Let $\alpha_i = (\mu_{\alpha_i}, \nu_{\alpha_i})(i = 1, 2, \ldots, n)$ be a collection of IFVs. An intuitionistic fuzzy ordered weighted averaging (IFOWA) operator is a mapping: $\Theta^n \to \Theta$, such that

$$\begin{aligned} \mathrm{IFOWA}(\alpha_1, \alpha_2, \ldots, \alpha_n) &= w_1 \alpha_{\sigma(1)} \oplus w_2 \alpha_{\sigma(2)} \oplus \cdots \oplus w_n \alpha_{\sigma(n)} \\ &= \left(1 - \prod_{j=1}^{n} (1 - \mu_{\alpha_{\sigma(j)}})^{w_j}, \prod_{j=1}^{n} \nu_{\alpha_{\sigma(j)}}^{w_j} \right) \end{aligned} \tag{4}$$

where $\alpha_{\sigma(j)}$ is the jth largest of $\alpha_j (j = 1, 2, \ldots, n)$, and $w = (w_1, w_2, \ldots, w_n)^T$ is an associated vector of the operator with $w_j \in [0, 1]$ and $\sum_{j=1}^{n} w_j = 1$.

2.2 Entropy and Similarity Measure for IFSs

Introduced by Burillo and Bustince [4], Intuitionistic fuzzy entropy is used to estimate the uncertainty of an IFS. Szmidt and Kacprzyk [13] defined an entropy measure E_{SK} for an IFS. Wang and Lei [14] gave an entropy measure E_{WL}

$$E_{SK}(A) = \frac{1}{n} \sum_{i=1}^{n} \frac{\max Count(A_i \cap A_i^C)}{\max Count(A_i \cup A_i^C)}, \tag{5}$$

where $A_i = \{\langle x_i, \mu_A(x_i), \nu_A(x_i)\rangle\}$ is a single element IFS, $A_i \cap A_i^C = \{\langle x_i, \min\{\mu_A(x_i), \nu_A(x_i)\}, \max\{\mu_A(x_i), \nu_A(x_i)\}\rangle\}$, $A_i \cup A_i^C = \{\langle x_i, \max\{\mu_A(x_i), \nu_A(x_i)\}, \min\{\nu_A(x_i), \mu_A(x_i)\}\rangle\}$. For every IFS A, $\max Count(A) = \sum_{i=1}^{n} (\mu_A(x_i) + \pi_A(x_i))$ is the biggest cardinality of A.

$$E_{WL}(A) = \frac{1}{n} \sum_{i=1}^{n} \frac{\min\left\{\mu_A(x_i), \nu_A(x_i)\right\} + \pi_A(x_i)}{\max\left\{\mu_A(x_i), \nu_A(x_i)\right\} + \pi_A(x_i)}. \tag{6}$$

Wei and Wang [18] proved that E_{SK} and E_{WL} are equivalent. For convenience, we use the entropy measure E_{WL} in the following.

Based on E_{WL}, the entropy measure for an intuitionistic fuzzy value $\alpha = (\mu_\alpha, \nu_\alpha)$ can be given as

$$E(\alpha) = \frac{\min\{\mu_\alpha, \nu_\alpha\} + \pi_\alpha}{\max\{\mu_\alpha, \nu_\alpha\} + \pi_\alpha}. \tag{7}$$

Similarity measure [10], another important topic in the theory of intuitionistic fuzzy sets, is to describe the similar degree between two IFSs. Wei and Tang [17] constructed a new similarity measure S_{WT} for IFSs based on entropy measure E_{WL}.

$$S_{WT}(A,B)=\frac{1}{n}\sum_{i=1}^{n}\frac{1-\min\{|\mu_A(x_i)-\mu_B(x_i)|,|\nu_A(x_i)-\nu_B(x_i)|\}}{1+\max\{|\mu_A(x_i)-\mu_B(x_i)|,|\nu_A(x_i)-\nu_B(x_i)|\}}. \tag{8}$$

Now we give a similarity measure between two IFVs $\alpha=(\mu_\alpha,\nu_\alpha)$ and $\beta=(\mu_\beta,\nu_\beta)$ based on S_{WT}:

$$S(\alpha,\beta)=\frac{1-\min\{|\mu_\alpha-\mu_\beta|,|\nu_\alpha-\nu_\beta|\}}{1+\max\{|\mu_\alpha-\mu_\beta|,|\nu_\alpha-\nu_\beta|\}}. \tag{9}$$

3 IFDOWA Operator and Its Properties

In [22], Xu pointed out that the IFOWA weights can be determined similarly to the OWA weights. For example, we can use the normal distribution-based method. However, those methods belong to the category of argument-independent approaches. Here we develop an argument-dependent approach to determine the IFOWA weights based on intuitionistic fuzzy entropy and similarity measure.

We suppose $\alpha_j=(\mu_{\alpha_j},\nu_{\alpha_j})(j=1,2,\ldots,n)$ is a collection of IFVs, $(\alpha_{\sigma(1)},\alpha_{\sigma(2)},\ldots,\alpha_{\sigma(n)})$ is a permutation of $(\alpha_1,\alpha_2,\ldots,\alpha_n)$ such that $\alpha_{\sigma(i)}\geq\alpha_{\sigma(j)}$ for all $i\leq j$. The weighting vector of IFOWA operator $w=(w_1,w_2,\ldots,w_n)^T$ is to be determined, such that $w_j\in[0,1]$ and $\sum_{j=1}^{n}w_j=1$.

During the information aggregating process, we usually expect that the uncertainty degrees of arguments are as small as possible. Thus, the smaller uncertainty degree of argument $\alpha_{\sigma(j)}$, the bigger the weight w_j. Conversely, the bigger uncertainty degree of argument $\alpha_{\sigma(j)}$, the smaller the weight w_j. The uncertainty degrees of arguments can be measured by Formula (7). Thus, the weighting vector of the IFOWA operator can be defined as:

$$w_j^a=\frac{1-E(\alpha_{\sigma(j)})}{\sum\limits_{j=1}^{n}[1-E(\alpha_{\sigma(j)})]},\quad j=1,2,\ldots,n. \tag{10}$$

In the following, we define the weighting vector of the IFOWA operator from another viewpoint. In real-life situation, the arguments $\alpha_{\sigma(j)}(j=1,2,\ldots,n)$ usually take the form of a collection of n preference values provided by n different individuals. Some individuals may assign unduly high or unduly low preference values to their preferred or repugnant objects. In such a case, we shall assign very small weights to these "false" or "biased" opinions, that is to say, the more similar an argument $\alpha_{\sigma(j)}$ is to others, the bigger the weight w_j. Conversely, the less similar an argument $\alpha_{\sigma(j)}$ is to others, the smaller the weights w_j. The similar degree between two arguments can be calculated by Formula (9).

Definition 3.1 Let $\alpha_i = (\mu_{\alpha_i}, \nu_{\alpha_i})(i = 1, 2, \ldots, n)$ be a collection of IFVs, $(\alpha_{\sigma(1)}, \alpha_{\sigma(2)}, \ldots, \alpha_{\sigma(n)})$ is a permutation of $(\alpha_1, \alpha_2, \ldots, \alpha_n)$ such that $\alpha_{\sigma(i)} \geq \alpha_{\sigma(j)}$ for all $i \leq j$. Then, the overall similarity degree between $\alpha_{\sigma(j)}$ and other arguments $\alpha_{\sigma(l)}(l = 1, 2, \ldots, n, l \neq j)$ is defined as

$$S(\alpha_{\sigma(j)}) = \sum_{\substack{l=1 \\ l \neq j}}^{n} S(\alpha_{\sigma(j)}, \alpha_{\sigma(l)}), \quad j = 1, 2, \ldots, n. \tag{11}$$

So, we define the weighting vector $w = (w_1, w_2, \ldots, w_n)^T$ of the IFOWA operator as following:

$$w_j^b = \frac{S(\alpha_{\sigma(j)})}{\sum_{j=1}^{n} S(\alpha_{\sigma(j)})}, \quad j = 1, 2, \ldots, n. \tag{12}$$

According to the above analysis, the weighting vector of the IFOWA operator associates not only with w^a, but also with w^b. Thus, we use the linear weighting method to derive the combined weighting vector of the IFOWA operator

$$w_j = \lambda w_j^a + (1 - \lambda) w_j^b, \quad \text{where } \lambda \in [0, 1], \quad j = 1, 2, \ldots, n. \tag{13}$$

Since $\sum_{j=1}^{n} [1 - E(\alpha_{\sigma(j)})] = \sum_{j=1}^{n} [1 - E(\alpha_j)]$ and $\sum_{j=1}^{n} S(\alpha_{\sigma(j)}) = \sum_{j=1}^{n} S(\alpha_j)$, Formulas (10), (12) and (13) can be rewritten as:

$$w_j^a = \frac{1 - E(\alpha_{\sigma(j)})}{\sum_{j=1}^{n} [1 - E(\alpha_j)]}, \quad j = 1, 2, \ldots, n. \tag{14}$$

$$w_j^b = \frac{S(\alpha_{\sigma(j)})}{\sum_{j=1}^{n} S(\alpha_j)}, \quad j = 1, 2, \ldots, n. \tag{15}$$

$$w_j = \frac{\lambda [1 - E(\alpha_{\sigma(j)})]}{\sum_{j=1}^{n} [1 - E(\alpha_j)]} + \frac{(1 - \lambda) S(\alpha_{\sigma(j)})}{\sum_{j=1}^{n} S(\alpha_j)}, \tag{16}$$

where $\lambda \in [0, 1] \quad j = 1, 2, \ldots, n$.

Definition 3.2 Let $\alpha_i = (\mu_{\alpha_i}, \nu_{\alpha_i})(i = 1, 2, \ldots, n)$ be a collection of IFVs. An intuitionistic fuzzy dependent OWA (IFDOWA) operator is a mapping: $\Theta^n \to \Theta$, such that

$$\begin{aligned}\text{IFDOWA}(\alpha_1, \alpha_2, \ldots, \alpha_n) &= w_1\alpha_{\sigma(1)} \oplus w_2\alpha_{\sigma(2)} \oplus \cdots \oplus w_n\alpha_{\sigma(n)} \\ &= \left(1 - \prod_{j=1}^{n}(1 - \mu_{\alpha_{\sigma(j)}})^{w_j}, \prod_{j=1}^{n} \nu_{\alpha_{\sigma(j)}}^{w_j}\right)\end{aligned} \tag{17}$$

where $(\alpha_{\sigma(1)}, \alpha_{\sigma(2)}, \ldots, \alpha_{\sigma(n)})$ is a permutation of $(\alpha_1, \alpha_2, \ldots, \alpha_n)$ such that $\alpha_{\sigma(i)} \geq \alpha_{\sigma(j)}$ for all $i \leq j$, $w = (w_1, w_2, \ldots, w_n)^T$ is the associated weighting vector which can be calculated by Formula (16).

By Formulas (16) and (17), we obtain

$$\begin{aligned}\text{IFDOWA}(\alpha_1, \alpha_2, \ldots, \alpha_n) &= \bigoplus_{j=1}^{n} \alpha_{\sigma(j)} \left\{ \frac{\lambda[1 - E(\alpha_{\sigma(j)})]}{\sum_{j=1}^{n}[1 - E(\alpha_j)]} + \frac{(1-\lambda)S(\alpha_{\sigma(j)})}{\sum_{j=1}^{n} S(\alpha_j)} \right\} \\ &= \bigoplus_{j=1}^{n} \alpha_j \left\{ \frac{\lambda[1 - E(\alpha_j)]}{\sum_{j=1}^{n}[1 - E(\alpha_j)]} + \frac{(1-\lambda)S(\alpha_j)}{\sum_{j=1}^{n} S(\alpha_j)} \right\}\end{aligned} \tag{18}$$

Yager [27] pointed that an OWA operator is called neat if the aggregated value is independent of the ordering. Therefore, the IFDOWA operator is a neat operator. By Formulas (16) and (17), we can get the following properties.

Theorem 3.1 *Let $\alpha_i = (\mu_{\alpha_i}, \nu_{\alpha_i})(i = 1, 2, \ldots, n)$ be a collection of IFVs, $(\alpha_{\sigma(1)}, \alpha_{\sigma(2)}, \ldots, \alpha_{\sigma(n)})$ be a permutation of $(\alpha_1, \alpha_2, \ldots, \alpha_n)$ such that $\alpha_{\sigma(i)} \geq \alpha_{\sigma(j)}$ for all $i \leq j$. Suppose $E(\alpha_\sigma(j))$ is the entropy of $\alpha_\sigma(j)$ and $S(\alpha_\sigma(j))$ is the similarity degree between $\alpha_\sigma(j)$ and other arguments. If $E(\alpha_\sigma(i)) \leq E(\alpha_\sigma(j))$ and $S(\alpha_\sigma(i)) \geq S(\alpha_\sigma(j))$, then $w_i \geq w_j$.*

Theorem 3.2 *Let $\alpha_i = (\mu_{\alpha_i}, \nu_{\alpha_i})(i = 1, 2, \ldots, n)$ be a collection of IFVs. If $\alpha_i = \alpha_j$, for all i, j, then $w_j = \frac{1}{n}$ for all j.*

Yager [26] further introduced two characterizing measures called dispersion measure and orness measure, respectively, associated with the weighting vector w of the OWA operator, where the dispersion measure of the aggregation is defined as

$$disp(w) = -\sum_{j=1}^{n} w_j \ln w_j, \tag{19}$$

which measures the degree to which w takes into account the information in the arguments during the aggregation. Particularly, if $w_j = 0$ for any j, $disp(w) = 0$; if $w = (\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n})^T$, $disp(w) = \ln n$.

The second one, the orness measure of the aggregation, is defined as

$$orness(w) = \frac{1}{n-1}\sum_{j=1}^{n}(n-j)w_j, \tag{20}$$

which lies in the unit interval [0, 1] and characterizes the degree to which the aggregation is like an *or* operation. Particularly, if $w = (1, 0, \ldots, 0)^T$, $orness(w) = 1$; if $w = (\frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n})$, $disp(w) = 0.5$; if $w = (0, \ldots, 0, 1)^T$, $orness(w) = 0$.

From Formulas (16), (19) and (20), it follows that

$$disp(w) = -\sum_{j=1}^{n}\left\{\frac{\lambda[1-E(\alpha_{\sigma(j)})]}{\sum_{j=1}^{n}[1-E(\alpha_j)]} + \frac{(1-\lambda)S(\alpha_{\sigma(j)})}{\sum_{j=1}^{n}S(\alpha_j)}\right\}\cdot \tag{21}$$

$$\ln\left\{\frac{\lambda[1-E(\alpha_{\sigma(j)})]}{\sum_{j=1}^{n}[1-E(\alpha_j)]} + \frac{(1-\lambda)S(\alpha_{\sigma(j)})}{\sum_{j=1}^{n}S(\alpha_j)}\right\}.$$

$$orness(w) = \frac{1}{n-1}\sum_{j=1}^{n}(n-j)\cdot\left\{\frac{\lambda[1-E(\alpha_{\sigma(j)})]}{\sum_{j=1}^{n}[1-E(\alpha_j)]} + \frac{(1-\lambda)S(\alpha_{\sigma(j)})}{\sum_{j=1}^{n}S(\alpha_j)}\right\}. \tag{22}$$

Example 3.1 Let $\alpha_1 = (0.2, 0.5)$, $\alpha_2 = (0.4, 0.2)$, $\alpha_3 = (0.5, 0.4)$, $\alpha_4 = (0.3, 0.5)$, $\alpha_5 = (0.7, 0.1)$ be a collection of IFVs. The re-ordered argument $\alpha_j (j = 1, 2, 3, 4, 5)$ in descending order are $\alpha_{\sigma(1)} = (0.7, 0.1)$, $\alpha_{\sigma(2)} = (0.4, 0.2)$, $\alpha_{\sigma(3)} = (0.5, 0.4)$, $\alpha_{\sigma(4)} = (0.3, 0.5)$, $\alpha_{\sigma(5)} = (0.2, 0.5)$. Suppose $\lambda = 0.5$, by (14), (15) and (16), we obtain $w^a = (0.3823, 0.1433, 0.0956, 0.1638, 0.2150)$, $w^b = (0.1632, 0.2101, 0.2145, 0.2123, 0.1999)$.

Thus, $w = (0.27275, 0.17670, 0.15505, 0.18805, 0.20745)$. By (19) and (20), we have

$$disp(w) = -\sum_{j=1}^{5} w_j \ln w_j = 1.5902.$$

$$orness(w) = \frac{1}{5-1}\sum_{j=1}^{5}(5-j)w_j = 0.3609.$$

By (17) and (18), we have $\text{IFDOWA}(\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5) = (0.4724, 0.2648)$. Therefore, the collective argument is $(0.4724, 0.2648)$.

4 The Application of IFDOWA Operator in Multi-attribute Group Decision

In this section, we apply the IFDOWA operator to multi-attribute group decision-making problem which can be described as follows.

We suppose $X = \{x_1, x_2, \ldots, x_n\}$ is a set of evaluation alternatives, $D = \{d_1, d_2, \ldots, d_s\}$ is a set of decision makers, $U = \{u_1, u_2, \ldots, u_m\}$ is an attribute set, and $v = (v_1, v_2, \ldots, v_m)^T$ is a weighting vector of attributes such that $v_j \in [0, 1]$ and $\sum_{j=1}^{m} v_j = 1$. Let $R^{(k)} = \left(r_{ij}^{(k)}\right)_{n\times m}$ $(k = 1, 2, \ldots, s)$ be intuitionistic fuzzy decision matrices, where $r_{ij}^{(k)} = (\mu_{ij}^{(k)}, v_{ij}^{(k)})$ is an IFV and provided by the decision maker $d_k \in D$ for the alternative $x_i \in X$ with respect to the attribute $u_j \in U$.

Based on the IFWA operator and the IFDOWA operator, we rank the alternatives $x_i (i = 1, 2, \ldots, n)$ by the following steps:

Step 1. Utilize the IFWA operator to derive the individual overall aggregated values $z_i^{(k)} (i = 1, 2, \ldots, n,\ k = 1, 2, \ldots, s)$ of the alternatives $x_i (i = 1, 2, \ldots, n)$ by decision makers $d_k (k = 1, 2, \ldots, s)$, where

$$z_i^{(k)} = \text{IFWA}_v(r_{i1}^{(k)}, r_{i2}^{(k)}, \ldots, r_{im}^{(k)}) = v_1 r_{i1}^{(k)} \oplus v_2 r_{i2}^{(k)} \oplus \cdots \oplus v_m r_{im}^{(k)}, \tag{23}$$

where $v = (v_1, v_2, \ldots, v_m)^T$ is the weighting vector of the attributes of $u_j (j = 1, 2, \ldots, m)$, with $v_j \in [0, 1]$ and $\sum_{j=1}^{m} v_j = 1$.

Step 2. Utilize the IFDOWA operator to derive the overall aggregated values $z_i (i = 1, 2, \ldots, n)$ of the alternatives $x_i (i = 1, 2, \ldots, n)$, where

$$z_i = \text{IFDOWA}_w(z_i^{(1)}, z_i^{(1)}, z_i^{(2)}, \ldots, z_i^{(s)}) = w_1^{(i)} z_i^{\sigma(1)} \oplus w_2^{(i)} z_i^{\sigma(2)} \oplus \cdots \oplus w_s^{(i)} z_i^{\sigma(s)}, \tag{24}$$

where $w^{(i)} = (w_1^{(i)}, w_2^{(i)}, \ldots, w_s^{(i)}) (i = 1, 2, \ldots, n)$ are calculated by Formula (16).

Step 3. Utilize the Definition 2.2 to compare the overall aggregated values $z_i (i = 1, 2, \ldots, n)$ and rank the alternatives $x_i (i = 1, 2, \ldots, n)$.

We adopt the example used in [11, 19] to illustrate the proposed approach.

Example 4.1 The information management steering committee of Midwest American Manufacturing Corp. must prioritize for development and implementation a set of six information technology improvement projects $x_i (i = 1, 2, \ldots, 6)$, which have been proposed by area managers. The committee is concerned that the projects are prioritized from highest to lowest potential contribution to the firm's strategic goal of gaining competitive advantages in the industry. In assessing the potential contribution of each project, three factors are considered, u_1: productivity, u_2: differentiation, and u_3: management, whose weight vector is $v = (0.35, 0.35, 0.30)$. Suppose that there are four decision makers $d_k(k = 1, 2, 3, 4)$. They provided their preferences with IFVs $r_{ij}^{(k)} = (\mu_{ij}^{(k)}, v_{ij}^{(k)})(i = 1, 2, \dots, 6, j = 1, 2, 3)$ over the projects $x_i(i = 1, 2, \dots, 6)$ with respect to the factors $u_j(j = 1, 2, 3)$, which are listed as follows:

$$R^{(1)} = \begin{pmatrix} (0.3, 0.2) & (0.6, 0.1) & (0.5, 0.2) \\ (0.5, 0.1) & (0.3, 0.2) & (0.4, 0.2) \\ (0.4, 0.3) & (0.5, 0.2) & (0.3, 0.1) \\ (0.3, 0.1) & (0.5, 0.3) & (0.3, 0.2) \\ (0.4, 0.3) & (0.5, 0.3) & (0.4, 0.2) \\ (0.5, 0.4) & (0.2, 0.1) & (0.3, 0.2) \end{pmatrix} \quad R^{(2)} = \begin{pmatrix} (0.5, 0.3) & (0.2, 0.1) & (0.3, 0.3) \\ (0.3, 0.1) & (0.5, 0.3) & (0.4, 0.2) \\ (0.3, 0.4) & (0.4, 0.3) & (0.3, 0.1) \\ (0.5, 0.3) & (0.6, 0.3) & (0.5, 0.2) \\ (0.5, 0.3) & (0.3, 0.2) & (0.3, 0.2) \\ (0.5, 0.3) & (0.4, 0.3) & (0.2, 0.1) \end{pmatrix}$$

$$R^{(3)} = \begin{pmatrix} (0.4, 0.2) & (0.5, 0.1) & (0.5, 0.3) \\ (0.4, 0.1) & (0.6, 0.3) & (0.5, 0.2) \\ (0.2, 0.2) & (0.3, 0.1) & (0.5, 0.3) \\ (0.5, 0.4) & (0.6, 0.2) & (0.3, 0.1) \\ (0.6, 0.3) & (0.5, 0.2) & (0.6, 0.2) \\ (0.4, 0.2) & (0.3, 0.1) & (0.5, 0.1) \end{pmatrix} \quad R^{(4)} = \begin{pmatrix} (0.3, 0.1) & (0.5, 0.4) & (0.4, 0.3) \\ (0.5, 0.2) & (0.4, 0.3) & (0.7, 0.1) \\ (0.6, 0.1) & (0.4, 0.2) & (0.2, 0.1) \\ (0.3, 0.2) & (0.5, 0.3) & (0.3, 0.2) \\ (0.4, 0.3) & (0.3, 0.1) & (0.2, 0.2) \\ (0.3, 0.1) & (0.5, 0.2) & (0.4, 0.3) \end{pmatrix}$$

Step 1. Utilize the IFWA operator to derive the individual overall aggregated values $z_i^{(k)}(i = 1, 2, \dots, 6,\ k = 1, 2, 3, 4)$ of the alternatives $x_i(i = 1, 2, \dots, 6)$ by decision makers $d_k(k = 1, 2, 3, 4)$:

$$z_1^{(1)} = (0.4798, 0.1569),\ z_2^{(1)} = (0.4059, 0.1569),\ z_3^{(1)} = (0.4104, 0.1872),$$

$$z_4^{(1)} = (0.3778, 0.1808),\ z_5^{(1)} = (0.4371, 0.2656),\ z_6^{(1)} = (0.3480, 0.2000),$$

$$z_1^{(2)} = (0.3480, 0.2042),\ z_2^{(2)} = (0.4059, 0.1808),\ z_3^{(2)} = (0.3368, 0.2386),$$

$$z_4^{(2)} = (0.5376, 0.2656),\ z_5^{(2)} = (0.3778, 0.2305),\ z_6^{(2)} = (0.3864, 0.2158),$$

$$z_1^{(3)} = (0.4671, 0.1772),\ z_2^{(3)} = (0.5071, 0.1808),\ z_3^{(3)} = (0.3369, 0.1772),$$

$$z_4^{(3)} = (0.4884, 0.2071),\ z_5^{(3)} = (0.5675, 0.2305),\ z_6^{(3)} = (0.4004, 0.1275),$$

$$z_1^{(4)} = (0.4059, 0.2259),\ z_2^{(4)} = (0.5428, 0.1872),\ z_3^{(4)} = (0.4325, 0.1275),$$

$$z_4^{(4)} = (0.3778, 0.2305),\ z_5^{(4)} = (0.3097, 0.1808),\ z_6^{(4)} = (0.4059, 0.1772).$$

Step 2. Utilize the IFDOWA operator to derive the overall aggregated values $z_i(i = 1, 2, \dots, 6)$ of the alternatives $x_i(i = 1, 2, \dots, 6)$, where $\lambda = 0.5$:

$$z_1 = \text{IFDOWA}_{w^{(1)}}(z_1^{(1)}, z_1^{(2)}, z_1^{(3)}, z_1^{(4)}) = (0.4352, 0.1859),$$

$$z_2 = \text{IFDOWA}_{w^{(2)}}(z_2^{(1)}, z_2^{(2)}, z_2^{(3)}, z_2^{(4)}) = (0.4747, 0.1767),$$

$$z_3 = \text{IFDOWA}_{w^{(3)}}(z_3^{(1)}, z_3^{(2)}, z_3^{(3)}, z_3^{(4)}) = (0.3877, 0.1722),$$

$$z_4 = \text{IFDOWA}_{w^{(4)}}(z_4^{(1)}, z_4^{(2)}, z_4^{(3)}, z_4^{(4)}) = (0.4581, 0.2202),$$

$$z_5 = \text{IFDOWA}_{w^{(5)}}(z_5^{(1)}, z_5^{(2)}, z_5^{(3)}, z_5^{(4)}) = (0.4513, 0.2273),$$

$$z_6 = \text{IFDOWA}_{w^{(6)}}(z_6^{(1)}, z_6^{(2)}, z_6^{(3)}, z_6^{(4)}) = (0.3876, 0.1737).$$

Step 3. Utilize the score function to calculate the scores $s(z_i)(i = 1, 2, \ldots, 6)$ of overall aggregated values $z_i(i = 1, 2, \ldots, 6)$ of the alternatives $x_i(i = 1, 2, \ldots, 6)$:

$$s(z_1) = 0.2493, \ s(z_2) = 0.2980, \ s(z_3) = 0.2155,$$

$$s(z_4) = 0.2379, \ s(z_5) = 0.2240, \ s(z_6) = 0.2139.$$

Use the scores $s(z_i)(i = 1, 2, \ldots, 6)$ to rank the alternatives $x_i(i = 1, 2, \ldots, 6)$, we obtain

$$x_2 \succ x_1 \succ x_4 \succ x_5 \succ x_3 \succ x_6.$$

5 Concluding Remarks

In this paper, we proposed a new argument-dependent approach, based on entropy and similarity measure, to determine the weights of IFOWA operator. The approach could relieve the influence of unfair arguments on the aggregated results and reduce the uncertainty degrees of aggregated results. We then defined an IFDOWA operator and applied the operator to solving multi-attribute group decision making problems. It is worth noting that the results in this paper can be further extended to interval-valued intuitionistic fuzzy environment.

Acknowledgments The authors are grateful to the anonymous referees for their insightful and valuable suggestions to our original submission to the 8th International Conference on Knowledge,Information, and Creativity Support Systems (KICSS2013) [16]. The authors also owe gratitude to the on-site participants and the editors of proceedings for their comments for the modification of the conference paper into an extended and improved manuscript. The work is supported by the Natural Science Foundation of China (71171187, 71371107), the National Basic Research Program of China (2010CB731405), and Science Foundation of Shandong Province (ZR2013GM011).

References

1. Ahn, B.S.: Preference relation approach for obtain OWA operators weights. Int. J. Approximate Reasoning **47**, 166–178 (2008)
2. Atanassov, K.: Intuitionistic fuzzy sets. Fuzzy Sets Syst. **20**(1), 87–96 (1986)
3. Atanassov, K.: Intuitionistic Fuzzy Sets: Theory and Applications. Physica-Verlag, Heidelberg (1999)
4. Burillo, P., Bustince, H.: Entropy on intuitionistic fuzzy sets and on interval-valued fuzzy sets. Fuzzy Sets Syst. **78**, 305–316 (1996)
5. Chen, S.M., Tan, J.M.: Handling multi-criteria fuzzy decision making problems based on vague set theory. Fuzzy Sets Syst. **67**(2), 163–172 (1994)
6. Emrouznejad, A., Amin, G.: Improving minimax disparity model to determine the OWA operator weights. Inf. Sci. **180**, 1477–1485 (2010)
7. Filev, D.P., Yager, R.R.: On the issue of obtaining OWA operator weights. Fuzzy Sets Syst. **94**, 157–169 (1998)
8. Hong, D.H., Choi, C.H.: Multicriteria fuzzy decision-making problems based on vague set theory. Fuzzy Sets Syst. **114**, 103–113 (2000)
9. Liang, X., Wei, C.P., Chen, Z.M.: An intuitionistic fuzzy weighted OWA operator and its application. Int. J. Mach. Learn. Cybern. (2013). doi:10.1007/s13042-012-0147-z
10. Li, D.F., Cheng, C.T.: New similarity measure of intuitionistic fuzzy sets and application to pattern recongnitions. Pattern Recogn. Lett. **23**, 221–225 (2002)
11. Ngwenyama, O., Bryson, N.: Eliciting and mapping qualitative preferences to numeric rankings in group decision making. Eur. J. Oper. Res. **116**, 487–497 (1999)
12. O'Hagan, M.: Aggregating template rule antecedents in real-time expert systems with fuzzy set. In: Proceedings of 22nd Annual IEEE Asilomar Conference on Signals, Systems and Computers Pacific Grove, pp. 681–689, CA (1988)
13. Szmidt, E., Kacprzyk, J.: Entropy for intuitionistic fuzzy sets. Fuzzy Sets Syst. **118**(3), 467–477 (2001)
14. Wang, Y., Lei, Y.J.: A technique for constructing intuitionistic fuzzy entropy. Control Decis. **22**(12), 1390–1394 (2007)
15. Wang, Y.M., Parkan, C.: A minimax disparity approach for obtain OWA operator weights. Inf. Sci. **175**, 20–29 (2005)
16. Wei, C.P., Tang, X.J., Bi, Y.: Intuitionistic fuzzy dependent OWA operator and its application. In: Skulimowski, A.M.J. (ed.) Looking into the Future of Creativity and Decision Support Systems: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems, Krakow, Poland, 79, Nov 2013, Advances in Decision Sciences and Future Studies, vol. 2, pp. 298–309. Progress & Business Publishers, Krakow (2013)
17. Wei, C.P., Tang, X.J.: An intuitionistic fuzzy group decision making approach based on entropy and similarity measures. Int. J. Inf. Technol. Decis. Making **10**(6), 1111–1130 (2011)
18. Wei, C.P., Wang, P.: Entropy, similarity measure of interval-valued intuitionistic fuzzy sets and their applications. Inf. Sci. **181**(19), 4273–4286 (2011)
19. Xia, M.M., Xu, Z.S.: Some new similarity measures for intuitionistic fuzzy values and their application in group decision making. J. Syst. Sci. Syst. Eng. **19**(4), 430–452 (2011)
20. Xu, Z.S.: An overview of methods for determining OWA weights. Int. J. Intell. Syst. **20**(8), 843–865 (2005)
21. Xu, Z.S.: Dependent OWA operator. In: Torra, V., Narukawa, Y., Valls, A., Domin-goFerrer, J. (eds.) MDAI 2006, LNCS (LNAI), vol. 3885, pp. 172–178. Springer, Heidelberg (2006)
22. Xu, Z.S.: Intuitionistic fuzzy aggregation operators. IEEE Trans. Fuzzy Syst. **15**, 1179–1187 (2007)
23. Xu, Z.S., Da, Q.L.: The uncertain OWA operator. Int. J. Intell. Syst. **17**, 569–575 (2002)
24. Xu, Z.S., Yager, R.R.: Some geometric aggregation operators based on intuitionistic fuzzy sets. Int. J. General Syst. **35**(4), 417–433 (2006)
25. Yager, R.R., Kacprczyk, K.: The Ordered Weighted Averaging Operators: Theory and Applications, pp. 275–294. Kluwer Academic Publishers Norwell, MA, USA (1997)

26. Yager, R.R.: On ordered weighted averaging aggregation operators in multicriteria decision making. IEEE Trans. Syst. Man Cybern. **18**, 183–190 (1988)
27. Yager, R.R.: Families of OWA operators. Fuzzy Sets Syst. **59**, 125–148 (1993)
28. Yager, R.R.: Quantifer guided aggregation using OWA operators. Int. J. Intell. Syst. **11**(1), 49–73 (1996)
29. Yager, R.R., Filev, D.P.: Parameterized and-like and or-like OWA operators. Int. J. General Syst. **22**(3), 297–316 (1994)
30. Zadeh, A.: A computational approach to fuzzy quantifiers in natural languages. Comput. Math. Appl. **9**, 149–184 (1983)
31. Zadeh, A.: A computational theory of dispositions. Int. J. Intell. Syst. **2**, 39–64 (1987)
32. Zhao, N., Wei, C.P., Xu, Z.S.: Sensitivity analysis of multiple criteria decision making method based on the OWA operator. Int. J. Intell. Syst. **28**(11), 1124–1139 (2013)

Conceptualizing an User-Centric Approach for Context-Aware Business Applications in the Future Internet Environment

Emilian Pascalau and Grzegorz J. Nalepa

Abstract The obvious move toward a Future Internet environment that is distributed, mobile, cloud-based, semantically rich has raised and emphasized the need for different types of applications. The focus of this new type of application can no longer be on the software itself but directly on the relevant needs and goals of end-users. We argue that because these applications are strongly end-user oriented, context and context-awareness play an important role in their design and development. Hence, in this paper, we introduce and discuss a user-centric approach for building context-aware business applications. This approach proposes a new programming model through a composite system where a human user and an intelligent system are interacting with each other. The interaction is via environment and is according to a predefined plan.

Keywords Context-awareness · Business applications · User centrism · Requirements engineering

1 Introduction

We are witnessing an obvious trend toward the Future Internet environment which is essentially cloud-based and which comes with a new set of characteristics and challenges. In this new environment that is generative and fosters innovation [36] business models and customer relationships are changing, it is about software on demand, simple to use, software that takes into account users' needs, it is mobile, runs everywhere, including web browsers.

E. Pascalau (✉)
Conservatoire National des Arts et Métiers, 2 rue Conté, 75003 Paris, France
e-mail: emilian.pascalau@cnam.fr

G.J. Nalepa
AGH University of Science and Technology,
al. A. Mickiewicza 30, 30-059 Krakow, Poland
e-mail: gjn@agh.edu.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_8

A series of new challenges emerge from this new environment. To name just a few: how to design software that is highly end-user oriented and social, so it involves the end-user in the development process. Then how to take advantage of the end-user's knowledge, acquire and then share it. Finally how to design software that can be easily adapted either automatically or by other end-users.

We believe and assert that in this environment which is fundamentally oriented toward end-users, context will play an important role as context greatly influences the way humans or machines act, the way they report themselves to situations and things; furthermore, any change in context causes a transformation in the experience that is going to be lived and sensed [5].

The original contribution of this paper is the conceptualization of an user-centric approach that we consider to be required in order to design and build software systems that tackle context-aware business applications. This approach proposes a new programming model through a composite system where a human user and an intelligent system are interacting with each other. The interaction is via environment and is according to a predefined plan.

This paper is an enhanced version of the paper presented at the KICSS 2013 conference [26]. The rest of the paper is organized as follows. Section 2 presents directions and approaches related to business applications. Section 3 depicts *Slice.com* a motivating use case, end-user oriented. Section 4 is dedicated discussing and presenting our approach. We conclude in Sect. 5.

2 Approaches for Tackling Complexity in Business Applications

Although not explicitly stated, a business application or business software is any set or combination of tools, programs, and paradigms that are used by business users to perform various business tasks to innovate, to increase productivity, to measure productivity, to monitor, perform business functions accurately and so forth. If we look for instance at Google's business apps web site[1] we have an immediate confirmation of our statement. Moreover we see that even email is considered a business application, and even an important one, since for some business activities, receiving an email is a triggering event for starting a business process (see for instance [6]).

The move to the Internet as a platform and the shift from transaction-based Web pages to interaction-based ones opened the way to a whole new environment. Future Internet is a generative environment that fosters *innovation*, through an advance of technologies and a *shift in how people perceive, use and interact with it* [36]. Future Internet *is by and for the people*, it is about *services* and *things*, it is about *knowledge* and *contents*. This process included both consumers, businesses as well as governments around the world.

[1] http://www.google.com/enterprise/apps/business/, accessed 29 September 2014.

According to a survey by the Economist [32] from 2007, 31 % of participating companies think that the usage of the web as platform will affect all parts of their business and 38 % of companies expect to use *Web 2.0* tools and methods to boost revenue through customer acquisition. *Corporate winners and losers will be designated simply by who figures out how to use the network* [32].

Unlike traditional technologies such as ERP and CRM where users mostly process information in the form of reports or execute transactions, Web 2.0 technologies are *interactive* and *require users to generate new information and content or to edit the work of other participants* [7].

This new ecosystem requires more *intelligent tools* and means to tackle: huge amount of information; data coming from different sources; users with a low level of technological background but who have a lot of ideas, and who have to resolve by themselves different business tasks, see [19, 33]. Business Intelligence (BI) for the new economy requires *people* to take active part and do by themselves BI development tasks from within their *browsers* [19]. In terms of research, the main directions dealt with *business rules* [20, 24], *business processes*, *web services*, *semantic web services* and *service oriented architectures (SOA)*,[2] [17]. In the last 15 years business rules were employed to declaratively describe policies, business processes and practices of an enterprise. Applications in domains such as insurance, financial services, government, telecom, and e-commerce benefit greatly from using rule engines. Moreover, rules are becoming increasingly important in business modeling and requirements engineering, as well as in Semantic Web applications. Over time several rule languages have been defined as well as techniques to perform interchange between them [13].

A lot of work and effort has been put also in the topic of *workflows*, *workflow management*, *business processes* and *business process management* (BPM). As stated in [35] a "business process consists of a set of activities that are performed in coordination in an organizational and technical environment". Their all together scope is the fulfillment of a business goal. BPMN is the de facto standard for modeling business processes. Some of the most important directions of research in terms of business processes comprise management of process variants [15] and workflow mining with the purpose of discovering process models from event logs [1]. There are also approaches that tackle the natural combination between business rules and business processes, e.g., [18].

Service Oriented Architecture (SOA) is a flexible, standardized architecture that facilitates the combination of various applications into inter-operable services.

A *Web service* can be defined as a loosely coupled, reusable software component that encapsulates discrete functionality, which may be distributed and programmatically accessed. A web service is a service that is accessed using standard Internet and XML-based protocols. A more general definition given by Lovelock [21] "a service is an act or performance offered by one party to another." Although the process may be tied to a physical product, the performance is essentially intangible and does not normally result in ownership of any of the factors of production.

[2] https://www.oasis-open.org/committees/soa-rm/, accessed 29 September 2014.

However, in reality Web services have hardly been adopted beyond the boundaries of enterprises [10]. Hence, companies and institutions started to realize that allowing people to use the data they hold could unveil additional value with very little investment. Therefore companies started to open and allow usage of their data by mere customers which gave birth to what is called the Web of Data or Linked Data [4]. Linked Data as stated in [4] refers to a set of best practices for publishing and connecting structured data on the Web. Nowadays the Web has evolved from a global space of linked documents to one where both documents and data are linked together. The *Semantic Web* [3] or the web of Linked Data adds formal knowledge representation such that intelligent software could reason with the information.

To sum up, what we find out is that enterprise software *is the glue that ties together teams and business processes* [23], and *that enterprise systems are simplifying, webifying, mobilizing, and getting a lot more social*. These changes bring in, both, challenges and benefits. For the context-aware community users have been the fundamental motivation (see for instance [9, 16, 34] etc.). *Context* greatly influences the way humans or machines act, the way they report themselves to situations and things; furthermore any change in context, causes a transformation in the experience that is going to be lived, sensed [5]. As argued in the previous paragraphs the future business applications are also changing into being strongly user-centric. Therefore the binding link, the gluing agent, between business applications on one side and context-awareness on the other side, are the users and a switch to a more social business environment.

In addition, the concept of a business user changes. The massive adoption of *cloud computing* as a platform for business applications, that fosters online *collaboration*, *crowd sourcing*, *wisdom of the crowds* and other social aspects of using business apps, has absorbed in the business processes also the customers and simple users. They are part of the processes and their knowledge and input is important. In addition web-based business apps become accessible to a ever growing user base of entrepreneurs, startups companies. These require flexibility and a rapid adaptation of applications to user's needs. As such the step toward context-aware business applications is a natural one.

3 Motivating Example

Forbes enumerates in "10 Brilliant Apps Small Businesses Should Use"[3] a list of ten applications that have emerged from the Future Internet generative environment. The fundamental characteristic of this applications is that they are user-centric. In the sense that they focus on resolving a particular problem that is of interest for the end users.

[3] http://www.forbes.com/sites/ilyapozin/2012/05/29/10-brilliant-apps-small-businesses-should-use/, accessed 29 September 2014.

Slice.com is such a Future Internet specific project that uses emails to tackle a very specific end-user related problem. Slice is an online purchase management tool that gets hooked into user email account. Whenever a new email is received Slice automatically analyzes the content of the email. If the email contains order information from one of user's online shops, then Slice via pattern-based recognition techniques extracts order related contextual information and organizes this information for the user. Hence all of user's purchases will be gathered in one place, the user will be able to keep track of his shopping history, amount of money that he/she spends, type of products, time related information i.e. when a shipment is about to arrive and so forth.

We analyze from an end-user perspective what this use case is about.

- *Problem faced by users*: keeping track of the purchases made online.
- *Applications involved, Services*: Email as a legacy system; Services: online shops (Amazon, EBay), shipment services (FedEx, UPS); Geolocation services (Google Maps); other type of services i.e. topic extraction
- *Concepts*: shop, service, user, invoice, email, time and date, amount of money, product, type of product, location, address, tracking number. The list of concepts is not exhaustive, and is up to the user.
  Concepts are all those entities that are used in taking decisions and / or involved in any way in the process of resolving the end-user's problem.
- *Context*: For example, one context from the perspective of an end-user in the Slice use case, could comprise: a specific service such as FedEx; concepts associated with it, i.e., shipment, location, and address. Further more interaction related to this specific context could be provided, as what to do with this information and so forth.

Figure 1 depicts a general interaction process with respect to this use case.

As we have already argued in [27] applications tackling similar use case, and which are user-centric are required to provide at least the following capabilities:

- provide the means and allow the end-user to model and represent context;
- allow the modeling of relationships between context elements;
- allow the modeling of interactions between different contexts, this implies both in the form of conditions and sequences of events and actions (more precise business rules and business processes)
- based on the provided models have the capabilities to discover in the environment the modeled context(s)

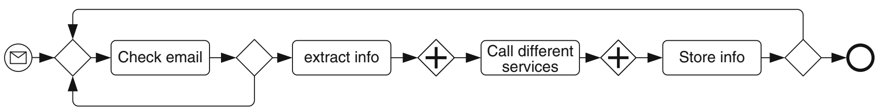


Fig. 1 Interaction process to solve the problem of keeping track of online purchases

- sense events and actions that are performed in the system
- perform actions according to models defined.

We believe that this example supports our statements from Sect. 1 and that it underlines the challenges and characteristics of business applications for the Future Internet environment. Moreover, it also emphasizes the need for context and context-awareness. On the other hand it also confirms that the notion of context is a vague notion and that the context notion is actually strongly connected to each user and to a specific problem.

4 User-Centric Conceptual Approach for Context-Aware Business Applications

So far we have discussed and underlined aspects, research directions and technologies related to business applications. We have further addressed a Future Internet specific example. We introduce in this section a user-centric conceptual approach for context-aware business applications that takes into account the assertions that ended the preceding section. We argue that using such an approach applications similar to the use case we discussed could be built by end-users.

Thus, such systems should be end-user oriented. Such systems should allow a new programming model for composite systems (humans + services). We argue that such systems are intelligent systems being able to interact with the end-user according with an agreed beforehand plan, supporting evolution, sharing and distribution. In consequence these systems are two layer systems: one high level layer, that deals with the problem at a conceptual and semantic level (the agreed on plan) and one low level layer that deals with the internals of the system and low level technologies i.e. direct access to services, etc. The low level layer should be hidden as much as possible from the end-user.

Therefore the aspects (also depicted in Fig. 2) that drive the development of our approach are: end-user oriented or user-centric; human user and intelligent system interaction; plan; two-layer system; intelligent system. These aspects have been previously discussed in the research literature, however almost always in a disconnect manner with almost no interaction with each other. In addition different

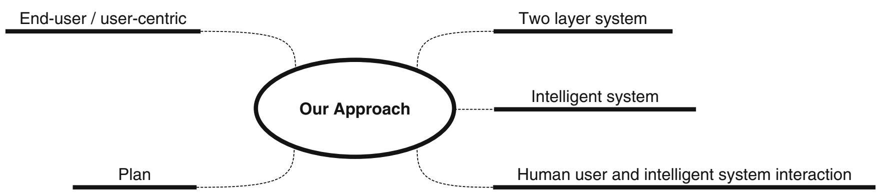


Fig. 2 Aspects that drive our approach

terminology, according to the research directions where it has been studied, has been used to identify actually the same concept.

The *end-users* whom we want to support, are most of them, *not professional developers*, who lack technical skills, and don't have the necessary knowledge to write software programs, according to technical specifications (specifications for Web Services, APIs, REST etc.). End-users have a variety of *goals*. These goals are achieved by *creating* new applications, via mashing up existing applications, or software artifacts, by *modifying* or *adapting* existing applications.

Now in order to allow end-users, who are not professional programmers to program, adapt existing applications or software artefacts according to their needs the technical layer needs to be hidden as much as possible, otherwise they will not be able to do it.

To achieve this separation of concerns and thus hiding the technical layer we argue that a *two layer system* is required. The high level layer should provide the means to allow both human users and the system to understand each other using a common set of concepts and following a beforehand agreed on *plan*. The low level layer on the other hand should be hidden from the end-user and should be accessed directly by the system according to what has been agreed upon in the *plan*. A professional developer should also be allowed access to this layer.

Software Engineering (SE) is the field of study that is concerned with all the aspect related to the design and development of software systems. Two of these aspects: Requirements engineering and system design are of interest for our approach because system design targets the internal behavior of the system while requirements are external, concerning the world.

The phase of gathering requirements precedes system design. Unfortunately in most of the situations the final product does not actually comply with the end-user expectations for various reasons: i.e., bad communication, different understanding of concepts, and situations, etc. [25, 31]. An end-user perceives and understands a software system through the user interface (UI). Based on the UI, which is supposed to be an accurate and complete representation of the system, the end-user builds its own understanding of the environment consisting of concepts, with which it is working, and associated behavior [8]. Haplessly in many situations the end-user's understanding is very different than the understanding and the message, developers actually tried to convey [31].

A requirement in Requirements Engineering (RE), as stated in [30], "defines both needs and goals of users, and conditions and properties of the system to be developed, that result for example, from organizational needs, laws, or standards."

In RE a goal is a stakeholder's intention with regard to the objectives, properties or use of the system [30]. Requirements are said to define *what* should be developed while system design defines *how* the system should be developed [30].

In consequence, we debate that RE stands for the high level layer and system design stands for the low level layer. And therefore, we believe that both RE and system design need to be unified when dealing with proper intelligent user-centric systems (see Fig. 3).

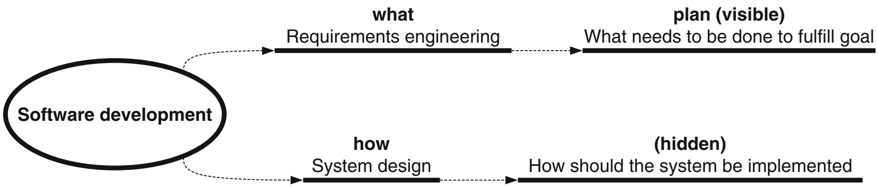


Fig. 3 Software development aspects

Requirements Engineering approaches propose also the use of scenarios, which are concrete positive or negative examples of satisfying or failing to satisfy a goal or a set of goals [30]. Such scenarios for our approach are the *plans* that end-users create. Software requirements are usually expressed in natural language. However, natural language can not be an option here. We need a structured and well defined set of concepts to express end-user plans, such that an intelligent system can understand, reason and use that plan, in its interaction with the environment and the end-user. Because this plan needs to be from the perspective of end-user we looked first to research domains such as Human Computer Interaction (HCI), conceptual design and cognitive psychology.

In these domains, such a user defined plan is assimilated to the notion of a mental model [8]. Such plans, in cognitive psychology have been conceptualized as "representations in episodic memory of situations, acts or events spoken or thought about, observed or participated in by human actors" [11]. According to [22] such a plan consist of several parts:

- *an image*: if the mental model refers to a physical object then the model should contain a simplified image of the object;
- *a script*: if the mental model refers to a process, it should contain a description of that process;
- *a set of related mental models*: mental models can be composed of other mental models;
- *a controlled vocabulary*: each mental models has a set of key definitions and variants;
- *a set of assumptions*: allow users to predict behavior.

Translating this description into our context, an end-user plan contains a set of concepts, with which the end-user works with in order to fulfill a goal, their relationship with each other, the context; and means to express behavior in the forms of processes and rules. Hence, for our approach, the user has a plan of *what* needs to be done in order *to achieve* a *goal*. *Needs to be done* here means the *interaction (behavior)* that a user needs to exhibit with the application (applications) in order to fulfill the goal. Moreover the user is expecting a particular answer from the system as a response to the actions he/she, the user, performs. User behavior is imposed (influenced) by the *context (environment)*.

Traditionally, context has been perceived in computer science community as a matter of location and identity, see for instance [2]. However, interaction and problems concerning interaction require more than just the environmental context (location, identity) used traditionally in context-aware systems [14]. As such lately the notion of context has been considered not simply as state but as part of a process in which users are to be involved [9]. Fischer, however gives a definition that takes into account the human-centered computational environments. Context is defined in [12] as being *the 'right' information, at the 'right' time, in the 'right' place, in the 'right' way to the 'right' person*.

Context greatly influences the way humans or machines act, the way they report themselves to situations and things; furthermore any change in context, causes a transformation in the experience that is going to be lived, sensed [5]. Many psychological studies have shown that when humans act, and especially when humans interact, they consciously and unconsciously attend to context of many types as stated in [14]. Nardi underlines this aspect clearly in [25] stating that "we have only scratched the surface of what would be possible if end users could freely program their own applications… As has been shown time and again, no matter how much designers and programmers try to anticipate and provide for what users will need, the effort always falls short because it is impossible to know in advance what may be needed… End users should have the ability to create customizations, extensions and applications…"

For the approach that we envision and design in this paper, the user creates a *plan* that is shared (given) to the system. This plan contains a description of the *context(s)* and the *behavior* that both the human user and the system need to perform in relationship with the context(s) as we introduced it in [28]. The plan explains how should the system react in response to the actions that the human user performs, or how the system should react in response to changes that appear in the environment (context(s)). In this way, both the human user and the system will follow and will share the same understanding, the same plan. Petrelli et al. emphasize this [29] by stating that the main objective of context should be to make technology invisible for the user, hence the focus will not be anymore on how to use the technology in a proper way but the focus will be once again on resolving our activities and achieving our goals in daily business activities and not only. To achieve this objective will require a completely and reliable world representation; a shared understanding of concepts [29]. For our design we adhere to the ideas discussed in [5] and we argue that the world representation needs to be from an individual's perspective. Such an individual representation of the world needs to be a subset of the entire world that a system can understand.

The shared plan needs to be represented in a way such that both the human user and the system be able to use it and reason about it, and then reasoning techniques are required to reason about it. Hence, to represent the plan we subscribe to the current approaches for semantic web, by means of ontologies.

The last two aspects that we identified concerning our approach are: *intelligent system* able to *interact with the human user* according to the agreed plan.

We link together the aspects we just discussed and we resume our approach in Fig. 4. The approach we have introduced in this section proposes a new programming model through a composite system where human user and system interacting with

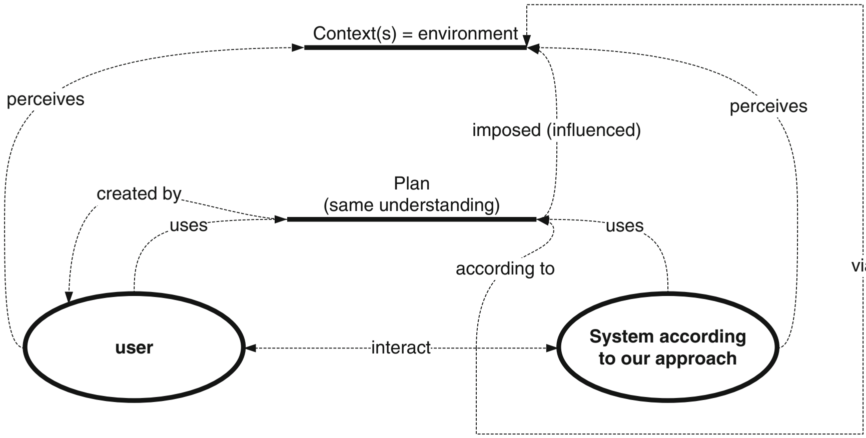


Fig. 4 Our approach

each other. The interaction is via environment and is according to a predefined plan. This plan is created by the human user, hence the human is the coordinator of how the system works. Both the intelligent system and the human user follow the same plan. Such a plan serves to achieve a particular goal and it has been created taken into account the context(s). Interaction is perceived via changes that appear in the environment.

5 Conclusions

In the Future Internet environment business applications are different and they need to comply with different challenges than *legacy systems*. For this new type of applications an end-user orientation is an essential characteristic.

We have introduced and discussed in this paper a new user-centric conceptual approach for building context-aware business applications. This approach proposes a new programming model through a composite system where a human user and an intelligent system are interacting with each other. The interaction is done via environment and is according to a predefined plan. This plan is user defined and contains a description of context(s) and behavior associated with this context. Later on such a plan is used both by the human user and the intelligent system in order to achieve a user goal by interacting with each other.

This paper presents results of research that is work in progress toward achieving a unified modeling and methodological approach for context-aware business applications, and (2) a unified execution framework of context-aware business applications.

Acknowledgments The paper is supported by the AGH UST Grant 11.11.120.859.

References

1. Aalst, W., Weijters, A., Maruster, L.: Workflow mining: discovering process models from event logs. IEEE Trans. Knowl. Data Eng. **16**(9), 1128–1142 (2004)
2. Abowd, G.D., Dey, A.K.: Towards a better understanding of context and context-awareness. In: Gellersen, H.-W. (ed.) Handheld and Ubiquitous Computing. Lecture Notes in Computer Science, vol. 1707, pp. 304. Springer, Heidelberg (1999)
3. Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web, pp. 34–43. Scientific America (2001)
4. Bizer, C., Heath, T., Berners-Lee, T.: Linked data—the story so far. Int. J. Semant. Web Inf. Syst. (2010)
5. Bolchini, C., Curino, C.A., Quintarelli, E., Schreiber, F.A., Tanca, L.: A data-oriented survey of context models. ACM SIGMOD Rec. **36**(4), 19–26 (2007)
6. Burkhart, T., Werth, D., Loos, P.: Context-sensitive business process support based on emails. In: WWW 2012—EMAIL'12 Workshop, April 2012
7. Chul, M., Miller, A., Roberts, R.P.: Six ways to make web 2.0 work. Business Technology, The McKinsey Quaterly, pp. 1–6 (2009)
8. Clark, L., Sasse, M.A.: Conceptual design reconsidered: the case of the internet session directory tool. In: Proceedings of HCI'97 People and Computers XII (1997)
9. Coutaz, J., Crowley, J.L., Dobson, S., Garlan, D.: Context is key. Commun. ACM—The Disappearing Comput. **48**(3), 49–53 (2005)
10. Davies, J., Potter, M., Richardson, M., Stincic, S., Domingue, J., Pedrinaci, C., Fensel, D., Gonzalez-Cabero, R.: Towards the open service web. BT Technol. J. **26**(2), 1694–1719 (2009)
11. Dijk, T.A.: Cognitive context models and discourse. In: Stamenow, M. (ed.) Language Structure, Discourse and the Access to Consciousness, pp. 189–226. Benjamins (1997)
12. Fischer, G.: Context-aware systems—the 'right' information, at the 'right' time, in the 'right' place, in the 'right' way, to the 'right' person. In: AVI'12: Proceedings of the International Working Conference on Advanced Visual Interfaces, pp. 287–294. ACM (2012)
13. Giurca, A., Wagner, G.: Rule modeling and interchange. In: Proceedings of 9th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, SYNASC2007, pp. 485–491. IEEE (2007)
14. Grudin, J.: Desituating action: digital representation of context. Human-Comput. Interact. **16**(2), 269–286 (2001)
15. Hallerbach, A., Bauer, T., Manfred, R.: Capturing variability in business process models: the Provop approach. J. Softw. Maintenance Evol.: Res. Pract. **22**(6–7), 519–546 (2010)
16. Heer, J., Peddemors, A., Lankhorst, M.: Context-aware mobile business applications. In: Proceedings of the First CoCoNet Workshop, Context Aware Collaborative Environments for Next Generation Business, pp. 3–4 (2002)
17. Keen, M., Acharya, A., Bishop, S., Hopkins, A., Milinski, S., Nott, C., Robinson, R., Adams, J., Verschueren, P.: Patterns: Implementing an SOA Using an Enterprise Service Bus. IBM Redbooks (2005)
18. Kluza, K., Kaczor, K., Nalepa, G.J.: Enriching business processes with rules using the Oryx BPMN Editor. In: Proceedings of the 11th International Conference ICAISC 2012. LNCS, vol. 7268, pp. 573–581. Springer (2012)
19. Kobielus, J., Karel, R., Evelson, B., Coit, C.: Mighty mashups: do-it-yourself. Business intelligence for the new economy. Inf. Knowl. Manage. Prof. Forrester (47806), 1–20 (2009)
20. Ligęza, A., Nalepa, G.J.: A study of methodological issues in design and development of rule-based systems: proposal of a new approach. Wiley Interdiscip. Rev.: Data Mining Knowl. Discovery **1**(2), 117–137 (2011)
21. Lovelock, C., Vandermerwe, S., Lewis, B.: Services Marketing. Prentice Hall (1996)
22. McDaniel, S.: What's your idea of a mental model? http://boxesandarrows.com/whats-your-idea-of-a-mental-model/ (2003). Accessed 29 Sept 2014

23. McLellan, C., Hammond, T., Dignan, L., Hiner, J., Gilbert, J., Ranger, S., Gray, P., Kwang, K., Lui, S.: The evolution of enterprise software. ZDNet and TechRepublic. http://www.zdnet.com/topic-the-evolution-of-enterprise-software/ (2013). Accessed 29 Sept 2014
24. Nalepa, G.J., Kluza, K.: UML representation for rule-based application models with XTT2-based business rules. Int. J. Softw. Eng. Knowl. Eng. (IJSEKE) **22**(4), 485–524 (2012). http://www.worldscientific.com/doi/abs/10.1142/S021819401250012X. Accessed 29 Sept 2014
25. Nardi, B.A.: A Small Matter of Programming: Perspectives on End User Computing. The MIT Press (1993)
26. Pascalau, E., Nalepa, G.J.: Understanding context-aware business applications in the future internet environment. In: Skulimowski, A.M.J. (ed.) Looking into the Future of Creativity and Decision Support Systems—Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems. Progress & Business Publishers (2013)
27. Pascalau, E.: Identifying guidelines for designing and engineering human-centered context-aware systems. In: Nalepa, G.J., Baumeister, J. (eds.) KESE 9: Proceedings of 9th Workshop on Knowledge Engineering and Software Engineering Co-located with the 36th German Conference on Artificial Intelligence (KI 2013), vol. 1070. CEUR-WS (2013)
28. Pascalau, E.: Mashups: behavior in context(s). In: KESE 7: Proceedings of 7th Workshop on Knowledge Engineering and Software Engineering at the 14th Conference of the Spanish Association for Artificial Intelligence (CAEPIA 2011), vol. 805, pp. 29–39. CEUR-WS (2011)
29. Petrelli, D., Not, E., Strapparava, C., Stock, O., Zancanaro, M.: Modeling context is like taking pictures. In: CHI'2000 Workshop on Context Awareness. Georgia Institute of Technology (2000)
30. Pohl, K.: Requirements Engineering: Fundamentals, Principles, and Techniques. Springer (2010)
31. Tognazzini, B.: TOG on Interfaces. Addison-Wesley (1992)
32. Unit, T.E.I.: Serious business. web 2.0 goes corporate. Technical report, The Economist (2007)
33. Verizon: Web 3.0: Its promise and implications for consumers and business. White Paper, VerizonBusiness.com. http://www.verizonbusiness.com/resources/whitepapers/wp_web-3-0-promise-and-implications_a4_en_xg.pdf (2010). Accessed 29 Sept 2014
34. Wang, M.: Context-aware analytics: from applications to a system framework. http://e-research.csm.vu.edu.au/files/apweb2012/download/APWeb-Keynote-Min.pdf (2012). Accessed 29 Sept 2014
35. Weske, M.: Business Process Management: Concepts, Languages, Architectures. Springer (2007)
36. Zittrain, J.: The Future of the Internet and How to Stop It. Yale University Press, New Haven and London (2008)

Preprocessing Large Data Sets by the Use of Quick Sort Algorithm

Marcin Woźniak, Zbigniew Marszałek, Marcin Gabryel and Robert K. Nowicki

Abstract Sorting algorithms help to organize large amounts of data. However, sometimes it is not easy to determine the correct order in large data sets, especially if there are special poses on the input. It often complicates sorting, results in time prolongation or even unable sorting. In such situations, the most common method is to perform sorting process to reshuffled input data or change the algorithm. In this paper, the authors examined quick sort algorithm in two versions for large data sets. The algorithms have been examined in performance tests and the results helped to compare them.

Keywords Computer algorithm · Data sorting · Data mining · Analysis of computer algorithms

1 Introduction

Quick sort algorithm is known from [1] or [6]. In classic method, described also in [3, 18, 20–22], we sort elements indexed i_{left}, i_{left-1}, …, $i_{right-1}$, i_{right}. All the elements are placed according to the order on both sides of the one reference element, chosen to divide them. A reference element index is an arithmetic average of indexes

M. Woźniak · Z. Marszałek (✉)
Institute of Mathematics, Silesian University of Technology,
ul. Kaszubska 23, 44-100 Gliwice, Poland
e-mail: Zbigniew.Marszalek@polsl.pl

M. Woźniak
e-mail: Marcin.Wozniak@polsl.pl

M. Gabryel · R.K. Nowicki
Institute of Computational Intelligence, Czestochowa University of Technology,
al. Armii Krajowej 36, 42-200 Czestochowa, Poland
e-mail: Marcin.Gabryel@iisi.pcz.pl

R.K. Nowicki
e-mail: Robert.Nowicki@iisi.pcz.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_9

in the form of lower minimum total value, for more details please see [1, 6, 18, 22]. Unfortunately, the cost of calculating this element index is determined by the number of elements in the sequence. This slows down sorting of large data sets. An additional complication is the fact, that in practice there are often sequences merely sortable for classic version. Therefore one may often face some complications. These difficulties encourage looking for an optimal solution. Authors of [4, 10, 16, 20] describe specific features of quick sort. Moreover, the authors of [5, 7, 8, 11, 12, 21] or [20, 23] pay attention to impact on memory management or practical application of sorting algorithms. At the same time the authors of [16, 19–21] present results of research on increasing efficiency. Authors of this paper present some important research on quick sort performance for large data sets, more details can be found in [24, 26]. In Sect. 2 we have examined the quick sort to make it faster and more stable for large data sets. The results, presented in Sect. 2.2, suggest that there are some aspects of this method that may help to make it more stable and faster. For special poses of big data collections classic version may be interfered. It is also possible to face time prolongation or stack overflow, which will result in unexpected breaks.

1.1 Killers—Special Poses on the Input

Quick sort algorithm, in classic form described in [3, 6, 22], may behave erratically or overflows stack for adverse poses of the input elements when adapted to large data sets. The authors of this study present some special poses that may slow down sorting, some other aspects of "killing" poses are also described in [10, 26]. Let us consider the case of some possible arrangements of elements for quick sort, that in practice come very often (for more details see [26]). Some very interesting, examined in this paper examples are presented in Fig. 1. Classic version of quick sort is very often unable to sort a large sequence, if it is in one of killing poses, as confirmed studies described in Sect. 2.2, for more details see also [26]. Most unfavorable arrangement of elements can be described in the form of formulas.

Series $\{a_i\}$, where $i = 0, 1, \dots, n$ of odd number of n elements is in unfavorable pose, when:

1. Middle element and the following ones up to the last one have even values, what means that $a\left[\frac{n+1}{2}\right] = k$, $k = 2, 4, 6, \dots, n$ and $i = 0, 1, 2, 3, \dots, \frac{n}{2}$.
2. Elements from first to middle one have odd values, what means that $a[i] = k$, $k = 1, 3, 5, 7, \dots, n$ and $i = 0, 1, 2, 3, \dots, \frac{n}{2}$.

Series $\{a_i\}$, where $i = 0, 1, \dots, n$ of even number of n elements is in unfavorable pose, when:

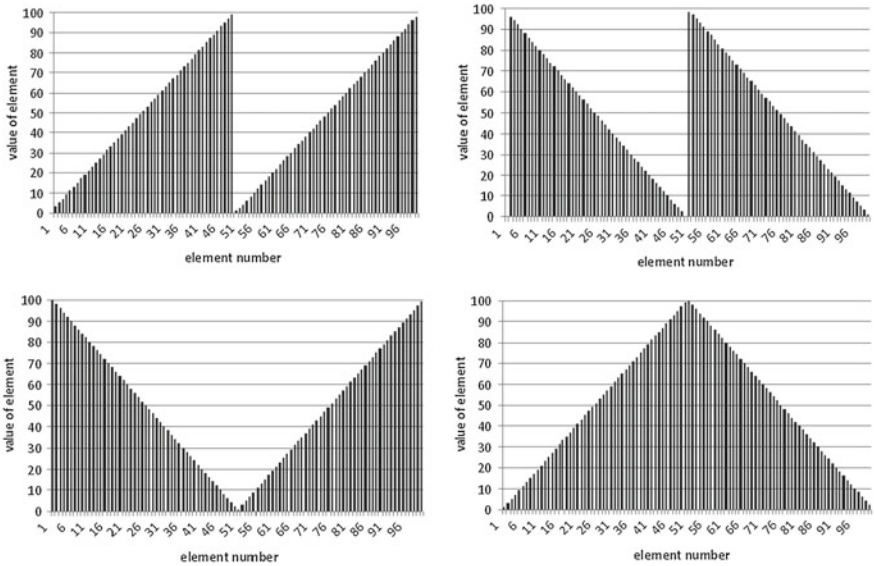


Fig. 1 Example of examined “killing” poses for quick sort

1. Middle element is $a\left[\frac{n}{2}\right] = 1$.
2. Elements after middle one $a\left[\frac{n}{2}\right]$ have following even values starting from 2, what means that $a\left[\frac{n}{2} + i\right] = k$, where $k = 2, 4, 6, \ldots, n$ and $i = 0, 1, 2, 3, \ldots, \frac{n}{2}$.
3. First element has a value $n - 1$, what means that $a[0] = n - 1$.
4. Elements ranging from first to middle have odd values starting from 3, what means that $a[i] = k$, where $k = 3, 5, 7, \ldots, n - 1$ and $i = 0, 1, 2, 3, \ldots, \frac{n}{2}$.

Presented unfavorable poses cause flipping so many elements, that for large data sets sorting may be even not possible. Research showed that above 100,000 elements stack overflow may occur and sorting is stopped. The practice shows that such situation comes frequently. Discussed in [3, 6, 22] or [24, 26] classic version of the algorithm may be unable to perform the sorting in case of one of presented in Fig. 1 arrangements of elements. However in [4, 10, 16] or [20] are shown an interesting quick sort optimizations. Authors of [2, 9, 14–16, 23] present opportunity to improve the algorithm for the adverse poses of input. Moreover the authors of [4, 16, 20, 23] indicate the possibility of using multi-threading. Authors of [13, 19] show the possibility to apply special selection strategy. Therefore, the authors of the present study examined described changes to accelerate the quick sort algorithm, regardless of the content of the input data or incoming pose, for more details see also [24, 26]. Here, the algorithm was examined in classic version and in changed one, so that in each iteration a reference element was chosen at random. This change reduces possibility of any complication or time prolongation. At the same time possibility of stack overflow and crash practically is reduced to minimum.

2 Research and Examination

Let us first describe modified quick sort algorithm. If the reference element is taken randomly, we do not deal with the cost of calculating reference element index. This makes quick sort algorithm more stable, as shown performed studies and tests see [24, 26] or [27]. This makes it applicable in NoSQL database systems what was presented in [12]. Stability of the algorithm, we understand it as a repetition of performance (sorting time or CPU usage) in performed experiments. In the research, we used modified quick sort shown in Algorithm 1. Some other quick sort versions dedicated for large data sets are also presented in [24, 26].

```
Start
Load data
Pass a pointer to an array
Calculate index of left element
Calculate index of right element
if left < right then
    Set index t as drawn number from the scope from left to right
    Set temp as a[t]
    Set element a[t] as a[right]
    Set last as right
    Set index i as right − 1
    while index i ≥ left do
        if element a[i] > temp then
            Decrease index last − 1
            Swap element a[last] with a[i]
        end
        Decrease index i = i − 1
    end
    Set element a[right] as a[last]
    Set element a[last] as temp element
    Proceed algorithm with index left and index last − 1
    Proceed algorithm with index last + 1 and index right
    Stop
else
    Return
end
```

Algorithm 1: Quick sort with random reference element

2.1 *Quick Sort Algorithm with Random Reference Element Selection—Time Complexity Discussion*

Theorem 1 *Examined algorithm sorts n element sequences in* $T_{avg}(n)$ *average time*

$$\vartheta\left(n \cdot \log_2 n\right). \quad (1)$$

Proof At the beginning let us assume that for a constance B, it is $T_{avg}(0) = T_{avg}(1) = B$. We take randomly element i from table with sorted string. Then average sort time is $T_{avg}(i-1)$ and $T_{avg}(n-i)$. Index i can be taken with equal probability and sorting takes $C \cdot n$ for a constance C of basic computing operation time. Therefore, we have formula

$$T_{avg}(n) = C \cdot n + \frac{1}{n} \cdot \sum_{n}^{i=1} \left[T_{avg}(i-1) + T_{avg}(n-i)\right], \text{ for } n \geq 2. \quad (2)$$

Reducing elements of the sum in (2) we have

$$T_{avg}(n) = C \cdot n + \frac{2}{n} \cdot \sum_{n-1}^{i=0} T_{avg}(i), \text{ for } n \geq 2. \quad (3)$$

Then applying mathematical induction to n, for $n \geq 2$ we have formula

$$T_{avg}(n) \leq K \cdot n \cdot \log_2 n, \quad (4)$$

where K is $K = 2B + 2C$. For $n = 2$ we have

$$T_{avg}(n) \leq 2C + 2B = 2(C+B) \cdot \log_2 n = K \cdot \log_2 n. \quad (5)$$

Using inductive assumption and formula (4) we can estimate average sort time

$$T_{avg}(n) \leq C \cdot n + \frac{4B}{n} + \frac{2}{n} \cdot \sum_{n-1}^{i=0} K \cdot i \cdot \log_2 i. \quad (6)$$

Next, we can evaluate sum from formula (6) as

$$\sum_{n-1}^{i=0} K \cdot i \cdot \log_2 i = K \cdot \sum_{n-1}^{i=0} i \cdot \frac{\ln i}{\ln 2} \leq \frac{K}{\ln 2} \cdot \int_{n-1}^{2} i \cdot \ln i \, \mathrm{d}i \leq \frac{K}{\ln 2} \cdot \int_{n}^{2} i \cdot \ln i \, \mathrm{d}i. \quad (7)$$

In formula (7), we can replace integral

$$\int_{n}^{2} i \cdot \ln i \, \mathrm{d}i = \frac{n^2}{2} \cdot \ln n - \frac{1}{4}n^2 - \frac{2^2}{2} \cdot \ln 2 + \frac{1}{4}2^2 == \frac{n^2 \cdot \ln n}{2} - \frac{n^2}{4} - 2 \cdot \ln 2 + 1. \quad (8)$$

Therefore formula (8) is

$$\int_{n}^{2} i \cdot \ln i \, \mathrm{d}i \leq \frac{n^2 \cdot \ln n}{2} - \frac{n^2}{4}. \quad (9)$$

Formula (7) is then

$$\sum_{n-1}^{i=0} K \cdot i \cdot \log_2 i \leq \frac{K}{\ln 2} \cdot \left(\frac{n^2 \cdot \ln n}{2} - \frac{n^2}{4} \right). \tag{10}$$

Placing formula (10) into (6), we have

$$T_{avg}(n) \leq C \cdot n \cdot + \frac{4B}{n} + \frac{2}{n} \cdot \frac{K}{\ln n} \cdot \left(\frac{n^2 \cdot \ln n}{2} - \frac{n^2}{4} \right). \tag{11}$$

From formula (11), we have

$$T_{avg}(n) \leq C \cdot n \cdot + \frac{4B}{n} + \frac{2}{n} \cdot \frac{K}{\ln n} \cdot \frac{n^2 \cdot \ln n}{2} - \frac{2}{n} \cdot \frac{K}{\ln n} \cdot \frac{n^2}{4}. \tag{12}$$

We can group elements of (12)

$$T_{avg}(n) \leq C \cdot n \cdot + \frac{4B}{n} + \frac{K \cdot n \cdot \ln n}{\ln 2} - \frac{K}{\ln 2} \cdot \frac{n}{2}. \tag{13}$$

Then, using logarithmic function in (13) we have

$$T_{avg}(n) \leq C \cdot n \cdot + \frac{4B}{n} + \frac{K}{\ln 2} \cdot \frac{n}{2} + K \cdot n \cdot \log_2 n. \tag{14}$$

We evaluate $C \cdot n \cdot + \frac{4B}{n} + \frac{K}{\ln 2} \cdot \frac{n}{2}$ using assumption that for $n \geq 2$ we have $K = 2C + 2B$

$$C \cdot n \cdot + \frac{4B}{n} - \frac{K}{\ln 2} \cdot \frac{n}{2} = C \cdot n \cdot + \frac{4B}{n} - B \cdot \frac{n}{\ln 2} - C \cdot \frac{n}{\ln 2}. \tag{15}$$

In formula (15), we can group elements $C \cdot n \cdot \left(1 - \frac{n}{\ln 2}\right) + B \cdot \left(\frac{4}{n} - \frac{n}{\ln 2}\right)$. Moreover for $n \geq 2$ we can write down $\left(1 - \frac{n}{\ln 2}\right) \approx 0$ and $\left(\frac{4}{n} - \frac{n}{\ln 2}\right) \approx 0$.

$$C \cdot n \cdot + \frac{4B}{n} - \frac{K}{\ln 2} \cdot \frac{n}{2} = C \cdot n \cdot \left(1 - \frac{n}{\ln 2}\right) + B \cdot \left(\frac{4}{n} - \frac{n}{\ln 2}\right) \approx 0. \tag{16}$$

Placing formula (16) in (13) we finally have

$$T_{avg}(n) \leq K \cdot n \cdot \log_2 n. \tag{17}$$

□

Similar theoretical complexity represents other versions of quick sort algorithm dedicated for large data sets, for details please see [26].

2.2 Results of Examinations

Let us now present results of examinations. The algorithms were tested to evaluate performance for unfavorable poses of large data sets. The examined methods were implemented in CLR C++ in MS Visual Studio 2010 Ultimate. To test were taken random samples of 100 series in each class of frequencies, including unfavorable positioning presented in Fig. 1. Tests were performed on a quad core amd opteron processor 8356 8p. During the tests there were no difficulties associated with sorting special collections described in Sect. 1.1. At the beginning let us compare theoretical time complexity. In Figs. 2 and 3 charts are showing time complexity on basis of Theorem 1 for the general arrangement of elements in a sorted set and for the arrangement of elements impossible to sort described in Sect. 1.1. Comparison of complexity in Figs. 2 and 3 shows that unfavorable positioning does not significantly increase complexity of examined method. However, the classic form of the algorithm is not able to organize such collection, if it is more numerous than 100,000

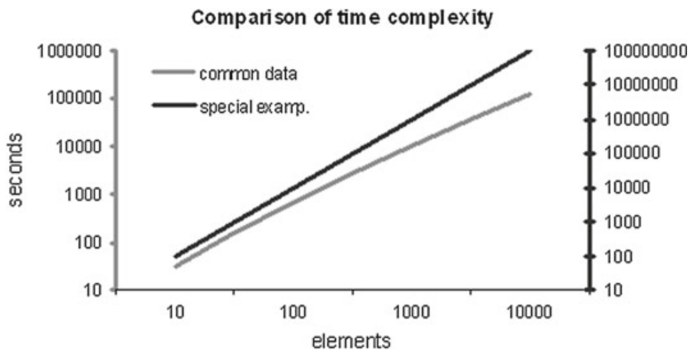


Fig. 2 Comparing theoretical time complexity

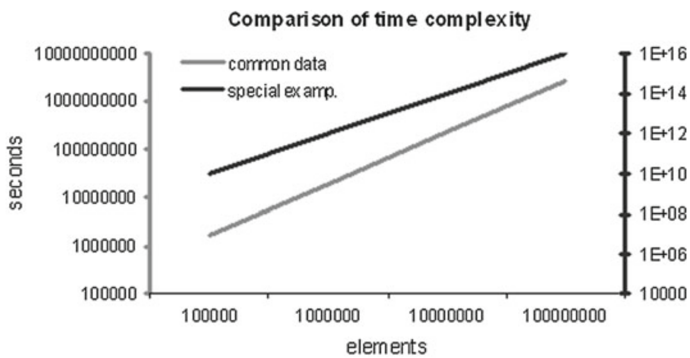


Fig. 3 Comparing theoretical time complexity

Table 1 Table of CPU clock cycles of the examined quick sort algorithm

CPU tics [ti]	Number of sorted elements				
	100	1000	10,000	100,000	1000,000
Avg	1439	7143, 2	210889, 4	19607490, 4	1936698906
Std deviation	51, 67	231, 88	18888	80952, 85	15481790, 82
Avg deviation	44, 8	180, 96	16268, 72	65988, 88	11949578, 72
Coef. of variation	0, 035	0, 033	0, 089	0, 004	0, 008
Var. area upper end	1387, 33	6911, 32	192001, 4	19526537, 55	1921217116
Var. area lower end	1490, 672043	7375, 081867	229777, 4005	19688443, 25	1952180697

elements. Table 1 presents statistical study of CPU clock cycles. Comparison of CPU clock cycles is shown in Fig. 4. Analyzing Fig. 4 we see that classic version can be less processor loading. Unfortunately it is not able to manage unfavorable collection of data, including specific layouts of the elements as described in Sect. 1.1. Figure 5 shows comparison of standard deviation for CPU clock cycles. The changes increase stability and allow to sort any arrangement of input elements. This property is important when strings are longer than 100,000 elements. Analysis and comparison charts of standard deviation presented in Fig. 5 shows that for sorting sets of less than 1000 elements classic version is faster. Sorting collection of over 100,000 elements can be more stable using method examined and discussed in Sect. 2. Comparison of coefficient of variation is shown in Fig. 6. Table 2 shows comparison of aggregate volatility of CPU clock cycles. The values of variability of CPU clock cycles are shown in Fig. 6. Classic version may be more volatile than examined algorithm. This confirms thesis formulated in previous sections. At the same time, analyzing the chart, we see that examined method remains stable for large collections and unfavorable input poses.

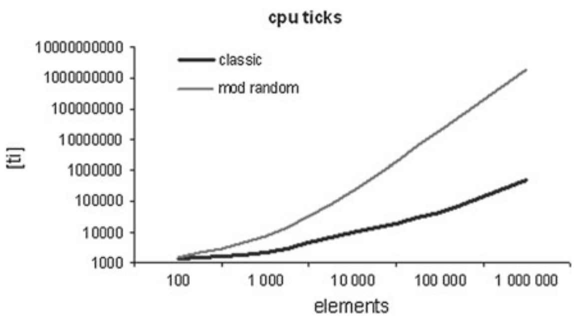


Fig. 4 Comparing the values of the characteristics of CPU clock cycles

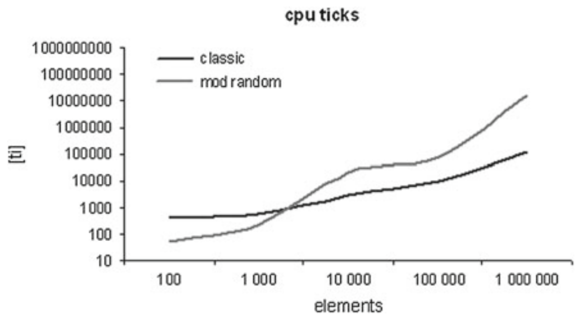


Fig. 5 Comparing the values of standard deviation of CPU clock cycles

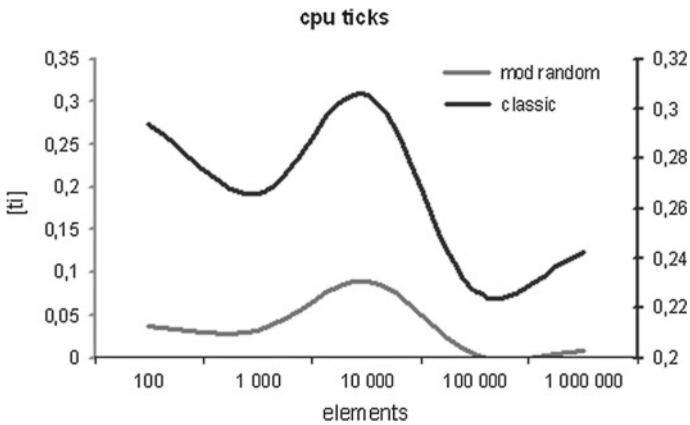


Fig. 6 Comparing the values of expected variability of CPU clock cycles

Table 2 Comparative table of CPU clock cycles variations of the classic and the examined algorithm

CPU tics [ti]	Number of sorted elements				
	100	1000	10,000	100,000	1000,000
Classic	0, 3	0, 27	0, 3	0, 23	0, 24
Mod random	0, 035	0, 033	0, 09	0, 004	0, 008

3 Final Remarks

In conclusion, examined modified method allows to sort large data sets and increases stability of the algorithm. It also allows to sort unfavorable input sequences without complications. As a result, sorting is stable for large data sets and work is more efficient. Thus, the examined algorithm appears to be appropriate for large data sets, even these described in Sect. 1.1. As presented in [12, 26, 27] these kind of algorithms can efficiently work in NoSQL database systems. Therefore, the authors consider increasing efficiency of sorting algorithm for large data sets in further research and development.

References

1. Aho, I.A., Hopcroft, J., Ullman, J.: The design and analysis of computer algorithms. Addison-Wesley Publishing Company, USA (1975)
2. Bing-Chao, H., Knuth, D.E.: A one-way, stack less Quick sort algorithm. BIT **26**, 127–130 (1986)
3. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms. The MIT Press, Cambridge (2001)
4. Francis, R.S., Pannan, L.J.H.: A parallel partition for enhanced parallel quick sort. Parallel Comput. **18**(5), 543–550 (1992)
5. Gedigaa, G., Duntschb, I.: Approximation quality for sorting rules. Comput. Stat. Data Anal. **40**, 499–526 (2002)
6. Knuth, D.E.: The Art of Computer Programming. Addison-Wesley, USA (1998)
7. LaMarca, A., Ladner, R.E.: The influence of caches on the performance of sorting. In: Proceedings of ACM-SIAM Symposium on Discrete Algorithms, pp. 370–379 (1997)
8. Larson, P., Graefe, G.: Memory management during run generation in external sorting. In: Proceedings of SIGMOD, pp. 472–483 (1998)
9. Larson, P.: External sorting: run formation revisited. IEEE Trans. Knowl. Data Eng. **15**(4), 961–972 (2003)
10. MacIlroy, M.: A killer adversary for quick sort. Softw. Pract. Exp. **29**, 1–4 (1999)
11. Marszałek, Z., Połap, D., Woźniak, M.: On preprocessing large data sets by the use of triple merge sort algorithm. In: Proceedings of International Conference on Advances in Information Processing and Communication Technology, The IRED Digital Seek Library (2014) (accepted—in press)
12. Marszałek, Z., Woźniak, M.: On possible organizing Nosql database systems. Int. J. Inf. Sci. Intell. Syst. **2**(2), 51–59 (2013)
13. Martynez, C., Roura, C.: Optimal sampling strategies in quick sort and quick select. SIAM J. Comput. **31**(3), 683–705 (2002)
14. Pan, Y., Hamdi, M.: Quick sort on a linear array with a reconfigurable pipelined bus system. Working Paper 94–03, The Center for Business and Economics Research, University of Dayton (1994)
15. Raghaven, P.: Lecture Notes on Randomized Algorithms. Technical Report, IBM Research Division, New York (1990)
16. Rashid, L., Hassanein, W.M., Hammad, M.A.: Analyzing and Enhancing the Parallel Sort Operation on Multithreaded Architectures. J. Supercomput. Springer, Berlin Heidelberg (2009)
17. Rauh, A., Arce, G.R.: A fast weighted median algorithm based on quick select. In: Proceedings of IEEE International Conference on Image Processing, pp. 105–108. Hong Kong (2010)
18. Sedgewick, R.: Implementing quick sort programs. Commun. ACM **21**(10), 847–857 (1978)
19. Trimananda, R., Haryanto, C.Y.: A parallel implementation of hybridized merge-quick sort algorithm on MPICH. In: Proceedings of International Conference on Distributed Framework for Multimedia Applications (2010)
20. Tsigas, P., Zhang, Y.: A simple, fast parallel implementation of quick sort and its performance evaluation on SUN enterprise 10000. In: Proceedings of Euromicro Workshop on Parallel, Distributed and Network-Based Processing, pp. 372–381 (2003)
21. Tsigas, P., Zhang, Y.: Parallel quick sort seems to outperform sample sort on cache coherent shared memory multiprocessors: An evaluation on sun enterprise 10000. Technical report, Chalmers University of Technology (2002)

22. Weiss, M.A.: Data Structure & Algorithm Analysis in C++. Addison Wesley (1999)
23. Woźniak, M., Marszałek, Z., Gabryel, M., Nowicki R.K.: Modified merge sort algorithm for large scale data sets. Lecture Notes in Artificial Intelligence, Part II, vol. 7895, pp. 612–622. Springer, Berlin (2013)
24. Woźniak, M., Marszałek, Z., Gabryel, M., Nowicki R.K.: On quick sort algorithm performance for large data sets. In: Skulimowski, A.M.J. (ed.) Looking into the Future of Creativity and Decision Support Systems, pp. 647–656. Progress & Business Publishers, (KICSS'2013), Poland (2013)
25. Woźniak, M., Marszałek, Z., Gabryel, M., Nowicki R.K.: Triple heap sort algorithm for large data sets. In: Skulimowski, A.M.J. (ed.) Looking into the Future of Creativity and Decision Support Systems, pp. 657–665. Progress & Business Publishers, (KICSS'2013), Poland (2013)
26. Woźniak, M., Marszałek, Z.: Selected Algorithms for Sorting Large Data Sets. Silesian University of Technology Press, Poland (2013)
27. Woźniak, M., Marszałek, Z.: Extended Algorithms for Sorting Large Data Sets. Silesian University of Technology Press, Poland (2014)

ST Method-Based Algorithm for the Supply Routes for Multilocation Companies Problem

Lidia Dutkiewicz, Edyta Kucharska, Krzysztof Rączka and Katarzyna Grobler-Dębska

Abstract This paper presents an optimization algorithm, based on the substitution tasks method (ST method). It is designed for the supply routes for multilocation companies problem. This problem is NP-hard and belongs to the class of problems for which it is impossible to establish all values and parameters a priori. The substitution tasks method uses a mathematical model of multistage decision process named algebraic-logical meta-model (ALMM). This method allows one to create many algorithms, also automatically. A formal algebraic-logical model of the problem and an algorithm based on ST method are introduced in this paper. Results of computer experiments are presented as well.

Keywords Substitution tasks method (ST method) · Algebraic-logical meta-model (ALMM) · Multistage decision process · Scheduling problem · Multiple traveling salesman problem (*m*TSP)

L. Dutkiewicz · E. Kucharska (✉) · K. Rączka · K. Grobler-Dębska
Department of Automatics and Biomedical Engineering, AGH University of Science and Technology, 30 Mickiewicza Av, 30-059 Krakow, Poland
e-mail: edyta@agh.edu.pl

L. Dutkiewicz
e-mail: lidia@agh.edu.pl

K. Rączka
e-mail: kjr@agh.edu.pl

K. Grobler-Dębska
e-mail: grobler@agh.edu.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_10

1 Introduction

This paper presents an NP-hard logistics problem of supply routes for multilocation companies and the method for solving it. Most of real optimization problems in logistics and manufacturing are NP-hard. Different exact and approximate methods are used to solve them, such as: mathematical methods (e.g., Petri nets, branch and bound, integer programming), heuristic methods (genetic algorithms, tabu search, simulated annealing) [8, 10, 11], and other approaches including e.g., agent methods [6, 12, 14]. Furthermore, there are many problems for which values of some important parameters cannot be determined a priori, because they depend on the current state of the process. For example, time or cost of performing an activity (a job) can depend on types and amount of the resources available at the moment—some resources can even be created during the optimized process. In such cases, it is impossible to determine a priori values of all variables and parameters. In consequence, it is hard or impossible to implement the methods mentioned above. Such problems need an approach that allows one to construct the solution sequentially, during simulation of the process, and provides a mechanism for optimization. An approach that meets these needs is the algebraic-logical meta-model (ALMM).

The algebraic-logical meta-model was created by Dudek-Dyduch [1]. ALMM involves constructing a formal mathematical model of a discrete process. Then a simulation of the process is performed. Thus it is possible to optimize the complicated processes. Following methods have been developed on the basis of ALMM: a method that uses an idea of semi-metric to design local optimization tasks [1], a learning method that uses the information gathered during the searching process [2], a learning method with pruning non-perspective solutions [4], substitution tasks method [3], and a method of switching algebraic-logical models [5]. It should be emphasized that these methods are approximate and allow one to find suboptimal solutions in relatively short time.

In this paper, we present the substitution tasks method (ST method). According to it, a solution is generated by means of a sequence of dynamically created local optimization tasks, so-called substitution tasks. This method provides a basis for developing two-level algorithms, where the first level allows one to create different versions of algorithms by combining parametric elements of the method. It can also be done automatically according to predefined rules, or even randomly. We use the ST method to solve the problem of supply routes for multilocation companies. The problem is to plan routes for salespersons from a supplier company. They have to visit all customer companies, while some of them have many branch offices. The difficulty of the problem is the necessity of visiting the head offices of customer companies prior to visiting branch offices. Thus availability of branch offices depends on the state of the process (it cannot be a priori determined). The problem corresponds to the multiple traveling salesmen problem (mTSP) with additional constraints.

This paper presents an extension of research described in [7] and is organized as follows. Section 2 presents a description of the problem. In Sect. 3, definitions of algebraic-logical meta-model and algebraic-logical model of the problem are given. In Sect. 4 the general idea of the ST method and the algorithm based on it are described. In Sect. 5, an application of the ST method to the considered problem and the experimental results are presented. The last section contains conclusions.

2 Description of the Problem

Let us consider a supplier company that provides goods to several customer companies. Most of them are multilocation companies and have branch offices scattered throughout a wide area. The supplier company employs a number of sales representatives (salespersons) to sell and deliver goods to customers' offices. In the case of multilocation companies, there is a need to visit the head office prior to visiting the branch offices (in order to negotiate, to establish the transaction conditions, etc.).

The problem can be stated as follows. We are given a team of traveling salespersons and a set of locations to visit. Each location represents a customer company office—a head one or a branch one (one-location companies have head offices only). There is also one initial location for the salespersons (i.e., supplier company office). Each location (except the initial one) must be visited exactly once, and any salesperson from the team can do it. Salespersons start from the initial location and return to it at the end. In addition, the locations representing branch offices of a given company can be visited only often the head office of this company has already been visited. However, after the head office has been visited, any salesperson from the team can visit any branch office of that company. There are known distances between locations. The goal is to plan the salespersons' routes such that all locations are visited as quickly as possible.

We use the following notation: $M = \{m_1, m_2, \ldots m_{|M|}\}$ is a set of salespersons and $N = \{n_1, n_2, \ldots, n_{|N|}\}$ is a set of locations to visit. The initial location is denoted as n_0. In addition, a sequence $E = (e_1, e_2, \ldots, e_{|E|})$ is defined, where the individual elements are locations representing the head offices of the multilocation companies or home offices of the companies without branches. Each element of the sequence e_j (for $j = 1, \ldots, |E|$) corresponds to one set F_j of locations representing the given company branch offices. Sets F_j corresponding to companies without branches are empty. Distances between locations are presented in a distance matrix A, where each individual element a_{ij} represents the distance between the location n_i and the location n_j, where $i, j = 0, 1, 2, \ldots, |N|$ and $i \neq j$. All a_{ii} elements are set to infinity. It is assumed that all the salespersons travel with the same unitary speed, so the travel time and the distance traveled during this time have the same value. Also, the time spent in each location is omitted.

Note that the well-known NP-hard problem of multiple traveling salesmen problem (mTSP) can be treated as a special case of our problem—when all customer companies are not multilocation but single ones. Similarly, after visiting all head offices we are left with a mTSP problem to solve. Thus, the considered problem is NP-hard.

3 Algebraic-Logical Model of the Problem

The mentioned problem is modeled as a multistage decision process. Let us recall its formal definition, devised by Dudek- Dyduch [1, 2].

Definition 1 Algebraic-logical meta-model of multistage decision process is a process that is defined by a sextuple $MDP = \left(U, S, s_0, f, S_N, S_G\right)$, where U is a set of decisions, $S = X \times T$ is a set of generalized states (X is a set of proper states and $T \subset \mathbb{R}^+ \cup \{0\}$ is a subset of nonnegative real numbers representing time instants), $f : U \times S \to S$ is a partial function called a transition function (it does not have to be defined for all elements of the set $U \times S$), $s_0 = (x_0, t_0)$ is an initial generalized state, $S_N \subset S$ is a set of not admissible generalized states, $S_G \subset S$ is a set of goal generalized states, i.e., the states in which we want the process to be at the end.

Transition function f is defined by means of two functions, i.e., $f = (f_x, f_t)$, where $f_x : U \times X \times T \to X$ determines the next state and $f_t : U \times X \times T \to T$ determines the next time instant. As a result of the decision u that is taken at some proper state x and a moment t, the state of the process changes to $x' = f_x(u, x, t)$ that is observed at the moment $t' = f_t(u, x, t) = t + \Delta t$. Since not all decisions defined formally make sense in certain situations, the transition function f is defined as a partial one. Thanks to it, all limitations concerning the control decisions in a given state s can be defined in a convenient way by means of so-called sets of possible decisions $U_p(s) = \{u \in U : (u, s) \in Domf\}$.

To define a particular optimization problem in the ALMM methodology, one should build an algebraic-logical model of the problem P and give a specified optimization criterion Q. The optimization task is to find an admissible decision sequence $\widetilde{u}$ that optimizes criterion Q. Thus, the optimization problem is defined by the pair (P, Q).

In this section, an algebraic-logical model of the mentioned problem is described: the state of the system, the set of non-admissible states and the set of goal states, the decision, the set of possible decisions, and the transition function.

Process state. The process state $s = (x, t)$ in particular moment t can be described by the current state of all the salespersons and the set of visited locations. The proper state x is defined as

$$x = (x^0, x^1, \dots, x^{|M|}) \tag{1}$$

where: x^0—the set of locations, that have been visited until moment t, x^k—the state of kth salesperson, for $k = 1, 2, \dots, |M|$.

At each given time, a salesperson can either go to some location, or stay in the last visited location. The state of kth salesperson at time t is as follows: $x^k = (n, r)$, where $n \in N \cup n_0$ is the location the salesperson is going to or location in which the salesperson stays and $r \in \mathbb{R}^+$ is length of the road which the salesperson has to travel to reach the assigned location. If the salesperson is staying in some location, the value of r is equal to 0.

The initial generalized state $s_0 = (x_0, t_0)$ of the process is as follows: $x_0 = (x_0^0, x_0^1, \ldots, x_0^{|M|})$. In the initial state (when $t_0 = 0$), all salespersons are in the initial location n_0, therefore $x_0^k = (n_0, 0)$. There is no location visited at the time t_0, so the set x_0^0 is empty: $x_0^0 = \emptyset$.

We say that the kth salesperson is idle (not working) in state s when he or she is staying in the initial location or the last visited location and we can assign him/her the next location to visit. Therefore, the state of the kth salesperson: $x^k(s) = (n, 0)$. A salesperson is working (is busy) in a given state s when he or she is going to a designated location or has finished the journey and returned to the initial location n_0: $x^k(s) = (((n, r) \wedge r > 0 \wedge n \neq n_0) \vee ((n_0, 0) \wedge t(s) > 0))$.

The set of non-admissible generalized states S_N includes the states in which all salespersons have returned to the initial location but not all locations have been visited: $S_N = \{s = (x, t) : x^0(s) \neq N \wedge \forall_{k=1,\ldots,|M|} (x^k(s) = (n_0, 0)) \wedge t > 0\}$. On the other hand, the aim of the salespersons is to visit all the locations and return to the initial location n_0. Therefore the set of goal states S_G is as follows: $S_G = \{s = (x, t) : s \notin S_N \wedge x^0(s) = N \wedge \forall_{k=1,\ldots,|M|} (x^k(s) = (n_0, 0))\}$.

Decisions. In a given state s, we must take a decision $u \in U = U^1 \times \ldots \times U^{|M|}$. The decision is picked from the set of possible decisions $U_p(x, t)$, i.e., the set of decisions that can be taken in particular state s. In the considered problem, the decision is to determine to which location particular salespersons should go at a given moment. We assume that the decision assigning a salesperson to the next location to visit can be taken only after he or she reaches the previously assigned location. Decisions cannot be changed, for the salespersons who have not yet reached their assigned locations. For a salesperson staying in some location, a decision can be to visit another location, to return to the initial location, or stay in place. Obviously, when assigning new locations, only the ones not yet visited and not currently assigned to any other salesperson can be taken into consideration. Moreover, these have to be locations which represent head offices or which represent branch offices for those companies whose head offices have already been visited. Thus, the decision is a vector $u = (u^1, u^2, \ldots, u^{|M|})$, where particular coordinates u^k represent separate decisions and refer to the respective salespersons (i.e., u^k is the decision for the kth salesperson). We note that the value of each coordinate u^k is the location where a salesperson is supposed to go, or where he or she should stay. Thus $u^k \in N \cup n_0$. The complete definition of the set of the possible decisions is as follows: $U_p(s) = U_p^1(s) \times \cdots \times U_p^{|M|}(s) \backslash H(s)$, where $H(s)$ is the set of decisions assigning the same location to more than one salesperson at the same time and decisions not assigning any new location when all salespersons are idle.

Transition function. Based on the current state $s = (x, t)$ and the decision u taken in this state, the subsequent state $(x', t') = f(u, x, t)$ is generated by means of the transition function f. The transition function is defined for each possible decision $u(s) \in U_p(s)$ and consists of two stages.

First, it is necessary to determine the moment t' when the subsequent state occurs. It is the nearest moment in which at least one salesperson reaches the previously assigned location. The subsequent state x' will occur in the moment $t' = t + \Delta t$, where Δt equals the lowest value of the established completion times:

- if kth salesperson, staying in location n_i, is idle and has assigned location n_j to visit, its completion time is equal to a_{ij},
- if kth salesperson is idle and no new location is assigned, its completion time is set to infinity,
- if kth salesperson is going to previously assigned location n_j, its completion time is equal to r^k (r^k denotes r for kth salesperson).

Once the moment t' is known, it is possible to determine the proper state of the process at this time. The first coordinate $x^{0'}$ (a set of visited locations), is modified by adding each location reached by one of the salespersons at time t': $(x^0(s))' = x^0(s) \cup \{n_i \in N : \exists_{k=1,2,\dots,|M|}(u^k(s) = n_i \wedge r^k = \Delta t)\}$. Afterward, the values of subsequent coordinates, representing the states of the salespersons, are determined. The state $x^k = (n, r)$ goes into $x^{k'} = (n', r')$, for $k = 1, 2, \dots, |M|$. The value of coordinates x^k are as follows:

1. if kth salesperson (staying in location n_i) is idle in a given state s and corresponding part of the decision $u^k(s) = n_j$ is to assign location $n_j \in N \cup \{n_0\}$ to be visited by this salesperson, then we set

$$n' = n_j, \ \ r' = \begin{cases} 0 & \text{for } t^k = \Delta t \\ a_{ij} - \lambda & \text{for } t^k > \Delta t \end{cases} \tag{2}$$

2. if kth salesperson (staying in location n_i) is idle in a given state s and corresponding part of the decision $u^k(s) = n_i$ is to remain in this location, then we set

$$n' = n_i, \ \ r' = 0 \tag{3}$$

3. if kth salesperson is going to location n_i (is busy) in a given state s and corresponding part of the decision $u^k(s) = n_i$ is to continue the previous activity, then we set

$$n' = n_i, \ \ r' = \begin{cases} 0 & \text{for } t^k = \Delta t \\ r - \lambda & \text{for } t^k > \Delta t \end{cases} \tag{4}$$

In the above, t^k denotes the time of completing its activity by the kth salesperson and λ denotes the distance traveled by the salesperson in time Δt.

4 The Idea of the Substitution Tasks Method

Substitution tasks method imitates the way in which a human decision-maker deals with a complex problem. Instead of analyzing the whole problem, the decision-maker usually tries to replace it with a few relatively simple partial problems (intermediate goals) with a shorter time horizon. Furthermore, after implementing each single decision, the decision-maker reconsiders resulting situation and he or she collects and applies additional knowledge in the process. The ST method can be, therefore, classified as an artificial intelligence method.

Substitution tasks method is a constructive method in which whole trajectories are generated. While generating the solution (i.e., the process trajectory), in each state s of the process a decision is made on the basis of a specially constructed optimization task named substitution task $ST(s) = (P_{ST}, Q_{ST})$, where P_{ST} is a certain substitution multistage process and Q_{ST} is a substitution criterion. In order to highlight that the substitution process is constructed for the state s, we use the notation $P_{ST}(s)$.

The substitution task may be different in each state of the process. Substitution tasks are created to facilitate decision making at a given state by substituting global optimization task with a simpler local task. After determining the best decision $\breve{u}(s)$, the next process state s' is generated. Then, an automatic analysis of the new process state is performed and, on the basis of information gained, a new or modified substitution task is defined. Thus, in each iteration of the method, computations are performed at two levels:

L1. The level of automatic analysis of the process and constructing substitution tasks.
L2. The level of determining possibly optimal decision for the substitution task and computing the next state.

Substitution task construction is based upon the concept of so-called intermediate goals d, which are defined to mean reaching a certain set of states S_d by the process as quickly as possible. The distinguished intermediate goals are used to define the set of final states of the substitution process P_{ST}. Thus for the substitution process a new set of goal states $S_{G_{ST}}$ is defined, the initial state s_{0z} is the current state s of the base process P, while the transition function and the sets U, S, S_N are the same as for the basic process P. As a result, the substitution process is a sextuple $P_{ST}(s) = (U, S, s_{0z}, f, S_N, S_{G_{ST}})$.

It needs to be emphasized that each substitution task $ST(s)$ is for choosing only one single decision in state s, and not for determining a sequence of decisions leading the process P_{ST} from this state to one of its final states from $S_{G_{ST}}$.

4.1 Algorithm

It is possible to propose various algorithms based on the substitution task method. In this section, we propose an algorithm with two kinds of intermediate goals:

- $d^{[1]}$—reaching by the process, as soon as possible, a certain set S_d of states in which a location corresponding to a head office of a customer company is visited $S_{d_i^{[1]}} = \{s = (x, t) : n_i \in x^0(s)\}$,

- $d^{[2]}$—reaching by the process, as soon as possible, a certain set S_d of states in which a location corresponding to a branch office of a multilocation company is visited $S_{d_i^{[2]}} = \{s = (x,t) : \forall_{n_i \in F_i} n_i \in x^0(s)\}$.

Goals of the first kind are created on the basis of elements e_j of set E ($j = 1, 2, \ldots |E|$). Corresponding sets S_d include states in which location e_j is visited. Thus, initially there are $|E|$ goals of this kind.

Goals of the second kind are created on the basis of particular elements of sets F_j ($j = 1, 2, \ldots |E|$), but only these ones which are available at the moment. Corresponding sets S_d include states in which locations $n_i \in F_j$ are visited.

In every new state of the process, it is necessary to update the set D of intermediate goals. Each goal (of the first or second type) associated with a location that has just been visited is removed from the set D. At the same time, when goal of the first type is reached and removed from the set D, corresponding goals of the second type are created and included in the set D. These goals are associated with branch offices of the company whose head office has just been visited.

It is important to reach the goals in such an order as to meet the optimization criterion (i.e., in an order that minimizes the time needed to visit all locations). It is accomplished by prioritizing intermediate goals. Priority of each goal is proportional to the number of branch offices that will be made available after visiting the location corresponding to the goal, and inversely proportional to the distance to the location corresponding to the goal. Thus priorities are determined according to the following formula:

$$\alpha(1 + \frac{B}{m_B}) + (1 - \alpha)\frac{m_a}{a_{min}} \tag{5}$$

where: a_{min} is the minimal distance from one of the idle salespersons to the location associated with the intermediate goal, B is the number of branch offices of company associated with the goal (obviously, for goals of second type, B is equal 0), m_a is the average distance between locations, m_B is the average number of branch offices (all companies taken into account), α is a coefficient which modifies the weight of the two components of the formula.

In each iteration of the algorithm (i.e., in each new state), a particular number of goals are chosen for realization. These goals are selected on the basis of priorities and form a set $D_W(s)$ of goals that are chosen for realization. In this set, $|M_W(s)|$ of intermediate goals are placed (where $M_W(s)$ is the set of idle salespersons). If the set D is smaller than the set $M_W(s)$, than there are $|D|$ intermediate goals in the set $D_W(s)$.

In order to make decision $\breve{u}(s)$ in the given state s, it is necessary to define all coordinates of this decision $u(s) = (u^1, u^2, \ldots, u^{|M|})$. It is possible to correlate the realization of particular goals with subsequent decision coordinates u^k, which means dedicating these goals to particular salespersons. Thus, coordinates $u^k(s)$ are determined sequentially by considering the following cases:

- if kth salesperson is going to location n_i (is busy), a decision for him or her can only be to continue its previous activity $u^k(s) = n_i$
- if kth salesperson is idle (staying in location n_i), a decision for him or her is to visit the locations corresponding to intermediate goals chosen for realization (goals from set $D_W(s)$): $u^k(s) = n_j$
- if there are no available locations (no more intermediate goals to assign), a decision for an idle kth salesperson staying in location n_i is to remain in the current location: $u^k(s) = n_i$
- if all locations are already visited, a decision for kth salesperson is to return to the initial location: $u^k(s) = n_0$

The optimization algorithm based on the proposed approach is presented in Fig. 1. Labels L1 and L2 denote levels of computation according to the ST method. Detailed steps of the algorithm are as follows:

Step 1. Setting the initial process state: $s := s_0$.

Step 2. Determining intermediate goals.
Each location that is a head office is associated with an intermediate goal d. The number of intermediate goals in the set D is, therefore, equal to the number of such locations.

Step 3. Calculating priorities of intermediate goals.
Each goal d is assigned a priority $p(d)$.

Step 4. Determining the set $D_W(s)$ of intermediate goals for realization.
In this set, $|M_W(s)|$ of intermediate goals with highest priorities are placed.

Step 5. Determining decision for the given state.
Each location corresponding to an intermediate goal $d \in D_W(s)$ is assigned for visiting, sequentially, to one of the idle salespersons. Busy salespersons continue current activity. If all locations are visited, salespersons return to the initial location n_0.

Step 6. Calculating the next state.
Having established the decision $\breve{u}(s)$, we calculate the next process state s' using the transition function $s' = f(\breve{u}, s)$.

Step 7. Verifying stop conditions.
If the state s' belongs to the set of goal states or the set of non-admissible states, then generating the trajectory is terminated and the result is stored. If this is not the case, the next step should be taken.

Step 8. Updating the set D.
Each goal d, for which the corresponding location has been visited, is removed from the set D. New intermediate goals are created if new branch offices are available.

Step 9. Initiating the next algorithm iteration.
To begin the next step, it is necessary to set $s = s'$, and then to move to step 3 (updating priorities of intermediate goals).

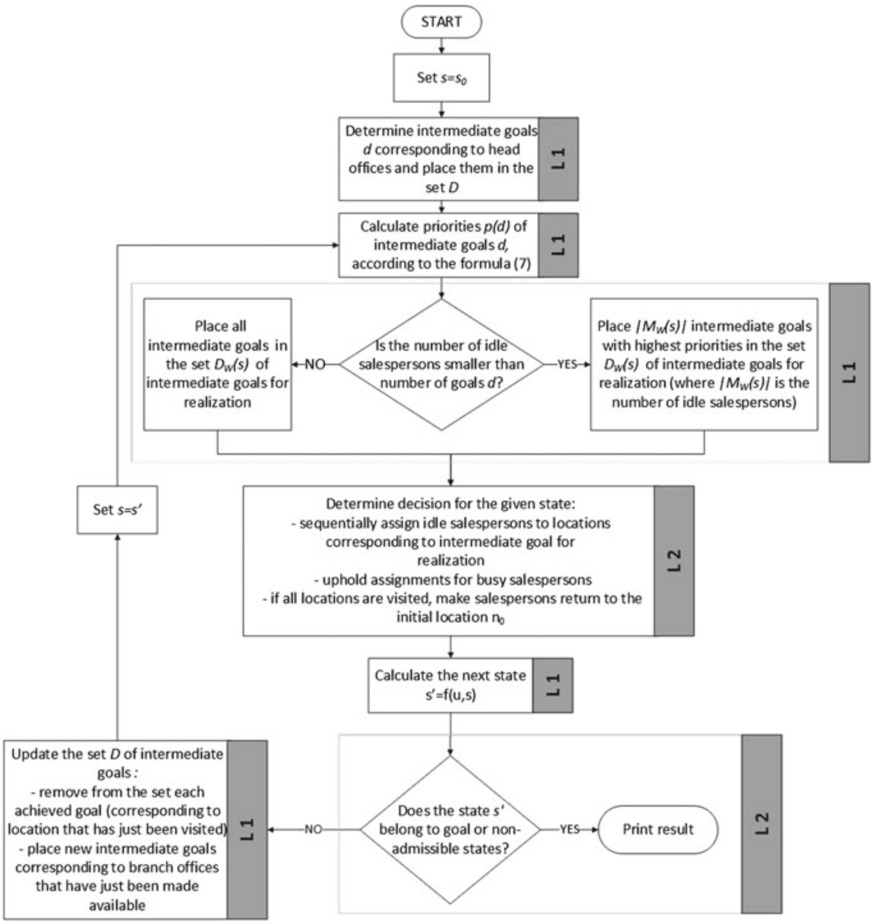


Fig. 1 Diagram of the proposed algorithm

5 Experimental Results

The effectiveness of the algorithm proposed in this paper has been verified through simulation experiments. The algorithm has been implemented in *C#* and tested on an Intel core i5-2410M /CPU 2.3GHz (8GB RAM). We have used data from the TSPLIB. The original instance (bayg29) is the traveling salesman problem with 29 cities. We used it as a base to create five instances of our problem involving multilocation companies. We set the initial location ($n_0 = 1$) and determined different selections of locations representing company head offices and branch offices (and also companies without branch offices). Four locations were established permanently as representing head offices. We modified the number of branch offices, particular locations which represent the branch offices and the number of companies without

Table 1 Results of experiments

Map name	Random decision	$\alpha = 0$	$\alpha = 0.2$	$\alpha = 0.4$	$\alpha = 0.5$	$\alpha = 0.6$	$\alpha = 0.8$	$\alpha = 1$
MAP1	2424	1142	1167	1454	1454	1487	1487	1456
MAP2	2355	1189	1556	1509	1519	1519	1519	1890
MAP3	2489	1434	1335	1538	1538	1538	1538	2590
MAP4	2126	1641	1626	1534	1534	1494	1739	1931
MAP5	2407	1310	1423	1403	1446	1987	1987	1781

branch offices. Instances have companies with the same number of branch offices or with varying numbers of branch offices. We created data sets where branch offices were centered around the head office, as well as data sets where branch offices were randomly dispersed. The computations were made for the case of two salespersons. Particular data items are marked as *MAP1, MAP2, MAP3, MAP4* and *MAP5*. We tested the proposed algorithm for the parameter $\alpha \in \{0, 0.2, 0.4, 0.5, 0.6, 0.8, 1\}$.

Computation time for the presented algorithm was shorter than 1 s. It should be noted that the optimal TSP solution (one salesperson) for instance bayg29 is equal to 1610 [7]. Based on this result, one can estimate the lower bound for the considered problem, taking into account the fact that there are two salespersons, which have to return to the initial location. This estimated value is equal to 841 but it does not include the important constraint that not all locations are available at the beginning. The results are presented in Table 1. A column "Random decision" shows the average objective function value for 10 attempts (decisions were chosen randomly in each process state). Other columns present the results obtained for various values of parameter α. The parameter α determines the weight of the two parts of the priority formula. Small values of α (i.e., the component associated with the distance is more important) give better results for instances with branch offices close to the head office of the firm. Larger values of α (i.e., the component associated with the number of branches is more important) are better for scattered branch offices and instances with many one-location companies. Based on the obtained results, it can be concluded that the proposed algorithm gives a relatively good solution very quickly.

Since the Supply Routes for Multilocation Companies Problem is not a common optimization problem, it is hard to find results to compare with. Nonetheless, we present a rough comparison using a relaxed version of the problem. If, for any instance, we remove the precedence constraints (i.e., each location corresponds to a home office of a company without branches) and employ only one salesperson, we get a typical TSP problem and can compare it with some other heuristic methods. Thus, we run our algorithm for five instances from TSPLIB: Gr24, Bayg29, Gr48, ST70, Ch130. Obtained results are, respectively, 1553, 2005, 6098, 874, 7641. (Average error compared to the optimal solution is about 25 %.) Time of computations was in each case much smaller than 1s. We compared these solutions with experimental results obtained for several versions of genetic algorithms [9, 13] (i.e., the non-revisiting genetic algorithm and algorithms with knowledge-based multiple inversion and neighborhood swapping). Although for the instances with smaller number of cities (24 and 29) our algorithm has provided mostly worse results, it was comparable to genetic algorithms for bigger instances (with 48 and 70 cities). For the biggest analyzed instance, Ch130 with 130 cities, our algorithm has provided a result about 5 % better than the best solution found by genetic algorithms. It should be emphasized that computational time of our algorithm was significantly smaller, especially for larger instances.

6 Conclusions

This paper presents the substitution tasks method, which has been developed on the basis of algebraic-logical meta model. We have used the ST method to solve NP-hard problem of supply routes for multilocation companies. In the problem, the availability of branch offices depends on the state of the process. We have shown that using a formal algebraic-logical model and the ST method a relatively good solution can be found very quickly. Additionally, we present a rough comparison, using a relaxed version of the problem, of the ST method with several other well-known algorithms. We have shown that the ST method can be also used to solve easier problems, even though this method was created to solve problems with variables, parameters or constrains dependent on the state of the process.

It should be noted that the methodology is very universal and can be widely used. It can be applied in decision support systems for logistic and manufacturing, as well as in particular difficult optimization problems. An example can be a problem of mobile robots navigating in a specific environment. The presented approach was implemented as a part of a larger application ALMM Solver. This Solver is intended to solve various discrete scheduling problems using methods based on ALMM. Within ALMM, extensive research is planned, e.g., a comparison of the ST method and learning method with information gathering [4], and examining a possibility of changing the decisions made earlier.

References

1. Dudek-Dyduch, E.: Formalization and analysis of problems of discrete manufacturing processes. Scientific bulletin of AGH UST. Automatics **54** (1990) (in Polish)
2. Dudek-Dyduch, E.: Learning based algorithm in scheduling. J. Intell. Manuf. **11**(2), 135–143 (2000)
3. Dudek-Dyduch, E., Dutkiewicz, L.: Substitution tasks method for discrete optimization. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Żurada, J.M. (eds.) ICAISC 2013, Part II. LNCS, vol. 7895, pp. 419–430. Springer, Heidelberg (2013)

4. Dudek-Dyduch, E., Kucharska, E.: Learning Method for Co-operation. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011 Part II. LNCS, vol. 6923, pp. 290–300. Springer, Heidelberg (2011)
5. Grobler-Dębska, K., Kucharska, E., Dudek-Dyduch, E.: Idea of switching algebraic-logical models in flow-shop scheduling problem with defects. In: Proceedings of the 18th International Conference on Methods and Models in Automation and Robotics: Miedzyzdroje, Poland, 26–29 Aug 2013, pp. 532–537. IEEE (2013)
6. Jędrzejowicz, P., Ratajczak-Ropel, E.: A-team for solving the resource availability cost problem. Computational Collective Intelligence, Technologies and Applications, LNCS, vol. 7654, pp. 443–452 (2012)
7. Kucharska, E., Dutkiewicz, L., Rączka, K., Grobler-Dębska, K.: ALMM approach for optimization of the supply routes for multi-location companies problem, advances in decision sciences and future studies. In: Skulimowsk, A.M. (ed.) Proceedings of the 8-th International Conference on Knowledge, Information and Creativity Support Systems: Krakow, vol. 2, pp. 321–332. Poland, 7–9 Nov 2013, (2013)
8. Oberlin, P., Rathinam, S., Darbha, S.: A transformation for a multiple depot, multiple traveling salesman problem. In: American Control Conference, pp. 2636–2641 (2009). ISBN: 978-1-4244-4523-3
9. Ray, S.S., Bandyopadhyay, S., Pal, S.K.: New operators of genetic algorithms for traveling salesman problem. In: Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04), vol. 2, pp. 497–500 (2004)
10. Shafie, A.A.: Optimal tour construction for multiple mobile robots. J. Eng. Sci. Technol. **6**(2), 215–227 (2011)
11. Sze, S.N., Tiong, W.K.: A comparison between heuristic and meta-heuristic methods for solving the multiple traveling salesman problem. Int. J. Mech., Ind. Sci. Eng. **1**(1), 27–30 (2007), World Academy of Science, Engineering and Technology
12. Wąs, J., Kułakowski, K.: Agent-based approach in evacuation modeling, Agentand Multi-agent Systems: Technologies and Applications—Lecture Notes in Computer Science, vol. 6070, pp. 325–330. Springer (2010)
13. Yuen, S.Z., Chow, C.K.: Applying non-revisiting genetic algorithm to traveling salesman problem. In: IEEE World Congress on Computational Intelligence, Hong Kong, 1–6 June 2008, pp. 2217–2224 (2008)
14. Zachara, M., Pałka, D., Majchrzyk-Zachara, E.: Agent based simulation of the selected energy commodity market. In: Ramik, J., Stavarek, D. (ed.) Proceedings of 30th International Conference Mathematical Methods in Economics, 11–13 Sept 2012, Karvina, Czech Republic, pp. 1004–1009 (2012)

Information Adaptation by an Intelligent Knowledge-Oriented Mechanism

Wiesław Pietruszkiewicz and Dorota Dżega

Abstract Herein, we present a mechanism used to adapt delivered information and oriented onto knowledge. It was deployed within the e-learning software platform and its content. This mechanism was developed and tested in a distributed learning environment, but its capabilities are not limited to this domain. As the e-learning is based on various pieces of software, it is possible to efficiently gather data and extract meaningful information about learner's needs. Together with the delivered knowledge about course, we can use them in the reasoning mechanism, deployed to select proper pieces of content—called as the learning pills—according to the learner's requirements. In the first part of this article, we analyse the organisation of learning process and basic pieces of knowledge delivered in it. Later, we introduce an intelligent auto-adaptive information delivery mechanism. Finally, we discuss its abilities, future research and summarise the paper.

Keywords Intelligent methods · Information and content adaptation · e-Learning · Knowledge engineering

1 Introduction

The development of Internet technology had a positive impact on the growing popularity of distance learning. The last decade was period of huge increase of e-learning (including m-learning) or blended-learning offers, deployed not only by universities and schools, but also by various companies.

W. Pietruszkiewicz
Faculty of Computer Science and Information Technology,
West Pomeranian University of Technology in Szczecin,
ul. Żołnierska 49, 71-210 Szczecin, Poland
e-mail: wpietruszkiewicz@wi.zut.edu.pl

D. Dżega (✉)
Faculty of Economics and Computers Science, West Pomeranian
Business School in Szczecin, ul. Żołnierska 53, 71-210 Szczecin, Poland
e-mail: ddzega@zpsb.szczecin.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_11

Together with the technological advancement, and accessibility to the vast amount of information stored in the distributed (Web) environment, it became necessary to provide requested information, additionally selected according to its relevancy. This also applies to the process of e-learning, where information provided by its mechanisms must be tailored to the users' requirements and expectations, concerning the expected outcome of learning process, i.e. the new gained skills.

The creation of modern courses involves various specialised resources. These resources include the project team, e-contents manufacturing technology and IT systems supporting the process of distance learning. Information provided for students should meet their needs, preferences and environmental determinants. The adjustment of e-courses to the varying conditions is a process of their adaptation.

An analysis done for the research in this area reveals that most papers are focused on the theoretical aspects of adaptation. The implementation of these solutions in practice is still a big challenge. This area is an important application of information and knowledge management in the distributed Web environment. Adaptive Web mechanisms, orientated towards information and knowledge, fit into general tendency of Web 3.0.

1.1 The Definition of Adaptation in e-Learning

The issues of adaptive approach to e-learning have been a subject of studies done by many researchers and research centres. The forerunner of the adaptive approach in e-learning was research presented in [2]. This paper pointed out the benefits of adaptive and intelligent technologies for large-scale Web-based education. The definition of e-learning adaptation given in [12] drew attention to the needs of design, where learning environment would assemble the adaptation of e-learning content, pedagogical, e-learning personalisation and interaction between the participants.

Other researchers in [8] have identified four categories of learning environment adaptations, i.e. adaptive interaction, adaptive course delivery, content discovery and assembly and adaptive collaboration support. In the proposed approach they have emphasised that the adaptation modelling in context of adaptive learning environments is a more complex issue. According to their opinion it is caused by a potential involvement of adaptation "logic" and "actions", resulting from the application of that logic in relation to the various static and dynamic models maintained by the adaptive learning environments. The general idea of adaptive e-learning was also a subject of research done by [14], where adaptation included an analysis of learner's performance and selection of proper learning strategy.

In this paper we introduce the concept of learning pills and PCR matrix, allowing the adaptive mechanism not to be oriented towards the content, but towards the competences build by providing the proper content.

1.2 Adaptation of e-Learning Content

The adaptation e-learning content is oriented on people skilled for work in an online environment and being properly motivated. The e-learning adaptation usually leads to the design of appropriate individualised learning paths. Possible solutions include models build over the ACO (Ant Colony Optimization) and other path-finding methods [11]. Another approach used in creation of adaptive learning paths is the instructional design PENTHA model (Personalization, Environment, Network, Tutoring, Hypermedia, Activity), based on five conceptual dimensions: Knowledge, cognitive, didactical, semiotic and social [3].

1.3 Personalisation and Interaction in e-Learning

The personalization in e-learning is similar to the personalization of websites. The most of e-learning platforms, such as Moodle, Blackboard and Sakai, offer personalization of educational services built over them. Adaptation in these sites is possible at three levels: for service administrator, e-teachers (later called as tutor) and learners. The range of personalisation and relevant LMS features have been presented in Table 1. In addition, the functionality of e-learning platforms allows one to display the content linked with the author's profile. The examples of such functionality include messages posted on forums, posts in chat rooms, blogs or dictionaries.

A very important element, which has to be taken into account when preparing adaptive e-learning courses, is an interaction between learners. The most popular solutions, provided by the standard functionality of e-learning platforms, include discussion forums and chat rooms, shared creation of documents (e.g. articles in the wiki) and cooperative working on whiteboards.

The mechanism proposed herein fits into all levels of personalisation:

- it offers auto-personalisation at the learners level,
- requires tutors to define information relating to course content,
- allows administrators to define overall parameters (according to the organisation's policy and expectations).

1.4 Individual Learning Paths

In addition to providing personalised learning paths (discussed in [9]), we must pay attention to the scope and quality of feedback [13]. Feedback is an essential component of pedagogical aspects during e-content design and its implementation. The involvement of feedback in e-learning can be arranged in form of

Table 1 The range of personalisation and related LMS features for different levels of personalisation

Group	Personalisation range	Personalisation features
Learners	The smallest privileges, i.e. adjustments of displayed content	Information block's visibility, visual themes, configure notification methods for incoming messages, management private budges, files, calendars, comments, connect with external resources (e.g. Dropbox, Google Drive, Picasa, YouTube), tracking of course progress
Tutors	The medium rank privileges, i.e. adjustments done to meet the course's needs	Learners' features, course content, grades configuration, configure course completion, users enrolment and management of their permissions
Administrators	The highest privileges, i.e. can perform service adjustments matching the university's or company's needs	Tutors's features, platform structure and functionality

- content delivery done according to the gathered data about learner;
- proper design of messages displayed for learners, after answering the test questions included in the e-content. It is recommended to prepare other messages after the correct answers (to strengthen the learning effect), and different replies to the incorrect ones (guidance on correct reasoning);
- parameter information about learning containing: the data representing activity time, data about intensity of learner's activity and navigational data.

Slightly a different approach to the adaptation in e-learning introduced [1], where adaptation was defined to be a combination of two technologies: intelligent tutoring systems and adaptive hypermedia systems. Intelligent tutoring system may include items such as measures of site quality, users, observation, adaptive mechanism or course evaluations of achievement [5].

1.5 One-Size-Fits-All Versus One-to-One—Requirements, Costs and Effects

There is another aspect of adaptation in e-learning—it replaces one-size-fits-all with one-to-one teaching approach. This transformation generates a number of requirements, and a need to incur significantly more resources to prepare e-contents, and

Table 2 Requirements and costs for one-to-one approach

Group	Effects	Requirements and costs
Learners	More comfortable learning	Additional time in case of incorrect self-learning
Tutors	Individual control, better employee assessment, development of teaching skills	Additional content must be prepared, increased time of learners' observation
Organisation	Increased service quality, improvement of market competitiveness	Increased cost of highly skilled personnel (tutors)

later on to carry out e-courses. In the design of adaptive e-contents, one must take into account the needs of person being simultaneously: a learner, user of a Web-based system user and a business client for university, school or training company [7]. The main challenge, apart of financial aspects for such project, is to prepare e-content in a way ensuring the learning outcomes for all students (Table 2).

2 Learner's Experience-Based Adaptation in e-Learning

The adaptive design of e-learning materials should stimulate occurrence of deep processing for learners and the incremental construction of new content. The process of deep processing is a repeated manipulation of content one will learn.

Additionally, it can be achieved by a simple paraphrasing of processed content. This process should take place at the time of learning. In the result, an incremental building of new knowledge will be a result of experience acquired by participants during the e-course. It may also take a form of reflection, and after examination newly acquired knowledge applied in practice.

2.1 Kolb's Learning Cycle Applied in the e-Content Design

The Kolb's cycle is an experimental learning model, where deep processing of content and different styles of learning are preserved. These styles include feeling, watching, thinking and action (see [4, 6]), and this is a basic mechanism intuitively used by learners, especially by adults.

A properly designed e-content should consist of a number of teaching units—modules. The smallest unit of module is a learning pill, which should reflect a separate portion of information related to the module's topic. Fig. 1 presents components of the e-content inspired by the Kolb's cycle.

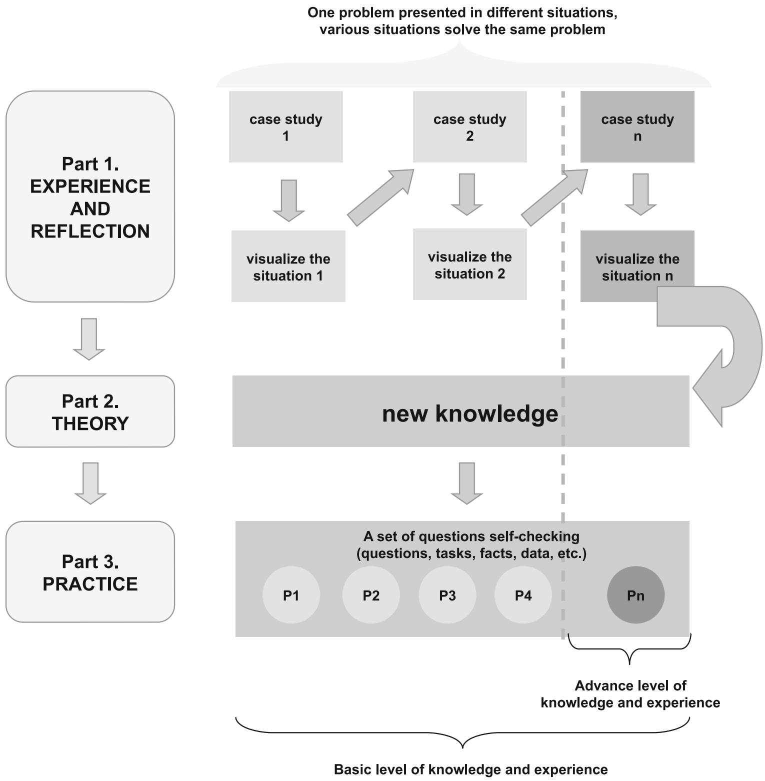


Fig. 1 Visualisation of e-content based on Kolb's cycle

In proposal presented herein, we have taken into account the concept of adaptation, involving the levels of learners' advancement. Determination of levels can be done by pretests, done prior to the participation in e-course. Each unit has four parts (in practice first two of them have been merged, to ensure higher relation between examined case study and its reflection):

1. **The experience** is understood as "immersion in the experience". This part of e-content makes a reference to experiences that learners may have. The reference to the experience contributes to a different problem solving, which can be done for the same problem. In the result, "immersion in the experience" is achieved by providing a properly constructed content in form of a case study.
2. **The reflection** is understood as a moment to think about what happened in the case study, and what was illustrated on the screens with experience. Due to the nature of e-learning materials this element shall confront the learner's and expert's (author of e-content) reasoning.
3. **The theory** is understood as a new knowledge to be acquired. This part of learning pill concludes knowledge delivered in the previous parts. It has a form of mini

lectures containing text, figures, internet links, etc. The presented content must strongly correspond with the content of experience and reflection parts.

4. **The application in practice** is an opportunity to test newly acquired knowledge in practice. This part of pill checks, via questions, if learner is able to use new knowledge. An important element of this part is a feedback received after each question. It takes form of extensive comments after each learner's answer.

3 The Adaptive Content Recommendation

The procedure of content adaptation must be organised in a way minimising costly manual operations and it must be a general framework, not tailored to a particular course or students. This section presents much mechanism that was developed to achieve one-to-one relation between course participant and learning platform.

3.1 From the Facts to Recommendation

To create adaptive content recommending mechanism it is necessary to develop a structure capable to examine information about learner, match it with the requirements relating to course and deploy knowledge build in it produce outcomes, being a set to recommended pills to learner by particular user (learner). According to our analysis it is possible to distinguish various elements relating to the data, information and knowledge within adaptation of e-learning content (see Table 3).

Table 3 Data, information and knowledge deployed in the adaptive e-learning

Level	Elements	Used as
Data	Answers to the preliminary tests; answers to the expectations questionnaire; events logged for user's behaviour (a way user is following his/her learning path)	Basic observation required to be processed at start-up and runtime, and to be transformed into learner's profile
Information	Evaluated user's competences and expectations	Processed observation influencing the mechanism outcome for a particular learner
Knowledge	Relations between learning pills and competences; shape and weights of objective function	The definition of course expected outcome, matching organisation's learning standards and course characteristics

They are essential to the parametrisation and development of e-course. Some elements are gathered during course usage, while other must be provided before any user starts learning, as a initial course configuration. The analysis of data & information & knowledge sources can be performed including their sources, and for the proposed mechanism we have identified three of them:

1. **course/content creator (usually a tutor)**—defines the relations between learning pills and competences,
2. **learner**—answers to the preliminary test and defines his/her expectations i.e. new competences to be learned during the course,
3. **administrators**—define competences and parameters for objective function used by the recommendation mechanism.

3.2 The Proposed Information Adaptation Mechanism

Let us start analysis of adaptive mechanism from the overall structure of content, containing learning pills (discussed earlier). Each pill is an independent piece of knowledge to be learned and is an element of course. The course is a set of pills (see Fig. 2).

The adaptation of course is done via recommendation mechanism. Its general algorithm follows these steps:

1. **Creator** defines new content and performs parametrisation.
2. **Administrator** starts a new course and deploys content, performing parametrisation of objective function.
3. **Learner**, before taking a course, answers pretest and defines his/her expectations.
4. **Recommendation mechanism** delivers an individual learning path.

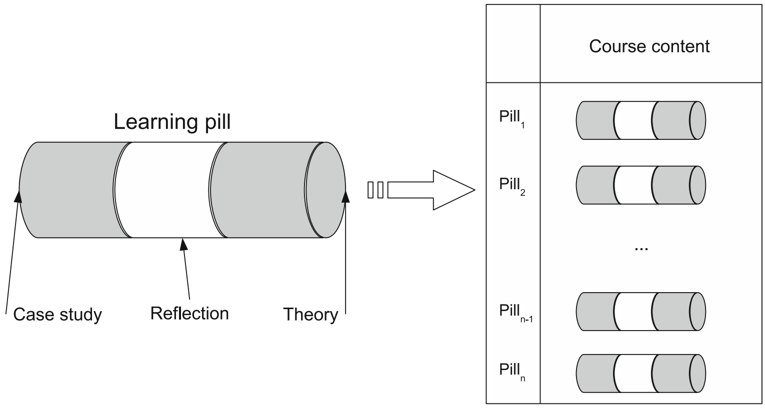


Fig. 2 The concept of learning pills and their relation to course

5. **Learner** follows the path.
6. `If` significant deviations from learning path occur, `then` **recommendation mechanism** adjusts learning path.

As it can be noticed the recommendation mechanism has two modes of functioning (see Fig. 3a and b). According to the time of recommendation, we can define two stages:

1. **Start-up recommendation**: When one starts to learn it is necessary to deliver a recommendation, which is based on two pieces of information—target competences and PCR matrix (explained later)—while the outcome is a set of learning pills being an individual course structure. It is also called as the selected learning path.
2. **Runtime recommendation**: If one does not learn according to the recommended path (with a tolerance for small deviations), it is necessary to recommend learned pills, to ensure they have not been forgotten.

The objective function in general is a function class that must be adjusted for each course. Together with information about user and course, it is used to individually select learning pills. It can be denoted as

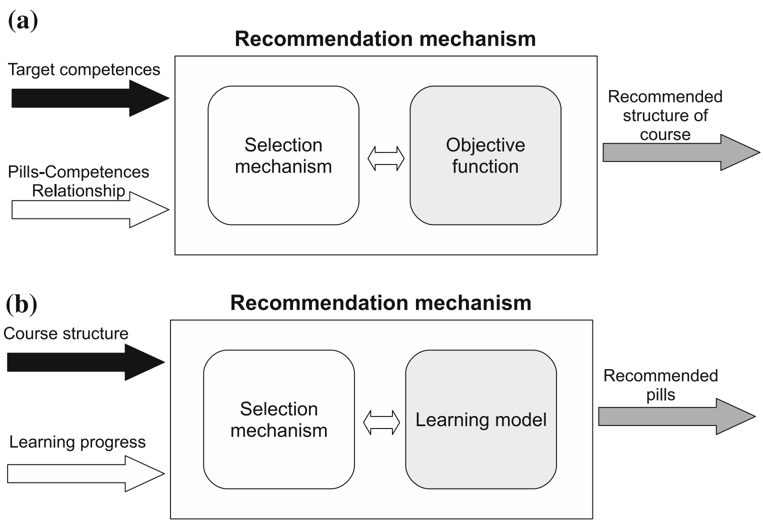


Fig. 3 Recommendation mechanism at start-up time (**a**) and runtime (**b**)

$$F(length, hardness, outcome) = \overbrace{weight_{length} * length()}^{\text{length component}} + \overbrace{weight_{hardness} * hardness()}^{\text{hardness component}} + \overbrace{weight_{outcome} * outcome()}^{\text{outcome component}}$$

where each component denotes

- *length*—an average length of course expected by person creating its content (it influences number of learning pills that will be delivered),
- *hardness*—assumed hardness for course, depends on types of study (e.g. very low for employees attending company course, very high for graduated students),
- *outcome*—competences that must be acquired by user, other competences are considered to be optional and optimally selected according to user's profile.

To successfully recommend pills matching learner's requirements (what learner should learn) and expectations (what learner would like to learn), we must introduce the PCR matrix. The PCR (pill–competence relationship) matrix is unique for each course. It is a two-dimensional structure (see Table 4) and measures the strength of relations between pills and corresponding competences.

The PCC_{ij} (pill–competence coverage) is a measure of $competence_i$ coverage in $pill_j$. Its boundary values ($PCC_{ij} \in \langle 0, 1 \rangle$) should be interpreted as follows: 0—there is no competence coverage in pill, 1—competence is fully covered by pill. This matrix is created by a person responsible for learning pills belonging for a particular course.

As it can be noticed in the provided example (see Table 5), PCR is a sparse matrix. Usually, each competence is covered by different learning pills (e.g. C_5 is covered by P_1, P_7, P_8, P_{10} and P_{11}), and some each pill covers different competences, e.g. Pill P_1 covers competences C_1 and C_5. However, in the second statement it is not always required, as some pills may cover a single competence. Assuming equal hardness of pills $\sum^i PCC_{i.} = 1$.

Table 4 The pill–competence relationship matrix

	$Pill_1$	$Pill_2$	$Pill_3$	...	$Pill_m$
$Competence_1$	PCC_{11}	PCC_{12}	PCC_{13}	...	PCC_{1m}
$Competence_2$	PCC_{21}	PCC_{22}	PCC_{23}	...	PCC_{2m}
$Competence_3$	PCC_{31}	PCC_{32}	PCC_{33}	...	PCC_{3m}
...	...	...	...	...	...
$Competence_n$	PCC_{n1}	PCC_{n2}	PCC_{n3}	...	PCC_{nm}

Table 5 The PCR matrix for a sample course

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}
C_1	0.5	0.5	0	0	0.5	0	0	0.33	0	0	0
C_2	0	0.5	0.5	0	0.5	0	0	0	0	0	0
C_3	0	0	0.5	1	0	1	0.5	0	0	0	0
C_4	0	0	0	0	0	0	0	0.33	1	0	0
C_5	0.5	0	0	0	0	0	0.5	0.33	0	1	1

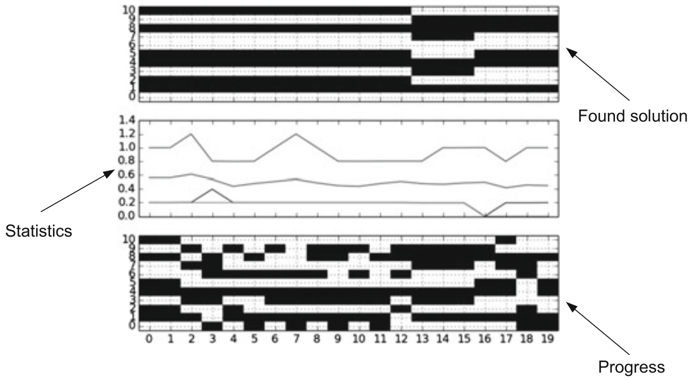


Fig. 4 Progress of information adaptation by pills optimisation for a sample course

Having defined objective function, PCR matrix and expected outcome, it is possible to perform content adaptation via optimisation. Technically this is a binary optimisation problem solved by the genetic algorithms. An example of this optimisation procedure is presented in Fig. 4. This figure contains three parts: the first one is a best solution found at each step, the second one is a plot with statistical data for populations and the last one contains the best individuals in each generation (population).

This method was developed to be used in e-learning process. However, with a proper course organisation it could be used in traditional or blended learning. Obviously in such situation it will not offer all benefits available in a distributed learning process fully supported by the software platform.

4 Conclusions

The adaptation of e-learning content is a process requiring different resources including financial, human and technological resources. The approach presented herein proposes an concept of intelligent auto-adaptive Web e-content presentation mechanism, using learning pills. It is an interesting solution, which combines the possibilities of using Web technology and knowledge accumulated during the learning process. Due to its construction it is easy to deploy and does not require costly add-ons, neither to the e-learning platform nor e-content. After an evaluation of method proposed herein (planned in our previous paper [10]), we observed its performance and practical issues relating to the preparation of courses according to the method's requirements. As it was mentioned earlier, author's additional tasks have been limited only to the preparation of PCR matrix. Therefore, it does not introduce any substantial burden.

Future work relating to this proposal will involve the general rules for the design of information systems, such as learnability, efficiency, memorability, error-prone and user satisfaction. In addition, this process should use additional information sources, gathered as the experience and observations of distance learning participants' behaviour. Another promising idea seems to be an introduction of repetition spacing algorithms, e.g. SM-8 [15] or SM-11 [16], which will reinforce the process of learning.

In general, adaptation of e-content should be perceived as one of the Web 3.0 activities, as it reshapes content to match one's expectations and requirements. Therefore, the traditional one-size-fits-all courses will be replaced by one-to-one. However, to introduce such functionality content adaptation methods must be feasible in the business terms.

Acknowledgments This research was funded by the research grant N115 413240 from the National Science Centre in Poland.

References

1. Beldaglia, B., Adiguzela, T.: Illustrating an ideal adaptive e-learning: a conceptual framework. Procedia Soc. Behav. Sci. **2**, 5755–5761 (2010)
2. Brusilovsky, P.: Adaptive and intelligent technologies for web-based education. Spec. Issue Intell. Syst. Teleteaching **4**, 19–25 (1999)
3. dall'Acqua, L.: A model for an adaptive e-learning environment. In: Proceedings of the World Congress on Engineering and Computer Science, WCECS 2009, vol. 1, pp. 604–609 (2009)
4. Dunlap, J., Dobrovolny, J., Young, D.: Preparing e-learning designers using kolb's model of experiential learning. Innov.: J. Online Educ. (2008)
5. Dżega, D., Pietruszkiewicz, W.: Intelligent Decision-Making Support within the E-Learning Process, pp. 497–521. Springer (2012)
6. Kolb, A., Kolb, D.A.: Experiential Learning Theory Bibliography 1971–2001 (2001)
7. Moebs, S.: A learner, is a learner, is a user, is a customer: Qos-based experience-aware adaptation. In: Proceedings of the 16th ACM international conference on Multimedia, pp. 1035–1038 (2008)
8. Paramythis, A., Loidl-Reisinger, S.: Adaptive learning environments and e-learning standards. In: 2nd European Conference on e-Learning (ECEL 2003), (November 2003)
9. Pietruszkiewicz, W., Dżega, D.: The Artificial Intelligence in the Support of e-Learning Management and Quality Maintenance. IGI Global, pp. 92–127 (2012)
10. Pietruszkiewicz, W., Dżega, D.: Auto-adaptive web e-content presentation. In: Skulimowski, A.M.J. (ed.) Proceedings of the 8th International Conference on Knowledge, Information, and Creativity Support Systems (KICSS 2013), pp. 531–540 (November 2013)

11. Pushpa, M.: ACO in e-learning: towards an adaptive learning path. Int. J. Comput. Sci. Eng. **4**(3), 458–462 (2012)
12. Stoyanov, S., Kirschner, P.: Expert concept mapping method for defining the characteristics of adaptive e-learning: Alfanet project case. Educ. Tech., Res. Dev. **52**(2), 41–56 (2004)
13. Vasilyeva, E., Pechenizkiy, M., Bra, P.: Adaptation of feedback in e-learning system at individual and group level. In: Proceedings of the 5th international conference on Adaptive Hypermedia and Adaptive Web-Based Systems, pp. 235–244 (2008)
14. Wiggins, G., Trewin, S.: A system for concerned teaching of musical aural skills. In: Gauthier, G., Frasson, C., VanLehn, K. (eds.) Intelligent Tutoring Systems. Lecture Notes in Computer Science, vol. 1839, pp. 494–503. Springer, Berlin (2000)
15. Woźniak, P.: Algorithm used in supermemo 8 for windows. http://www.supermemo.com/english/algsm8.htm. Accessed: (September 2013)
16. Woźniak, P.: Repetition spacing algorithm used in supermemo 2002 through supermemo 2006. http://www.supermemo.com/english/algsm11.htm. Accessed: (September 2013)

Eval-net: Elements of Evaluation Nets as Extension to Petri Nets

Michał Niedúwiecki, Krzysztof Rzecki and Krzysztof Cetnarowicz

Abstract Evaluation nets and Petri nets are an easy, readable and functional methods for visualizing states and communication between computer systems using diagrams. They can be considered as similar methods, because they base on bilateral graph and there is token flow. However, they differ in elements, token flow rules, structure, and runtime process. Petri nets are much more popular than evaluation nets and there are many tools to model, visualize, verification and simulation such nets, unlike forgotten evaluation nets. Evaluation nets and related source program code represent the algorithm, evaluation nets control the code execution. In this article, an exstension to Petri nets called Eval-nets is presented. Eval-nets introduce the most useful elements of evaluation nets into Petri nets. This extension is capable of being used in existing tools for Petri nets. As a result a functional tool for creating, analysing, running, debugging, and simulating communication protocols may be build based on Petri nets.

Keywords Evaluation nets · Eval-net · Petri net · Simulation

1 Introduction

Evaluation nets [17] were invented at the beginning of 1970 and became an alternative to Petri net [22]. This method has some restrictions and is not so easy in automated analysis, so is not so popular.

M. Niedúwiecki (✉) · K. Rzecki
Cracow University of Technology, ul. Warszawska 24, 31-155 Krakow, Poland
e-mail: nkg@pk.edu.pl

K. Rzecki
e-mail: krz@pk.edu.pl

M. Niedúwiecki · K. Cetnarowicz
AGH University of Science and Technology, al. Mickiewicza 30,
30-059 Krakow, Poland
e-mail: cetnar@agh.edu.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_12

Evaluation nets consists of two parts: a frontend and a backend.

The frontend is similar to a Petri net. The backend is the source code that is executed at the appropriate time and has an impact on the future flow of tokens. The backend determines which of the enabled transitions will be firing.

It allows for easily linking a visualization state and sending messages to the actual source code. However, this relationship makes it difficult or even impossible to use mathematical tools to analyse the entire system because these tools examine only the graph and not the source code. For example, a reachability graph based on Petri net can be built. A similar graph can be built on the basis of the evaluation nets frontend. However, when you start such evaluation nets it may be the case that the backend will never run some routes and some states of the reachability graph will became unattainable.

The properties of evaluation nets had been very useful for modeling and verifying negotiation protocols (e.g. Complex Negotiations protocol [15]) and our other communication protocols between software agents and mobile robots. Network analysis tools allow searching for potential pitfalls, and the subsequent analysis of the algorithm for their verification. However, this initial enthusiasm has cooled due to the lack of tools for working with evaluation nets. This fact was the motivation to develop own tools for this aspect.

2 Related Works

Petri net was invented in the 1960s by the German mathematician Carl Adam Petri, and described in his doctoral dissertation [18]. Petri net is used to model systems mathematically. The system (including software systems) is modelled using conditions and events represented by state transition diagrams. Petri nets have become very popular because they provide a language that allows the automatic generation of a specific simulation. However Petri nets can become too large to generate all states of the system. This caused new extensions to be introduced.

One of the groups of extensions involved is conflict structures, situations in which several transitions are enabled, but only one of them can be fired. This group includes extensions like Timed Petri nets [19], Stochastic Petri nets [2], Controlled Petri net [11], Labelled Petri nets [3], etc.

Another group deals with the attributes of extension tokens. The original Petri net tokens have no attributes and there is no limit to their number. This group includes extensions like Coloured Petri nets [20], Objected Petri nets [13], Hierarchical Petri nets [9], etc. Some extensions are applicable for all tokens, like inhibitor arc [21], reset arc [6], capacity of places [4], etc.

There are a lot of mathematical tools available to analyse Petri nets. A large set of tools was presented in [22]. These tools have been implemented in some WYSIWYG Petri net editors like YASPER [23], WoPeD [7], PIPE2 [5], CNP Tools [20], etc. Thanks to them, creation and analysis of Petri nets is easy. A large set of Petri net editors was found and described in [10]. Some tools like PNtalk [12] allow joining Petri net flow with source code execution.

Original evaluation nets (*E*-Nets) are described in [16, 17]. Since their development, evaluation nets have evolved. Some extensions were presented in [1]. These extensions were used in [24].

Evaluation nets are based on Petri nets, but many of their features, such as marking, transition rules, token manipulation, etc. vary quite significantly. Above all evaluation nets add a backend layer that connects the graph (frontend) with the source code. This is the greatest advantage.

Coloured Petri nets have this advantage but it introduces significant changes in the frontend and decreases the readability of the diagram. Therefore Eval-net is proposed.

Eval-net extensions try to adapt the backend, but without introducing additional and unnecessary changes to the frontend. Moreover Eval-nets support communication between separated nets because Eval-net is dedicated to modeling agents behaviour.

As the result we got the tool for applications modeling, implementing and graphical monitoring its activities. In addition, the Eval-net diagram can be easily exported to a Petri net. It gives the ability to use numerous Petri net analyzing tools.

3 Concept of Eval-net

The aim of the Eval-net is to enable implementation of the algorithm and state machine in the form of Petri net diagram combined with source code stored in a programming language. This collaboration must be bilateral, it means:

1. The order of procedures in the source code must be determined by the Petri net.
2. The result calculated in the source code, or a signal received out of the program must affect the future flow of tokens in the Petri net.

While the Petri net model is adequate for the entire system, the Eval-net is more suitable for the implementation of the single elements of the system. Instance of the class (object) or software agent are the examples of such element. Element keeps its status and communicates with other elements. Execution of procedures (events) in the element takes place in a multi-threaded system.

Each algorithm is composed of branches, condition instructions. In the Petri net branches are modeled in the form of *conflicted structures*, situations where many of the transitions may be active simultaneously, and the firing one of them disables remaining. Firing transition is associated with the execution of the assigned procedure (event) in the element, but it is not possible to receive feedback, that would determine the firing of concrete transition in conflicted structure. The firing transition control may be moved to source code element, but Petri net will act as monitor only.

3.1 Eval-net Definition

Eval-net formalism and naming conventions are based on [22, 25]. For those extensions adopted from evaluation nets that have no equivalents in Petri net, names from [16, 17] are used.

Definition 1 *Eval-net* is defined as:

$$EN = (F, B) \tag{1}$$

- F—frontend (graph),
- B—backend (data and code).

Petri net is frontend only. Evaluation nets have a frontend and a backend, but the frontend is not compatible with the frontend of the Petri net. These differences are presented in [14]. Eval-net inherits the frontend from Petri net and the backend from evaluation nets. The frontend provides the primitive for describing synchronization of concurrent threads and communication between other processes (i.e., agents). It is used for modelling, visualizing, and analysing the element of the system. The backend is the body of the system. The backend provides methods for the manipulation of program data. The backend makes decisions but the flow is managed by frontend.

Definition 2 *Frontend* is defined as:

$$F = (P, T, I, O, R, S, U, r, M_0) \tag{2}$$

- $P = \{p_1, \dots, p_n\}$ is a finite and nonempty set of places,
- $T = \{t_1, \dots, t_m\}$ is a finite and nonempty set of transitions, $P \cup T \neq \emptyset$,
- $I : P \times T \mapsto N_0$ is a mapping $P \times T \mapsto 0, 1$ corresponding to the set of directed arcs from places to transitions (input arc),
- $O : T \times P \mapsto N_0$ is a mapping $T \times P \mapsto 0, 1$ corresponding to the set of directed arcs from transitions to places (output arc),
- $R = \{r_1, \dots, r_m\}$ is a finite resolution locations, $r_i = (t_j, q_j)$ where t_j is an assigned transition and q_j is a *resolution varible* from the backend,
- $S = \{s_1, \dots, s_{m:n}\}$ is a finite set of message senders, $s_i = (d, o_i)$ where d is message name and o_i is an assigned output arc,
- $U = \{u_1, \dots, u_m\}$ is a finite set of message receivers, $u_i = (d, r_i)$ where d is message name and o_i is an assigned resolution location,
- $f \in P$ is a optional special place named final place, which highlights the final state of the Eval-net,
- $M_0 : P \mapsto N_0$ is the initial marking.

Figure 1 shows graphical notation of the frontend elements. This notation is based on the Petri net and introduces several enhancements to it in the form of *special places*. These extensions are designed to provide ability for receiving control messages from the source code.

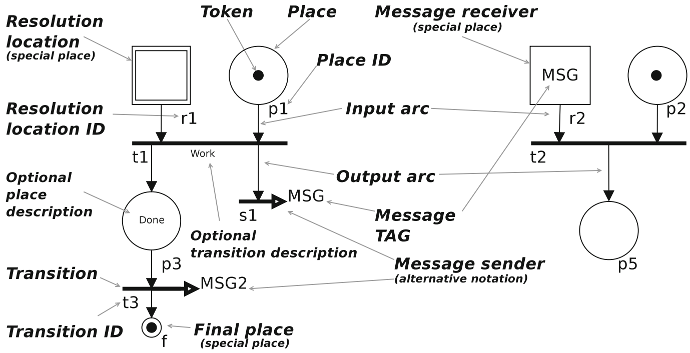


Fig. 1 Graphical notation of Eval-net

The *resolution location* represents signal reciving directly from thread running in the source code (i.e. `enable('r1')` in Fig. 3). It is presented a as a square with a double border on the diagram.

When in the *message sender* the token appears, the proper procedure responsible for assigning the contents of the message and send it to another element (i.e. `t1MSG` function in Fig. 3) is called. The *Message sender* has a tag for the communication module to identify the recipient. In the diagram the *message sender* is presented in two ways: as a place in the form of arrow or as an arrow united with transition. Both records are equivalent, but the first allows you to connect several *message sender* to one transition.

When a message is sent and reaches the recipient, it is processed by a procedure currently contained in the source code (i.e., `rcvMSG` procedure in Fig. 3). When this procedure is completed there in the *message* receiver appears token. It is presented as a square with a single frame on the diagram.

Using f place is optional. When f place contains a token, then the Eval-net instance can be shut-down.

Conditional construction `switch...case` is very popular in algorithms, but Petri net representation is not enough clear. This is why in Eval-net there are introduced macros. Macro allows to save conditional construction in an easy way. On the picture 2, there is an example of such macro with original diagram fragment it represents.

Definition 3 *Backend* is defined as class interface with methods and fields. The code execution can be performed in several threads.

The fields are unsigned integer variables and specifies the number of tokens in the associated *special places*. The procedure in the source code has the ability to increase the number of tokens. The combination of *special places* and transitions in the conflicted structure gives the opportunity to influence the source code execution.

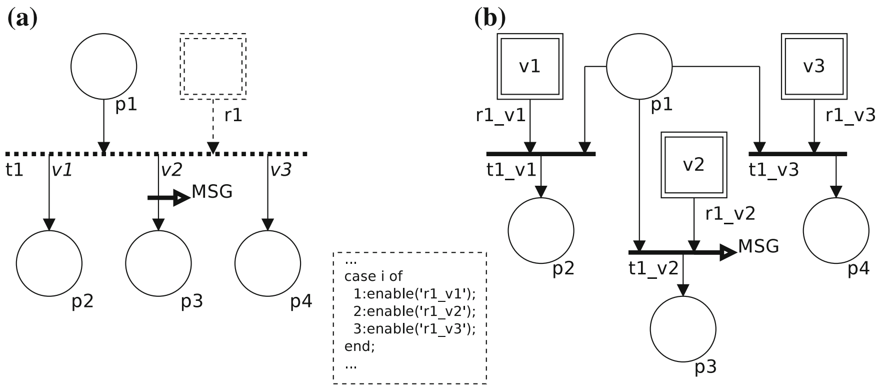


Fig. 2 **a** Example of a macro for the structure `switch...case` and **b** explication of this macro

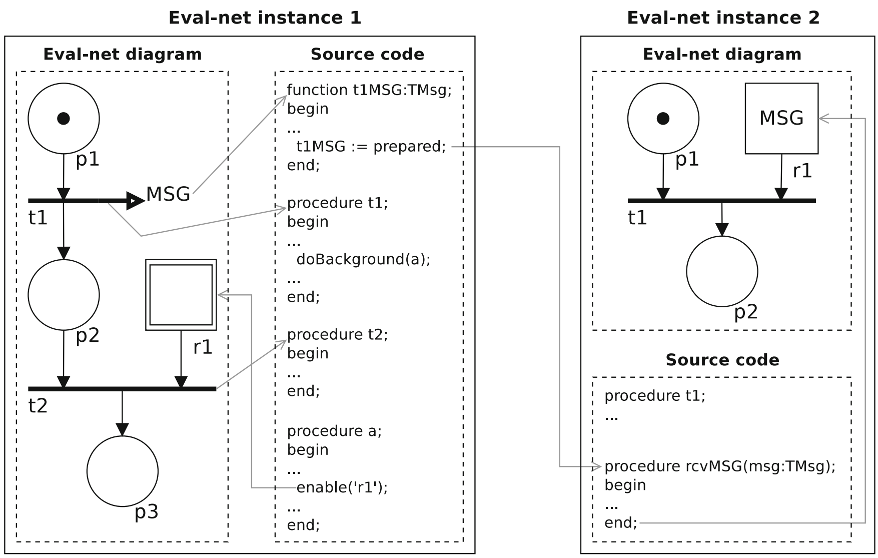


Fig. 3 Flow control between diagram and source code

Methods of the class are more different and there are:

- *transition procedure*—is executed by firing transition, just before the number of tokens in input places is decreased, just after the number of tokens in output places is increased,
- *receive procedure*—is executed by message receive from other Eval-net process, just after the number of tokens in *receive location* is increased,
- *send function*—is responsible for message preparing and sending to another Eval-net process.

In the Eval-net, the diagram flow and source code execution is performed in seperate threads (Fig. 3). When the diagram fires the transition, the invoke is given to the thread pool to execute assigned procedure. The number of fired transitions may be not limited, if the thread pool allows.

3.2 Compatibility with Existing Tools

Linking transition with the procedures and *special places* does not cause interference with the old structure of the Petri net. Implementing a system modeled in the Petri net diagram it must be separated into individual components first. Connections between elements should be replaced by pairs *message sender* and *message receiver*. In this way, we obtain a separate Eval-net diagrams to be combined with the source code element.

Reverse process, menas export from the Eval-net to the Petri net executes as

1. Combine Eval-net diagrams of multiple system elements into one diagram.
2. Remove all *resolution location*.
3. Glue pairs of *message sender* and *message receiver*, such one having identical tag.

Example of such export was there is on Fig. 4.

Extensions, like Colored Petri net [20] allow to data in tokens and to implement conditional functions in transitions. Unfortunately it is not combining flow control with execution of procedures in element. It is more advanced modeling algorithm only. In addition, Eval-net extensions do not conflict with Colored Petri net extensions, therefore, it is possible to use the power of both.

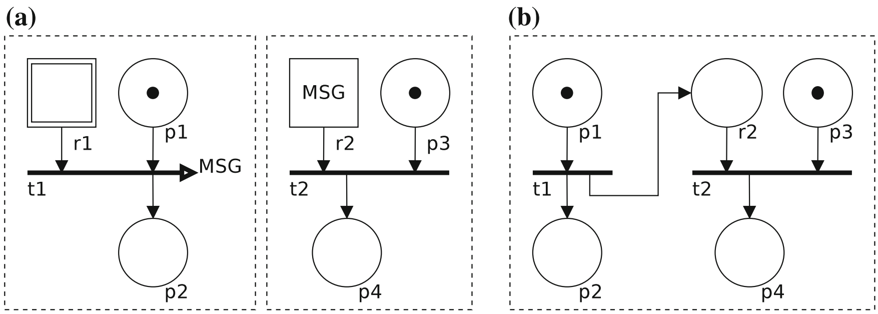


Fig. 4 Transform two Eval-nets with message sender and message receiver (**a**) to one Petri net (**b**)

4 Example of Eval-net

For testing purposes, two Eval-nets were implemented (Fig. 5). These nets represent the behaviour of two agents, which negotiate by FIPA Contract Net Interaction Protocol [8]. The first agent has an initiator role, second agent has a participant role.

The initiator wants to have the task be performed by the participant. The participant is ready to work.

The initial marking of both nets is $M_{0,int} = [0, 0, 0, 0, 1]$ and $M_{0,part} = [1, 0, 0, 0]$ where $M_{0,int}$ vector corresponding to the marking of $[p_1, p_2, p_3, p_4, f]$ and $M_{0,part}$ vector corresponding to the marking of $[p_5, p_6, p_7, p_8]$. t_1 and t_8 transitions fulfil pseudoenabled rules and turn into active states. This makes the evaluation of r_1 and r_8 begin.

r_8 is waiting for CFP message and blocking of firing t_8 transition. In another side, r_1 preparing data for call for the proposal (CFP) message which must contain the preconditions of the task.

After preparing the preconditions, for example "buy me a train ticket from Krakow to Warsaw on August 10th between 6:00 and 10:00 AM", the r_1 end of evaluation and t_1 transition fulfil full enabling rules. Now, the t_1 transition can be fired.

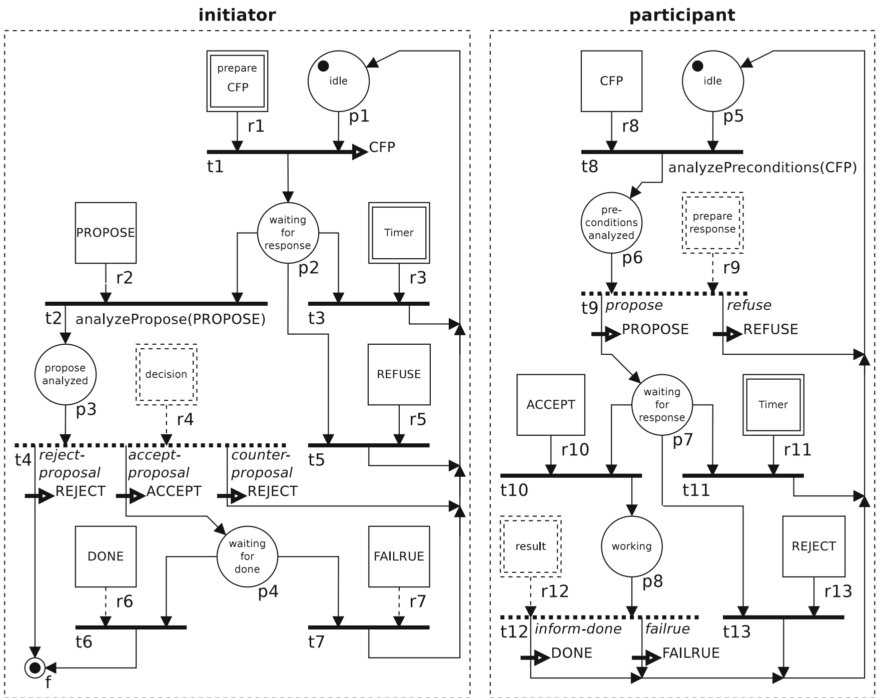


Fig. 5 Sample FIPA Contract Net Interaction Protocol [8] implementation as two Eval-nets. First net is executing by initiator agent, second net is executing by participant agent

After firing t_1, the message sender sends the CFP message to the participant.

The participant receives CFP and fires the t_8 transition. Now r_9 looks for trains. If it finds some t then make proposals (offers) the parameters of which are railway connections and ticket prices. The proposals are stored in local variables as prepared in the PROPOSAL message. After firing t_9 the PROPOSE message is sent.

Otherwise, if trains are not found, the participant answers with the REFUSE message.

If the initiator receives the REFUSE message then it returns to its initial state. Otherwise, if it receives the PROPOSE message then it analyses its offers and makes a decision: accept propose (buy ticket), order cancellation (buying ticket resignation) or change the preconditions (choose other day or hours). These choices were represented by flow condition functions of output arcs from t_4 transition. Backend implementation assumes only one of their functions will return non 0 (in other nets it may be otherwise). So it is a switch-case like transition. If the initiator decides to cancel then it sends the REJECT message and goes to the final state (token in f place). If the initiator wants to change the precondition (wants to send counter offer) then it sends REJECT message too and goes back to the initial state. If the initiator accepts the proposal then it sends the ACCEPT message with ticket parameters and turns to waiting for result.

The participant receives response for proposals. If it receives REJECT then it goes to initial state. Otherwise, if it receives ACCEPT then it goes to work and evaluation of r_{12}. If work is finished (ticket bought) then the evaluation of r_{12} is finished too and the DONE message is sent. In the case of failure (no empty places on the train), then the evaluation of r_{12} is finish and the FAILURE message will be sent. In both cases, participants return to init sate.

When the initiator receives the DONE message then it goes to the final state. Otherwise, it returns to the initial state.

5 Running of Example Eval-nets

The log result of running was presented on Table 1. The I.f. column presents the firing of the transition. It may be referred to as the initiator event. The I.m. column presents marking for initiator places. It may be referred to as the initiator state after event. The P.f. and P.m. are the equivalents for the participant. The message sending column presents the name, direction and body of the sending message.

In the beginning the initiator wanted to buy a train ticket from Krakow to Warsaw on 10th August between 10:00 and 11:00 AM, but the participant could not find any train and refused. Then the initiator tried again with 9:00 and 10:00 AM. The participant offered one train. However its offer was too expensive and initiator rejected it and tried again with 7:00 and 9:00 AM. The participant offered two trains, the first cheap and the second expensive. The initiator selected the cheap train and accepted its offer. The participant bought the ticket and replied done. The initiator went to finish state, but the participant went to initial state and it is ready for another initiator.

Table 1 Log of both example Eval-nets flow

Step	I.f.	I.m.	Message sending	P.f.	P.m.
0	—	[1, 0, 0, 0, 0]		—	[1, 0, 0, 0]
1	t_1	[0, 1, 0, 0, 0]	**CFP** → {buy for me train ticket from Krakow to Warsaw on August 10th between 10:00 and 11:00 AM}	—	[1, 0, 0, 0]
2	—	[0, 1, 0, 0, 0]		t_8	[0, 0, 0, 1]
3	—	[0, 1, 0, 0, 0]	← **REFUSE**	t_9	[1, 0, 0, 0]
4	t_3	[0, 0, 0, 0, 1]		—	[1, 0, 0, 0]
5	t_1	[0, 1, 0, 0, 0]	**CFP** → { [. . .] between 9:00 and 10:00 AM }	—	[1, 0, 0, 0]
6	—	[0, 1, 0, 0, 0]		t_8	[0, 0, 0, 1]
7	—	[0, 1, 0, 0, 0]	← **PROPOSE** { Offer 1: IEC train dep. 9:40 AM, arr. 12:41 AM, price: 127.00 PLN }	t_9	[0, 0, 1, 0]
8	t_2	[0, 0, 1, 0, 0]		—	[0, 0, 1, 0]
9	t_4	[0, 0, 0, 0, 1]	**REJECT** →		[0, 0, 1, 0]
10	t_1	[0, 1, 0, 0, 0]	**CFP** → { [. . .] between 7:00 and 9:00 AM }	—	[1, 0, 0, 0]
11	—	[0, 1, 0, 0, 0]		t_8	[0, 0, 0, 1]
12	—	[0, 1, 0, 0, 0]	← **PROPOSE** { Offer 1: IR train dep. 7:42 AM, arr. 11:16 AM, price: 53.90 PLN Offer 2: IEC train dep. 8:01 AM, arr. 10:58 AM, price: 127.00 PLN }	t_9	[0, 0, 1, 0]
13	t_2	[0, 0, 1, 0, 0]		—	[0, 0, 1, 0]
14	t_4	[0, 0, 0, 1, 0]	**ACCEPT** → { Offer 1 }	—	[0, 0, 1, 0]
15	—	[0, 0, 0, 1, 0]		t_{10}	[0, 0, 0, 1]
16	—	[0, 0, 0, 1, 0]	← **DONE**	t_{12}	[1, 0, 0, 0]
17	t_6	[0, 0, 0, 0, 1]		—	[1, 0, 0, 0]

Shorts: I.—initiator, P.—participant, f.—firing, m.—marking after firing

6 Conclusion

In this article, an idea of a solution mixing the advantages of Petri nets and evaluation nets was presented. This idea was based on an extension of Petri net for elements of evaluation nets and adding a backend layer where source code is executed. This code operates on data saved and transferred data structures sent between networks.

This mix gives a simple to implement extension. Simplicity is available because it is based on solutions built for Petri nets and does not create conflicts with many existing extensions for Petri nets.

The described solution is a part of a simulation platform that consists of Eval-net interpreter, editor and tools for simulation and analysis. The interpreter runs within an agent that acts on the JADE platform. It passes net evaluation (created in the editor) and executes the source code backed (written in Python). The editor allows you to create and view the current state of the network together with the history of transitions. It also allows you to make changes to on-line in both the frontend and the backend. The analysis tools are adapted from Petri net tools, while the simulator allows you to perform a specific scenario, to generate test data and visualize simulation results.

Acknowledgments Author Michał Niedźwiecki would like to thank EFS (European Social Fund, pol. Europejski Fundusz Społeczny) of POKL (Operational Programme "Human Resources Development", pol. Program Operacyjny "Kapitał Ludzki") 4.1.1 (POKL.04.01.01-00-367/08-00), European Union programme for support.

References

1. André, F., et al.: Distributed Computing Systems: Communication, Cooperation, Consistency. Elsevier Science Publishers, New York (1985)
2. Balbo, G.: Introduction to stochastic Petri nets. In: Brinksma, E., Hermanns, Holger, Katoen, Joost-Pieter (eds.) Lectures on Formal Methods and Performance Analysis. Lecture Notes in Computer Science, vol. 2090, pp. 84–155. Springer, Heidelberg (2001)
3. Cabasino, M., Giua, A., Pocci, M., Seatzu, C.: Discrete event diagnosis using labeled Petri nets: an application to manufacturing systems. Control Eng. Pract. **19**(9), 989–1001 (2011)
4. Christensen, S., Hansen, N.D.: Coloured Petri nets extended with place capacities, test arcs and inhibitor arcs. Applications and Theory of Petri Nets. LNCS, vol. 691, pp. 186–205. Springer, Berlin (1992)
5. Dingle, N.J., Knottenbelt, W.J., Suto, T.: PIPE2: a tool for the performance evaluation of generalised stochastic Petri Nets. SIGMETRICS Perform. Eval. Rev. **36**(4), 34–39 (2009)
6. Dufourd, C., Finkel, A., Schnoebelen, P.: Reset nets between decidability and undecidability. In: Larsen, K.G., Skyum, S., Winskel, G., (eds.) ICALP 1998. LNCS, vol. 1443, pp. 103–115. Springer, Heidelberg (1998)
7. Eckleder, A., Freytag, T.: WoPeD—A tool for teaching, analyzing and visualizing workflow nets. Petri Net Newsletter 75 (2008)
8. FIPA: FIPA Contract Net Interaction Protocol Specification. FIPA (2001)
9. He, X.: A formal definition of hierarchical predicate transition nets. In: Billington, J., Reisig, W. (eds.) Application and Theory of Petri Nets 1996. Lecture Notes in Computer Science, vol. 1091, pp. 212–229. Springer, Berlin (1996)
10. Heitmann, F., Moldt, D.: Petri Nets Tools and Software (Accessed in 20130514). http://www.informatik.uni-hamburg.de/TGI/PetriNets/tools/
11. Holloway, L., Krogh, B.: Controlled Petri nets: a tutorial survey. In: Cohen, G., Quadrat, J.P. (eds.) 11th International Conference on Analysis and Optimization of Systems Discrete Event Systems. Lecture Notes in Control and Information Sciences, vol. 199, pp. 158–168. Springer, Berlin (1994)
12. Janoušek, V., Kočí, R.: PNtalk project: current research direction. In: Simulation Almanac 2005. pp. 50–62. Faculty of Electrical Engineering, Czech Technical University (2005)
13. Köhler, M.: Object Petri Nets: Definitions, Properties, and Related Models. Univ, Bibliothek des Fachbereichs Informatik (2003)
14. Niedźwiecki, M., Rzecki, K., Cetnarowicz, K.: Using the evaluation nets modeling tool concept as an enhancement of the Petri net tool. In: 2013 Federated Conference on Computer Science and Information Systems (2013)
15. Niedźwiecki, M., Rzecki, K., Cetnarowicz, K.: Complex negotiations in the conclusion and realisation of the contract. Procedia Comput. Sci. **18**(0), 1525–1534 (2013)
16. Noe, J.D., Nutt, G.: Macro E-nets for Representation of Parallel Systems. IEEE Trans. Comput. **C-22**(8), 718–727 (1973)

17. Nutt, G.J.: Evaluation nets for computer system performance analysis. In: Proceedings of the fall joint computer conference, part I. pp. 279–286. AFIPS ’72 (Fall, part I), ACM, New York, NY, USA, 5–7 December 1972
18. Petri, C.A.: Kommunikation mit automaten. Ph.D. thesis, Universität Hamburg (1962)
19. Ramchandani, C.: Analysis of Asynchronous Concurrent Systems by Timed Petri Nets. Technical Report, Cambridge, MA, USA (1974)
20. Ratzer, A.V., Wells, L., Lassen, H.M., Laursen, M., Qvortrup, J.F., Stissing, M.S., Westergaard, M., Christensen, S., Jensen, K.: CPN tools for editing, simulating, and analysing coloured Petri nets. In: van der Aalst, W.M.P., Best, E. (eds.) Applications and Theory of Petri Nets 2003. Lecture Notes in Computer Science, vol. 2679, pp. 450–462. Springer, Berlin (2003)
21. Reinhardt, K.: Reachability in Petri nets with inhibitor arcs. Electron. Notes Theor. Comput. Sci. **223**, 239–264 (2008)
22. Rozenberg, G., Engelfriet, J.: Elementary net systems. In: Reisig, W., Rozenberg, G. (eds.) Lectures on Petri Nets I: Basic Models. Lecture Notes in Computer Science, vol. 1491, pp. 12–121. Springer, Berlin (1998)
23. van Hee, K., Oanea, O., Post, R., Somers, L., van der Werf, J.M.: Yasper: a tool for workflow modeling and analysis. In: Proceedings of the Sixth International Conference on Application of Concurrency to System Design. pp. 279–282. ACSD ’06, IEEE Computer Society, Washington, DC, USA (2006)
24. Żabińska, M., Cetnarowicz, K.: Application of M-agent multi-profile architecture and evaluation nets to negotiation algorithm design. In: Amin K.Y.M. (ed.) The 2nd IEEE International Symposium on Signal Processing and Information Technology, ISSPIT’2002, pp. 214–219, 18–21 December 2002
25. Zurawski, R., Zhou, M.: Petri nets and industrial applications: a tutorial. IEEE Trans. Ind. Electron. **41**(6), 567–583 (1994)

Part III
Creativity

A Japanese Problem-Solving Approach: The KJ Ho Method

Susumu Kunifuji

Abstract In Japan, by far the most popular creative problem-solving methodology using creative thinking is the KJ Ho method. This method puts unstructured information on a subject matter of interest into order through alternating divergent and convergent thinking steps. In this paper, we explain basic procedures associated with the KJ Ho and point out some of its most specific applications.

Keywords Creative thinking · W-shaped problem-solving methodology · The KJ Ho method

1 Outline

There are a number of creative thinking methods in existence, for example, brainstorming, brain writing, mind mapping, the KJ Ho (method) [1], the NM method, the equivalent transformation method, etc. Human thinking processes for creative problem solving typically consist of four subprocesses: divergent thinking, convergent thinking, idea crystallization, and idea verification. In Japan, the most popular creative thinking method by far is the KJ Ho [6].

Jiro Kawakita explained in his paper published in 1991 [5] that human creative problem-solving processes often consist of the following steps (see Fig. 1 on the next page).

In a scientific inquiry, one encounters a problem at point A on the thought level. As the first step in solving this problem, he proceeds to explore the situation surrounding the problem between A and B, and next collect all relevant and

---

KICSS'2013 Invited Lecture [6].

S. Kunifuji (✉)
Japan Advanced Institute for Science and Technology, Kanazawa, Japan
e-mail: kuni@jaist.ac.jp

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_13

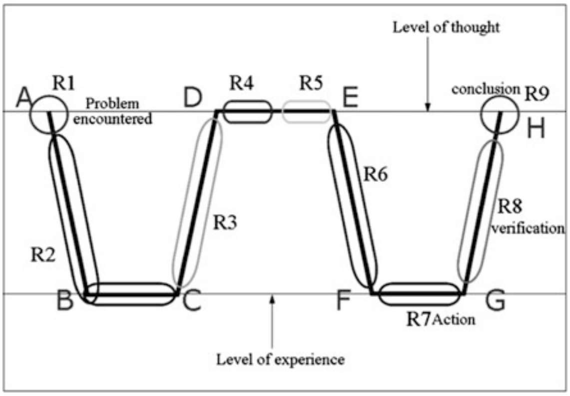


Fig. 1 The W-shaped problem-solving methodology [5]

accurate data through observation between B and C. By this data, he next formulates or develops a number of hypotheses between C and D. Having returned to the next level, at point D, he next evaluates hypotheses and decides one hypothesis which to adopt. Between D and E, he infers and revises the adopted hypothesis through deductive reasoning. Next, he plans an experiment for testing the adopted hypothesis between E and F, and observes the experiment between F and G. Given the results of the experiments, he can verify hypothesis between points G and H, and can finally acquire a correct conclusion at point H.

The very original nature of the KJ Ho is a bottom-up creative problem-solving methodology for practical use (almost all other creative problem-solving methodologies are top-down). The KJ Ho was created on the basis of cultural anthropology and the W-shaped problem-solving methodology, as depicted in Fig. 1, providing a unique integration of social and natural sciences.

2 Basic Steps

Now we can explain basic procedures associated with the KJ Ho: these are label making; label grouping; group label naming; nesting grouping and naming; spatial arrangement and relationships, and verbal or written explanation (see Fig. 2 below).

Label making: by brainstorming, brain writing, or idea marathon, etc. through thinking and fieldwork. You collect data, observe situations related to the problem, and write down everything that you have discovered on labels. Only one fact, thought, or concept related to the problem of concern should be written on each label. There is no limit to the number of written labels.

Label grouping: After writing up all facts, you should stack the labels and spread them around on a table or a floor. You must carefully consider what the labels are saying. Labels that appear to belong together should be arranged close to each other and kept at a distance from other labels to form a group (or an island).

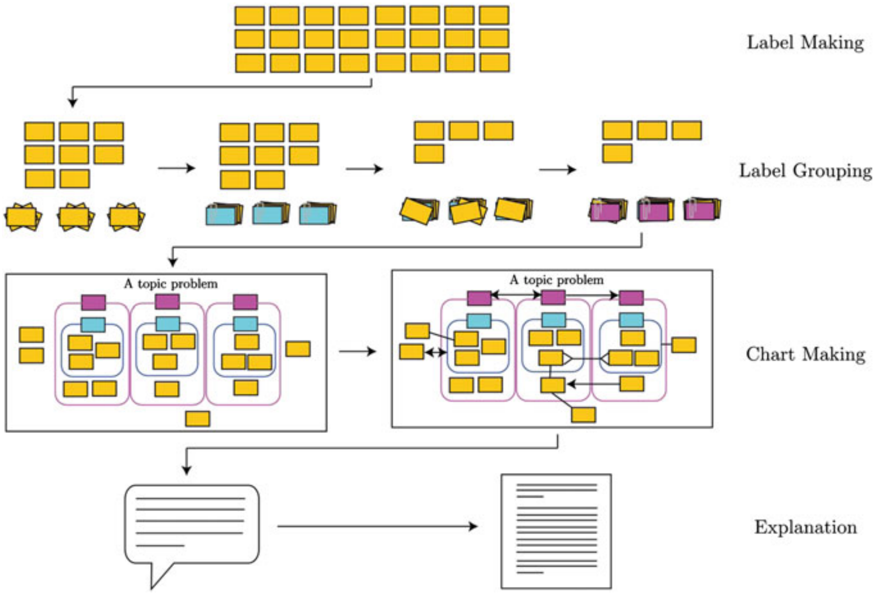


Fig. 2 Basic steps of the KJ Ho method [8]

Note that labels should not be grouped simply based on similarity (i.e., similar words being used), but rather on mental association. Labels that do not seem to be related to any other labels (called 'lone wolves') might become key concepts, or be merged into another group at a higher level of label grouping.

Group label naming: After about two-thirds of the labels have been grouped, you can start making one-line headers (summarized sentences) as titles for each grouping. You should reread all labels in the grouping and then think of a suitable title to describe the essence of all labels in that grouping. It should not be too abstract, nor too specific. Once a title is decided, you should write it on a new label and perform this process until all groupings have been done. Making a one-line title is a concept creation or an abduction, which includes the meaning of all underlying labels in the group. This process of label grouping and label naming is repeated until the number of groupings becomes less than an empirically intuitively determined number, say, ten.

Spatial arrangement: After several label grouping steps, a bundle of final groupings is obtained, which are to be spatially spread and arranged on a large sheet of paper. You should consider carefully a stable, meaningful arrangement of all labels and groupings. This process will continue until all labels and groups are placed using the same justification recursively.

Relationships: Labels, groups, teams, and islands are often called objects. Typically, the following relationship symbols are used in KJ Ho chart making between objects:

(a) Cause and effect: One is a predecessor or a cause of another.
(b) Contradiction: Objects are conflicting to each other.
(c) Interdependence: Objects are dependent on each other.
(d) Correlation: Both objects relate to one another in some way.

Relationships among objects in the chart must be easily and clearly expressed.

Verbal or written explanation: The last step is to explain the chart clearly. The explanation should begin with a general scenario of the problem, and then be more specific. Usually, the verbal explanation is given first, which could start from any position on the chart, and then proceed to an adjacent part until all parts are covered. After this, the written explanation is performed. You should write down the verbal explanation, making it clear, smooth, and concise. This step helps the audience to understand the interrelationships among components of the problem thoroughly. The written explanation of the chart (that is called Type A) is called the KJ Ho Type B and the verbal one that is preliminary is the KJ Ho Type B'.

3 W-Shaped Problem-Solving Method

The KJ Ho is a creative problem-solving methodology using creative thinking that puts unstructured information on a subject matter of interest into order through divergent and convergent thinking steps. As it was briefly mentioned in the introduction, the process model of the KJ Ho (the cumulative KJ Ho) is called 'W-shaped problem-solving' from Round 1 to 9 as follows (see Figs. 1 and 2 above).

Round 1: Presenting a given problem by clarifying the given task. Data is collected by recalling from memory, rather than from external investigations.

Round 2: Understanding the status quo of the given problem, i.e., understanding the current situation. Data is collected by fieldwork or observations from a 360° viewpoint.

Round 3: Hypothesis generation. This round is most important because we must find new hypotheses or ideas to solve the given problem by the KJ Ho. Note that if generated sentences (hypothesis or ideas) are in a negative form, then it is better to restate these as positive sentences. This is called the nega-posi transformation.

Round 3.5: After evaluating hypotheses or ideas by both subjective and objective judgments, you decide the most effective hypothesis to solve the given problem, which is in effect a decision making or a policy determination. This evaluation subprocess is often done by all participants, and sometimes called a highlighting or scoring process.

Round 4: Forming a grand plan, i.e., formulating a master plan for more detailed investigations.

Round 5: Forming a detailed plan, i.e., an implementation strategy for the problem.

Round 6: Establishing a practical procedure using PERT (Program Evaluation and Review Technique) or workflow.

Round 7: Practical actions to verify the hypothesis.

Round 8: Verification by testing. If the hypothesis is rejected, then retry another hypothesis generation by backtracking to Round 3.5. If it is acceptable, move to next Round 9. If all hypotheses are rejected, then go back to the checkpoint of Round 1.

Round 9: Conclusion and Reflection process. Accepted hypothesis must be added to our knowledge base.

4 Applications of the Cumulative KJ Ho

The Nomadic University [3, 4] is a form of project where the cumulative KJ Ho as a W-shaped problem-solving method is applied to a specific problem. These Nomadic Universities are typically held at the locations where the problems exist. The first Nomadic University was founded by Prof. Jiro Kawakita (then, a Professor at Tokyo Institute of Technology) in August 1969, in Kurohime, Nagano, in the midst of student uprisings in Japan. This first event was called the Kurohime Nomadic University [3], a 2-week workshop in an outdoor campus, with attendants staying in tents. The author participated in this first event and this series continued to be held twenty-five times in Japan up until 1999.

Further to this, there have also been numerous projects, seminars, and mini Nomadic Universities [8] (shorter events, from 2008), which still take place today covering a very wide range of domains, including management and administration, energy and environment, health services, psychotherapy, regional development, engineering, construction, etc. in Japan.

References

1. Kawakita J., Hasso-Ho (発想法), Chuko Shinsho, no. 136 (in Japanese) (1966)
2. Kawakita, J., Makishima, S.: Problem Solving—KJ Ho Workbook ((問題解決 - KJ法ワークショップ)) Kodansha, Japan (in Japanese) (1970)
3. Kawakita, J.: Clouds and Water—Nomadic University Chronicle of Hard Working ((雲と水と - 移動大学奮戦記)) Kodansha, Japan (in Japanese) (1971)
4. Kawakita, J. (ed.): Nomadic University—Japanese Islands are my Textbook ((移動大学 - 日本列島を教科書として)). Kashima Kenkyujo Shuppankai, Japan (in Japanese) (1971)
5. Kawakita, J.: The Original KJ Method, Revised edn. Kawakita Research Institute (1991)

6. Kunifuji, S.: A Japanese problem solving approach: the KJ-Ho (Method). In: Skulimowski, A. M.J. (ed.) Looking into the Future of Creativity and Decision Support Systems: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems, Kraków, Poland, 7–9 Nov 2013 (Invited lecture). Advances in Decision Sciences and Future Studies, vol. 2, pp. 333−338. Progress & Business Publishers, Kraków (2013)
7. Kunifuji, S., Kato, N., Wierzbicki, A.P.: Creativity support in brainstorming. In: Wierzbicki, A. P., Nakamori, Y. (eds.) Creative Environment: Issues of Creativity Support for the Knowledge Civilization Age, Studies in Computational Intelligence (SCI) vol. 59, pp. 93−126. Springer (2007)
8. Naganobu, M., Maruyama, S., Sasase M., Kawaida S., Kunifuji S., Okabe A. (the Editorial Party of Field Science KJ Ho): Exploration of Natural Fusion, Thought and Possibility of Field Science. Shimizu-Kobundo-Shobo, Japan (in Japanese) (2012) (Note: Published as a Kawakita Jiro Memorial Collection issue covering various topics of Fieldwork Science and the KJ Ho)
9. Viriyayudhakorn, K.: Creativity assistants and social influences in KJ-method creativity support groupware. Dissertation of School of Knowledge Science, JAIST, Mar 2013

Visual Anonymity in Online Communication: Consequences for Creativity

Thomas Köhler

Abstract Online communication (OC) is widely used for building social relations as well as for exchanging information in both private and professional settings. Having been conceptualized since the 1980s as "Computer-mediated communication" (CMC) within a number of disciplines, from psychology to communication, different predictions on how the users' self may be affected were made [14]. However, the potential of OC toward creativity enhancement is rarely discussed, even though the effects on the individual's behavior are obvious. The paper conceptualizes how in the "asocial" setting of using recent ICTs opportunities for an abnormal behavior may occur. Some of these discourses even observe a new potential for creative self-constructions. Based upon a series of experiments, this paper investigates how the social self develops creatively in OC, i.e., how such has a specific meaning for creativity development. Identifying further opportunities for creative co-constructions in forms of OC, the author follows a connectivist interpretation. This paper has been written as a revised and extended version of the paper presented at KICSS'2013 [15].

Keywords Online communication · Co-construction · Creativity · Self-perception · Computer-mediated communication · Visual anonymity

T. Köhler (✉)
Faculty of Education and Media Center, Educational Technology Chair
at the Institute for Vocational Education, TU Dresden, Dresden, Germany
e-mail: thomas.koehler@tu-dresden.de

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_14

1 Theoretical Considerations

1.1 Influences of Information and Communication Technologies of the User's Identity and Creativity

In recent years, information and communication technologies (ICT) have taken an increasing position of utmost importance in our private and professional environments. A great number of such social communities are nowadays inconceivable without this technology, while other forms emerged as completely new, e.g., chat rooms, online learning spaces, or virtual communities like Facebook. The term computer-mediated communication (CMC) was coined in psychological research and adopted by sociology, communication sciences, and other academic disciplines as well.

The online communication user's self is treated as a theoretical construct describing a hypothetical cognitive structure. A theoretical background of the user's self advises the existence of at least two core components: a personal (individual) and a social identity [19]. Whereas the individual identity involves the idiosyncratic aspects of a person, such as intellectual skills or personal attitudes, the social identity resembles a sum of social identifications of one person resulting from affiliations with social communities, e.g., groups, families, and communities. Whenever the term self is mentioned here, both components should be considered. Below the focus is on the aspects related to social identity.

One of the inspirations for the study reported here was an observation that modeling which describes the changes of the self in online communication has to be distinguished on specific conditions of CMC in three types of influences:

1. The characteristics of the respective communication channel;
2. The individual versus the mutual use of ICT;
3. Familiarity and experience with ICT.

The following picture illustrates this observation (Fig. 1)

Models of communication science increasingly reveal media effects, i.e., the influence of specific characteristics of the particular computer-mediated communication channel. As early as 1976, the Social Presence Theory [26] arrived to the conclusion that, in principle, social perceptions are influenced by the

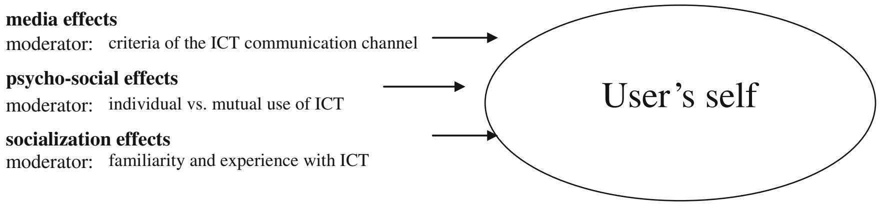


Fig. 1 Influences on the user's self in online communication [14] (translated by the author)

communication channel. The Social Cues Filtered out Hypothesis [28] accentuates the significance of missing social corollary information. The Concept of Social Information Processing [35] spots the potential for new, so-called "hyper-personal" possibilities of individual communication in this scarce situation.

On the other hand, the influence of individual versus mutual usage of ICT on identity is particularly chosen as a central factor in psychological research models. One such model is the Social Identity Deindividuation Theory [27], which describes the interplay between deindividualization and depersonalization. For example, the JEMCO Workshop Study [2] disputes models for group development, whereas [13] describes the effects on a perceived publicity. In addition, studies on interpersonal perception in intergroup interactions [31] as well as the experiments of McKenna and Bargh [18] on identity demarginalization are quite significant.

Finally, effects of familiarity and experiences with ICT, i.e., a kind of socializing effect, are discussed in the studies as [3, 4, 11, 21] and others. These perspectives act jointly to describe the development of civilizations in computer-mediated communities emerging over a longer period of use. Therefore, a long-term cultivated establishment of several artificial personalities of one real person—cp. [3, 16, 32, 33] —shall only be mentioned marginally. So far, distinguishing between familiarity and experience is reasonable as both concepts depend, on the one hand, on socialized components. On the other hand, however, experience is a cognitive category, which includes the knowledge about the use of the technology, whereas the familiarity deals with the affective category, seizing the emotional attitude toward the technology.

1.2 Visual Anonymity as Entry to Creativity—Findings from Different Fields of Online Communications

Already in 1973 Gergen et al. described in their paper "Deviance in the dark" the influence of visual anonymity on individual behavior [9]. Indeed it has been observed that within such an environment the individual behavior becomes much more open, not controlled by visual perceptions of a relatively normative surrounding. In other words: creative social behavior is invited! Meanwhile online communication has become a widely used activity for private settings, professional exchange, or education. Three examples may illustrate the current situation:

(A) Facebook as extension of the private sphere: With its 10th anniversary in 2014, Facebook is obviously a rather stable and, by number of users, the most widely used online communication channel worldwide. It would go far beyond the scope of this paper to discuss the meaning and functioning of Facebook in detail. However, interesting for our focus is the usage of the term 'Face.' Even though this term was coined by the practice of US colleges to produce books with portraits of all students' faces annually, the face is as well a symbol for being visible; however, there may not exist a direct connection to the person who is viewing this face. Slogans like "make yourself visible on

Facebook" do indeed have a much deeper meaning than its authors probably intend, as visual anonymity is a key component of Facebook. Academic literature has highlighted the role of the faces of so-called 'Facebookers' for their social enhancement [37] as well as the 'Privacy Paradox' in the social web in reaction to self-disclosure [30].

(B) Social media in the academic classroom: At the authors' institution, Technische Universität Dresden, the local data protection policy makers decided in 2010 not to allow students and teachers to upload individual portraits to the local learning management system anymore, which was being used by approximately 20,000 students at the time. The fear was that the portraits, which would be seen independently of a student's physical presence, would lead to abuse. For example, misuse either in the case of a direct correlation between the individual preference of a portrait by the teacher and the mark given, or the potential insertion of the portraits into private databases. What sounds like the very early intentions of Facebook are indeed assumptions about the serious influence of the online communicator's visual anonymity toward their individual and social behaviors. Academic literature described similar findings already in the year 2000 when investigating the effects of different multimedia communication channels on students' learning outcomes [22] and authors were able to demonstrate the independent correlations of the student–teacher relation and the learning outcome. To sum it up: the teacher's virtual presence matters.

(C) Professional exchange online: cooperation of competitors: In her thesis about cooperation in the Berlin media startup scene, Antoine [1] concludes, "The new creative economy provides impetus for the media networked cultural productivity, which may arise from the novelty in the form of new communicative orientations in these networks" (pg. 195, translated by the author). This finding is similar to the observation described by Kahnwald and Köhler [12], whose online community of practice for safety professionals links several thousand experts Germany-wide without a serious fear of abuse. Which role does the visual anonymity play here? Following the theory of Sproull and Kiesler [28], exactly this anonymity is rather inhibitive for participating in dialogs with other like-minded users—which is typical for a community of practice. Additionally, the online channel does help to bridge the geographic distances as well as the interfirm segregations that would both be serious barriers for a face-to-face collaboration.

2 Hypotheses

Users of online communication should show an increased self-openness: Since CMC is subject to different characteristics of the respective communication channel (e.g., e-mail versus chat rooms, etc.). A perception of the differences among these channels should be verifiable. In accordance with the basic approach of

Kiesler and Sproull [28, 29], we hypothesized that the reduced social context cues compared to face-to-face communication will lead to an increased openness in CMC. This assumption may be further supported by the observations of McKenna and Bargh [18], whereupon anonymous participation, e.g., in newsgroups, enables a reliable exchange of so-called "marginal identities."

- Hypothesis 1 (H1): Online communication is generally rated more open than face-to-face communication.

Experience changes the social perception: In his concept of "Hyperpersonal Communication," Walther [36] mentions the user's media experience as a major influencing factor. According to him, users are able to develop their abilities in handling online communication with an expanding know-how. He calls them "hyperpersonal," as they have developed a potential which goes further than face-to-face communication. Also, [24] showed that the attitude toward a medium is a core element which may be changed by the process of media use. This attitude exists mutually with a definite amount of skills before the first use, but it may also be unconstrained by those skills. Otherwise, the SIDE-Model [27] shows us that members of (online) communities categorize themselves and therefore view other users as a reference group to strengthen their in-group identity. This leads to an increased sense of community, the so-called "collective self-esteem." By combining these observations, we can reason that the collective self-esteem of online communicators depends mainly on the experience with this technology.

- Hypothesis 2 (H2): Users who rank themselves as highly sophisticated in Internet and online communication also rate their "collective self-esteem" as computer users more positively than users who rank themselves as less sophisticated.

Publicity (even a virtual one) changes the self-attention: Few of the already mentioned concepts [12, 36] deal with the modified perceived publicity as subject to CMC. So far, an amplified publicity leads to a rectified self-acceptance of the user, e.g., as shown in the SIDE approach [23] and especially by [18]. It occurs as follows: the focused attention of the user during online communication often includes the social environment in fewer dimensions than is normally the case. We can assume that the user's self-attention will increase with diminishing publicity, namely depending on the normative influence of the social context. This should apply according to the virtual and real publicity of online communication. Most significant is the fact that online the user community is only presented virtually, by text, video—and eventually as avatars. Upon other terms, it is also possible that additional users are physically present: e.g., in computer pools of universities, in Internet cafes, or in open-space offices. Our assumptions lead to the statement that the perceived publicity, whether physically or virtually presented, will change the self-attention of online communicators.

- Hypothesis 3 (H3): The more publicity a user perceives, the smaller their self-attention is. This effect is moderated by the user's experience.

3 Methods and Results

With an experimental approach, the proof of the effects specified should be possible. In order three experiments were conducted, using multigroup designs which were established specifically for the respective conditions. To verify the first hypothesis (H1), 60 subjects were interviewed electronically (with Email) using the Rochester Interaction Survey [25]. The Collective Self-Esteem Survey [17, 34] was used to test the second hypothesis (H2), and for hypothesis three (H3) three experimental groups were confronted with different publicity conditions and compared.

3.1 Experiment 1—Increased Publicity in Online Communication

The foundation of the first experiment was the Rochester Interaction Survey, which was translated from English into German by Schumacher [25] and was validated with a student population for face-to-face communication (84 subjects of both genders aged 20–31 years). The survey consists of a 7-step scale for each of the seven effect variables: intimacy, own openness, openness of others, quality, contentment, initiative/influence, and selected basic personal data (length of the interaction, point of time, gender, type of interaction). Correlation between the scales is the basis for pooling the two parameters "quality of social interaction" and "controlling social interaction," which have not been considered in the present work.

In addition to a paper survey, the survey was also generated and utilized with a comparable design in a digital format per e-mail. The following effects of the channel on perceived publicity communication were demonstrated (Table 1).

These findings on the perception of openness and publicity in different channels of mainly online communication can be summarized as follows:

- Own publicity is estimated to be significantly higher in MUD, e-mail, and for electronic surveys than the publicity of others.
- Publicity of others is perceived higher than the own publicity for two communication settings, face-to-face communication and IRC, whereas only the result for IRC with $p = 0.057$ is not significant.
- It shows that in all 1×1 communications, the own publicity is estimated significantly higher than the publicity of others (the tested MUD was a 1×1 setting).
- In all $1 \times n$ (with $n > 1$) communications, e.g., in IRC, the publicity of others is estimated higher than the own publicity.
- Surprisingly, the values for ftf-communication, which was dyadic, show an increased publicity of others.

Table 1 One's own publicity versus publicity of others while using different communication channels

				Own publicity					Relation
				E-mail	E-survey	MUD	IRC	ftf-C	
		Parameter		m 5.2	m 6.0	m 3.7	m 3.1	m 4.5	
				sd 1.2	sd .8	sd 1.4	sd 1.6	sd 1.3	
Publicity of others	E-mail	m 3.9	sd 1.7	[b]					OwO > OtO
	E-survey	m 4.9	sd 1.4		[b]				OwO > OtO
	MUD	m 2.9	sd 1.8			[a]			OwO > OtO
	IRC	m 3.8	sd 1.9				n.s (0.057)		OwO < OtO

Partially tested: [a] = 0.05; [b] = 0.01

- Consequently, it is retained that under certain circumstances CMC is estimated more open than is the case for 1 × 1 communication.

3.2 Experiment 2—Experience and Social Perception in ICT

It was expected (H2) that CMC users who estimate themselves as highly experienced in using the Internet and CMC rate their collective self-esteem as computer users more positive than the less-experienced user. The collective self-esteem survey, originally generated by Luhtanen and Crocker [17] and translated into German by Wagner and Zick [34], tests the perception of social relationships under the conditions of CMC. The four subscales identity value, membership value, public collective self-esteem and private collective self-esteem include the various facets of the self. For this experiment, the interviewed persons were instructed to imagine a CMC-publicity as their reference group. Subjects were alone at a single PC when they filled out the survey (which was accessed by a WWW-document) without any face-to-face contact with other users. Therewith the interviewed persons were therefore exposed to a testing situation similar to usual conditions of a virtual community. As a first step, a Spearman correlation was calculated to determine the interrelation between experience and the CSE-total value ($r = 0.295$; $p < 0.036$). A positive interrelation can already be seen as a confirmation of the hypothesis. The following figure shows how the self-esteem rises with increasing experience (Fig. 2).

Furthermore, it should be also clarified how the factors recorded by single CSE-partial scales take effect in detail. The total random test was divided into three different stages of experience with Internet and CMC, and the maximum of 183 interviewed people were assigned to these stages as follows (Table 2).

An H-Test according to Kruskal-Wallis was utilized as a parameter-free method due to the scaling level of the data. Again, the test was separately executed for all

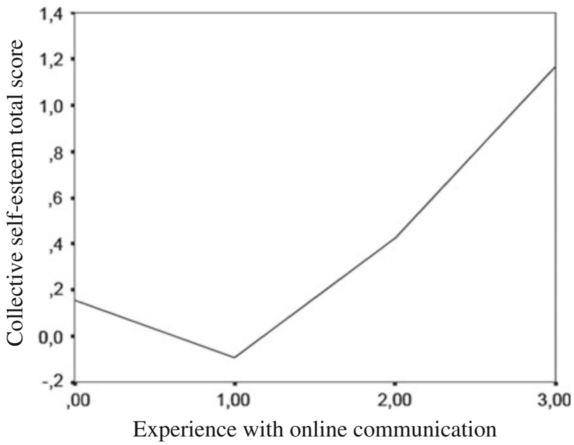


Fig. 2 Influence of the experience on the collective self-esteem

Table 2 Design and respondents per group for experiment 2

		Sub-scales of the collective self-esteem			
		Identity	Private CSE	Public CSE	Membership
Internet experience	Low	67	66	63	66
	Medium	42	43	44	43
	High	72	71	69	71

Table 3 H-Test—subscale identity as a dependent variable

CSE-sub-scale	N	M	SD	Minimum	Maximum	Chi-square	Degrees of freedom	Asym. significance
Identity	182	2.7473	1.2933	1.00	6.25	8.352	2	0.015
Public CSE	177	4.7161	1.0371	1.00	6.75	3.589	2	0.166
Membership	181	4.2528	1.2602	1.00	7.00	27.194	2	0.000
Private CSE	181	5.3191	0.9744	2.25	7.00	8.656	2	0.013

four CSE-partial scales. The following results were found regarding the variable "internet experience" (Table 3).

To summarize:

- There is a positive interrelation between experience and the total value of the collective self-esteem.
- The effect is based on a positive correlation of the experience with three out of four CSE-partial scales: significant for the subscales "identity," "membership," and "private collective self-esteem."
- There is no vital correlation between experience and the subscale "public collective self-esteem".

3.3 Experiment 3—Publicity and Changes of the Self-Attention During ICT Usage

The question of how would self-attention of a CMC user be influenced by the expected publicity during the usage is our last question to be examined. This variable was collected using the survey for dispositional self-attention (SAM) of Filipp and Freudenberg [7]. The SAM survey resembles a modified version of the American self-consciousness scale by Fenigstein et al. [6]. It consists of 27 items, where 13 items cover the private self-attention, while the remaining 14 items include the public self-attention. In order to compare the groups and to prevent possible replications, the survey was shortened to parallel forms that were presented to participants in the respective public conditions. In each case, only one parallel form was presented, which included nine items from the SAM survey. The average scale values were added per participant, and these values were again summarized to average group values that referred to one testing condition. In addition, testing conditions were crossed with parallel forms to prevent direction effects.

It was necessary for the first evaluation to review if the employed partial scales will act similarly before the proper hypothesis could be tested. While using two partial scales of the self-attention scale that were especially made for this comparison (parallel form two and three), we compared them using the Kolmogorov–Smirnow test for the two partial scales for each case under both group conditions. Comparisons of the parallel forms two and three resulted in probabilities of 0.439 and 0.999. This shows that both scales cannot be differentiated significantly for the present testing, both for the group testing condition and for the single testing condition.

The parallel forms were consequently used to test and compare self-attention for both testing conditions. The test for the influence of publicity on self-attention shows a meaningful result: $Z = -3.127$; p(bilateral) $= 0.002$. Thus, the self-attention of participants is higher in single rather than in group situations with a greater publicity. However, it was not possible to differentiate the two subscales of self-attention significantly: "private self-awareness": r [unilateral] $= -0.048$; $p < 0.382$ and "public self-awareness": r [unilateral] $= -0.147$; $p < 0.180$). Another test (Wilcoxon test) determined the influence of the variable publicity on the total value of the collective self-esteem and resulted in a noteworthy effect: $Z = -2.913$; p (bilateral) $= 0.004$.

Recapitulating this, we can verify the following effects:

- The greater the dimension of the publicity experienced by a CMC user, the more self-attention they show.
- It is not possible to differentiate between private versus public self-attention.

4 Discussion: Creative Co-construction in Online Communication

The substantial result of the present study lies in the integrated description of different factors that occur simultaneously while using online communication, but which are predicted in different theories. Observed effects can be interpreted as a reference to an extensive sample, where long-term socialized components of the self-esteem change the user, and a relatively short-term effective, psychosocial variable, e.g., the self-attention, is influenced. In conclusion, the channel itself carries decisive importance as the openness of the user changes in direct reliance of parameters of a particular channel.

More specifically, we may conclude that:

1. Obviously single effects characterize the creative potential of online communication; for example, in all 1 × 1 online communications one's own publicity is estimated significantly higher than the publicity of others. This is of course a paradox because in a communication of only two persons it is impossible that in average each person claims to be more open than the other one. On the other hand, it was shown that in classical face-to-face communications our perception is different, as well as in multiperson online communications. The knowledge of such an effect does open the door for creative usage of the online channels in a very effective way.
2. Short-term and long-term effects were taken into consideration in the first step of our modeling. The specified experiments of the current study have provided the following evidence: the three factors, media socialization, individual versus mutual usage, and channel effects, are effective, to some extent also synergistically. It is thereby possible to transfer the assumptions initially mentioned in the Social Identity Deindividuation Theory [27] to a context of a realistic group development as assumed by Arrow [2] or Taylor and MacDonald [31].
3. New combinations of different channels are still possible as Facebook, Professional Online Communities, or Learning Management Systems demonstrate. Contemporaneously, particular channel properties are combined directly with the subjectively perceivable characteristics, which can also be interpreted as a creativity enhancing consequence of the channel and again leads to a modified social perception as shown by Kawalek [13]. Substantially, the results of our three experiments could support the theoretically driven expectations.

Do those findings stand for a new pattern of interpersonal activity? If we introduce the principle of 'Co-construction' [8], we may interpret the online communication as an ideal sphere for collaboration, offering possibilities for the co-construction of knowledge, the comparison of alternative viewpoints, and the explication of plans, concepts, and ideas without a too strong influence of a visible other [10]. Indeed, De Loor et al. [5] have studied this in the context of a collaboration of a human with virtual partners, i.e., so-called 'avatars.' In their experiment, they found the co-construction improved the credibility of the interactive simulation. In other words, the trust in the others' contribution to an online collaboration is even valued while interacting with virtual humans.

By that, our findings resemble the basis of interpreting interpersonal online collaboration as the creative endeavor of jointly (co-)constructing activity, which provides certain spaces for different insights. Due to the fast-growing number of ICT applications for private and business communication, the current interpretation is transferrable to many specified configurations of online communication where the visual anonymity can be found in one form or another. Furthermore, the occurring interaction of channel, socialization, and psychosocial components will shape in a distinguishing way the user's behavior in exposure to these technologies and, ultimately, also our social everyday life using modern information and communication technologies.

Acknowledgment Studies on which this research is based were supported through several grants, including the Hans-Böckler Foundation, the German National Academic Foundation, and the German Research Council.

References

1. Antoine, N.M.: Zwischen Kooperation und Wetteifer Interaktionen und mediale Organisation von Kreativität am Beispiel des koopetitiven Ideennetzwerks jovoto. Dissertation, Frankfurt am Main, Goethe-Universität (2014)
2. Arrow, H.: Stability, bistability and instability in small group influence patterns. J. Pers. Soc. Psychol. **1**, 75–85 (1997)
3. Becker, B.: Virtuelle Identitäten: Die Technik, das Subjekt und das Imaginäre. In: Becker, B., Pateau, M. (eds.) Virtualisierung des Sozialen, pp. 163–184. Campus, Frankfurt am Main (1997)
4. Curtis, P.: Mudding: social phenomena in text-based virtual realities. In: Kiesler, S. (ed.) Culture of the Internet, pp. 121–142. Lawrence Erlbaum, Mahwah (1997)
5. De Loor, P. Bernard, R., Bossard, C.: Interactive co-construction to study dynamical collaborative situations. In: Proceedings in the International Conference on Virtual Reality, pp. 85–92. VRIC LAVAL, France (2008)
6. Fenigstein, A., Scheier, M.F., Buss, A.H.: Public and private self-consciousness: assessment and theory. J. Consult. Clin. Psychol. **4**, 522–527 (1975)
7. Filipp, S.-H., Freundenberg, E.: Der Fragebogen zur Erfassung dispositioneller Selbstaufmerksamkeit. Hogrefe, Göttingen (1989)
8. Gergen, K.J.: Social construction and the educational process. In: Steffe, L.P., Gayle, J. (eds.) Constructivism in Education, pp. 17–39. Hillsdale, LEA (1995)
9. Gergen, K.J., Gergen, M., Barton, W.H.: Deviance in the dark. Psychology Today **19**, 129–130 (1973)
10. Gijlers, H.: Confrontation and co-construction. exploring and supporting collaborative scientific discovery learning with computer simulations. Doctoral thesis, Twente, University (2005)
11. Hoffmann, U.: "Request for comments": Das Internet und seine Gemeinde. In: Kubicek, H., Klumpp, D., Müller, G., Neumann, K.H., Raubold, E., Rossnagel, A. (eds.) Jahrbuch Telekommunikation und Gesellschaft, pp. 104–117. Decker, Heidelberg (1996)

12. Kahnwald, N., Köhler, T.: Microlearning in virtual communities of practice? an explorative analysis of changing information behaviour. In: Micromedia, eLearning 2.0: Getting the Big Picture. Proceedings of Microlearning 06, pp. 156–172. University Press, Innsbruck (2007)
13. Kawalek, J.: Unterricht am Bildschirm. Peter Lang, Frankfurt am Main (1997)
14. Köhler, T.: Selbst im Netz? Die Konstruktion des Selbst unter den Bedingungen computervermittelter Kommunikation. Westdeutscher Verlag, Opladen (2003)
15. Köhler, T.: Creativity in online communication. empirical considerations from a connectivist background. In: Skulimowski, A.M.J. (ed.) Looking into the Future of Creativity and Decision Support Systems. Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems, pp. 310–319. Progress & Business Publishers, Kraków (2013)
16. Lehnhardt, M.: Identität im Netz: Das Reden von der "Multiplen Persönlichkeit". In: Rost, M Die Netz-Revolution: Auf dem Weg in die Weltgesellschaft, pp. 108–124. Eichborn, Frankfurt am Main (1996)
17. Luhtanen, R., Crocker, J.: A collective self-esteem scale: self evaluation of one's social identity. Pers. Soc. Psychol. Bull. **3**, 302–318 (1992)
18. McKenna, K., Bargh, J.: Coming out in the age of Internet: identity "demarginalization" through virtual group participation. J. Pers. Soc. Psychol. **3**, 681–694 (1998)
19. Mummendey, A.: Verhalten zwischen sozialen Gruppen: Die Theorie der sozialen Identität. In: Frey, D., Irle, M. (eds.) Theorien der Sozialpsychologie. Band II: Gruppen- und Lerntheorien, pp. 185–216. Hans Huber Bern (1985)
20. Pateau, M.: Kommunikationsweise und sozialer Raum. In: Bulmahn, E., Van Haaren, K., Hensche, D., Kiper, M., Kubicek, H., Rilling, R., Schmiede, R. (eds.) Informationsgesellschaft —Medien—Demokratie, pp. 168–173. BdWi-Verlag, Marburg (1996)
21. Reid, E.: Virtual Worlds: culture and imagination. In: Jones, S.G.(ed.) Cybersociety, pp. 164–183. Sage, London (1995)
22. Pächter, M., Schweitzer, K., Weidemann, B.: Parasoziale Beziehungen zu einer Dozentin im Netz. Medienpsychologie **12**(4), 242–259 (2000)
23. Postmes, T.: Social influence in computer-mediated groups. Print Partners Iskamp, Enschede (1997)
24. Scholl, W., Pelz, J.: Computervermittelte Kommunikation in der deutschen Wissenschaft. Waxmann, Münster (1997)
25. Schumacher, A.: Eine deutsche Fassung des Rochester-Interaktionsfragebogens: Anwendbarkeit und erste Ergebnisse. Zeitschrift für Sozialpsychologie **1**, 140–149 (1992)
26. Short, J., Williams, E., Christie, B.: The social psychology of telecommunications. Wiley, Chichester (1976)
27. Spears, R., Lea, M., Lee, S.: De-individuation and group polarization in computer-mediated communication. Br. J. Soc. Psychol. **2**, 121–134 (1990)
28. Sproull, L., Kiesler, S.: Reducing social context cues: electronic mail in organizational communication. Manage. Sci. **11**, 1492–1512 (1986)
29. Sproull, L., Kiesler, S.: Connections: new ways of working in the networked organization. MIT Press, London (1991)
30. Taddicken, M.: The 'privacy paradox' in the social web: the impact of privacy concerns, individual characteristics, and the perceived social relevance on different forms of self-disclosure. J. Comput. Mediat. Commun. **19**, 248–273 (2014)
31. Taylor, J., MacDonald, J.: The impact of computer-group interaction on group decision making, communication and inter-personal perception. Bournemouth, University, unpublished paper (1997)
32. Turkle, S.: Life on the Screen. Simon and Schuster, New York (1995)
33. Turkle, S.: Constructions and reconstructions of the self in virtual reality: playing in the MUDs. In: Kiesler, S. (ed.) Culture of the Internet, pp. 143–156. Lawrence Erlbaum, Mahwah (1997)

34. Wagner, U., Zick, A.: Selbstdefinition und Intergruppenbeziehungen. Der Social identity approach. In: Pörzgen, B., Witte, E.H. (eds.) Selbstkonzept und Identität, pp. 109–130. Universitätsverlag, Braunschweig (1993)
35. Walther, J.B.: Computer-mediated communication: impersonal, interpersonal and hyperpersonal interaction. In: Paper Presented on the Annual Conference of the International Communication Association. Albuquerque (1995)
36. Walther, J.B.: Relational aspects of computer-mediated communication: experimental observations over time. Organ. Sci. **2**, 186–203 (1995)
37. Zywica, J., Danowski, J.: The faces of Facebookers: investigating social enhancement and social compensation hypotheses. predicting Facebook™ and offline popularity from sociability and self-esteem, and mapping the meanings of popularity with semantic networks. J Comput. Mediat. Commun. **14**, 1–34 (2008)

Creativity Effects of Idea-Marathon System (IMS): Torrance Tests of Creative Thinking (TTCT) Figural Tests for College Students

Takeo Higuchi, Takaya Yuizono, Kazunori Miyata, Keizo Sakurai and Takahiro Kawaji

Abstract Idea-Marathon System (IMS) is a creativity training process, based on the use of notebooks, in which we make a daily habit of writing our ideas immediately by managing to create any new idea regardless of any thinking area. This paper presents an experimental analysis conducted at Ohtsuki City College (OCC) to quantitatively measure creativity effect on college students before and after a 3-month IMS training. TTCT (Torrance Tests of Creative Thinking) Figural Pre- and Posttests were used to confirm the creativity effects on students quantitatively. The group with 3 months of IMS training showed significant increases in "Total Score," "Fluency," "Originality," and "Resistance to Premature Closure (RPC)" while students in the control group showed a significant increase in "RPC" only. Support system of IMS "e-Training System (ETS)" was found moderately correlated with "Fluency." Top, Middle, and Low analyses showed improvement in Middle and Low through 3-month IMS.

Keywords Idea-Marathon System (IMS) · Torrance Tests of Creative Thinking (TTCT) Figural tests · e-Training System (ETS)

T. Higuchi (✉) · T. Yuizono · K. Miyata
School of Knowledge Science, Japan Advanced Institute of Science and Technology, Nomi, Japan
e-mail: info@idea-marathon.net

T. Yuizono
e-mail: yuizono@jaist.ac.jp

K. Miyata
e-mail: miyata@jaist.ac.jp

K. Sakurai
Graduate School of Management, Japan University of Economics, Dazaifu, Japan
e-mail: sakurai@tk.jue.ac.jp

T. Kawaji
Department of Economics, Ohtsuki City College, Otsuki, Japan
e-mail: kawaji@ohtsuki.ac.jp

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_15

1 Introduction

The four P's model of Mel Rhodes: a creative person, a creative product, the creative process, and the creative press [14], shows that creativity means wide ranges, many directions, and many fields. In this paper, we discuss the creativity education in college students.

Colleges and universities throughout the world are in the process of revising their curricula for creativity education as today's world is becoming borderless and globalizing and creatively competitive.

Alane J. Starko made discussion about the importance of creativity in class. Starko said creative students can learn more in class [15]. The National Center on Education and the Economy (NCEE) USA, made a strong appeal for a future curriculum in higher education by the importance of creativity and innovation [12]. The European University Association (EUA) announced the keywords of their main findings and conditions for creativity, "Diversity, Value and Ethical Principles, Human Potential, Future Orientation and Quality Mechanisms." *Moreover, the EUA proposed 10 key recommendations to European higher education institutions, governments,* etc.*, including the fact that "Universities should look towards the future in all their activities, rather than being grounded in the past.…should work towards developing internal quality processes that support the creativity agenda by being geared towards the future"* [2]. Larry Livingston of the University of Southern California insists strongly in reforming university curricula for teaching creativity [11]. Bonnie Cramond also predicted that, in the future, the world will continue to become increasingly complex with problems requiring novel and elegant solutions [1].

However, it is quite difficult to propose specific, stable, effective creativity education system for students, as many kinds of creativity education, including brainstorming, cannot be repeated or practiced everyday by a student or in group. Under these circumstances, it has been proposed that the Idea-Marathon System (IMS) can be one of the innovative methodological breakthroughs for building a creative infrastructure for college and university students in Japan. IMS will be explained in the next section in detail. In the aforementioned Sect. 1 we explained the necessity of creativity education in colleges and universities. In Sect. 2, IMS and related works are discussed. In Sect. 3, IMS Creativity Training for College Students is explained. Section 4 contains the results and discussion and in Sect. 5 the conclusion and future research are given. The references and bibliography conclude the chapter.

An earlier version of this paper was presented at the International Conference KICSS'2013 [9].

2 Idea-Marathon Systems and Related Works

2.1 Basic Concept of IMS

In 1984, Idea-Marathon System (IMS) was developed and implemented by Higuchi, who published the first book on IMS in Japanese in 1992 [4]. Since that time, Higuchi has continued to practice IMS almost everyday for the last 30 years. During that period, a number of books about IMS were published in Japanese [7]. IMS books were published also in English [5], Korean, Chinese, Thai, and in Hindi.

The principles of IMS are defined as

(1) Use notebooks of the same kind.
(2) Generate new ideas daily and write them in the notebook chronologically, numbering each idea regardless of any area.
(3) Draw pictures to illustrate your ideas as often as possible.
(4) Talk to your neighbors.
(5) Review your ideas.
(6) Implement the best ideas [10]

IMS is a process which involves daily idea creation and immediate writing in notebooks individually following a chronological order. The ideas expressed are not limited to any specific area or topic but may encompass any subject area. As IMS becomes a daily habit, the practitioner will be encouraged by experiencing the power of continuity, instantaneity, and self-confidence that stem from increased creativity. A synergistic effect can also be expected to emerge among those who practice IMS in groups if they discuss the ideas in their notebooks.

Several colleges and universities in Japan including the University of Electro-Communications, Japan Advanced Institute of Science and Technology, and Ohtsuki City College (OCC) have adopted IMS, and have evaluated its effects qualitatively. To make wider use of IMS lectures in universities in Japan, IMS must be studied quantitatively for its potential to innovate the creativity of students. TTCT Figural tests were implemented for studies to prove the creative effect of IMS in detail. Prior to this study at OCC, the creativity effect of IMS for researchers at one major food manufacturer's laboratory near Tokyo was already tested and analyzed by Higuchi et al. in 2011 [8].

In this study, we made our first study from the figural test on the creative effect of IMS for college students. Another important aspect of IMS process is that there are support systems in place from the beginning to help participants develop the consistent habit of thinking and writing about their own ideas everyday, such as an e-Training System (ETS) and a distribution of weekly "Thinking e-Hints."

2.2 ETS (e-Training System) and Its Weekly e-Hints

2.2.1 ETS (e-Training System)

ETS is a support system for IMS practitioners using emails. During the IMS training period, the students send regularly the current idea number written in their notebooks and they receive individual comment with encouragement, useful references, thinking hints from IMS instructors. This is a humane monitoring system for IMS, though the actual content of students' ideas is not checked.

Even after starting IMS, if IMS students are not advised and encouraged regularly and properly, many students will naturally and gradually slow down or stop practicing IMS as their motivation decreases. During the IMS training period, students are regularly requested to report the number of ideas every week by showing their notebooks to their instructor; thus, they can keep IMS in mind at all times.

Upon receipt of students' reports of their weekly number of ideas, teachers will give each student comments according to the range and number of idea.

The teachers will check if the student keeps writing more ideas than number of days since starting. These comments will keep positively encouraging students to continue or accelerate IMS. This is named e-Training System (ETS). During 3 months of IMS training, five to six times ETS were implemented. In these experiments at OCC, we tried to study the correlation between the student's score of creativity test and number of ideas recorded in ETS.

2.2.2 Weekly e-Hints

At the beginning stage of IMS training, it is important for beginners to find some thinking hints for ideas in order to continue practicing IMS everyday.

To ameliorate this situation, students participating IMS training, can receive weekly hints from the teacher by email (e-Hints), from which they may create new ideas until they get used to creating hints themselves. Some examples of e-Hints are: (1) Create a new Sushi recipe; (2) Think of a new type of vending machine; or (3) Think of new functions and uses for a calculator.

2.3 Related Works

2.3.1 Brainstorming (BS) Method

Brainstorming was proposed by Osborn in 1939. Though BS method is quite popular and still used in companies to find out new ideas on spot and all the ideas proposed on the table in the meeting are collected and used for and by the companies anonymously [13].

But in case of IMS, all the ideas obtained individually are duly written once in each person's notebook and the proposed ideas, therefore, in the meeting have the creator's name of these original ideas prior to the meeting.

IMS is a method of self-brainstorming though IMS is initially not aiming for the ideas to cope with the given problems in the meeting, but IMS is the method of collecting ideas as much as possible throughout your work and life, then becoming better reaction at the brainstorming meetings with others.

2.3.2 Brainwriting (BW) Creativity Method

Brainwriting is a method of creating ideas in which a group of participants are requested to write ideas into a certain form without talking [3]. In IMS, ideas are individually written everyday in the same notebooks chronologically with running numbers any time of the day. It is highly recommended to talk to the neighbors for comments.

2.4 Creativity Training Procedure

Before starting IMS training at colleges, all students were requested to obtain one new notebook. To start IMS in class or group, a 1-h lecture on IMS was given so that the participating students have adequate understanding and motivation of starting and continuing IMS. The IMS lecture covered the origin of IMS, its merits, the use of notebooks, application of IMS and notebooks, and examples of ideas.

For the ETS (e-Training System) of IMS, the experimental freshmen students were requested to report their idea numbers to their instructors every week. The instructors returned their feedback to each student each time according to the change in the number of ideas over past two weeks, with advices for creative encouragement (Fig. 1).

On the ETS of the Idea-Marathon, if any students were found to be slower or to have stopped creating and writing ideas down in their notebooks, they were individually encouraged to restart, promote, and accelerate IMS practice by email, possibly supplied with more hints to think.

On April 12, 2012, experimental students started IMS with an explanation of IMS from their teacher. During 3 months, their teacher checked the notebooks of IMS controlled group every week to get their updated idea numbers. ETS was done 12 times.

The results of abbreviated reporting idea numbers after 1 week (ETS-1), after 1.5 months (ETS-8), and after 3 months (ETS-12) from IMS starting date are shown in Table 1. The comments and advices are all in Japanese folded inside Excel cell.

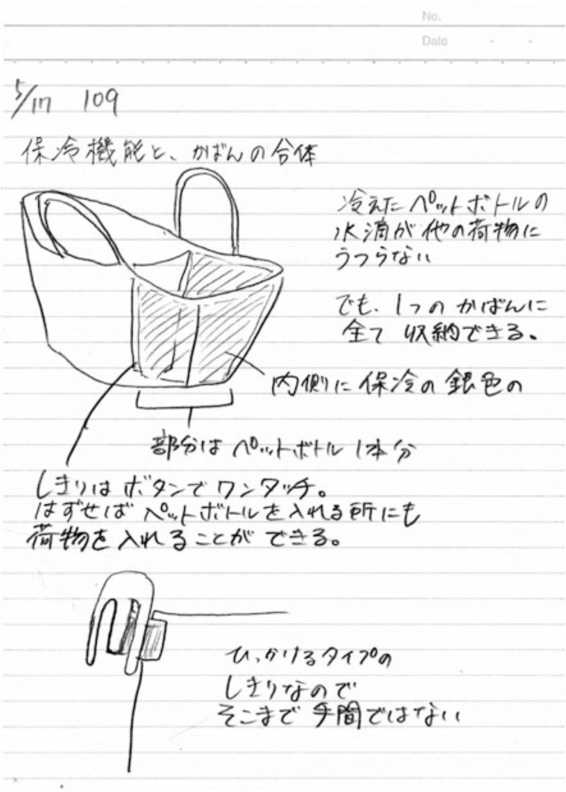


Fig. 1 Sample of IMS idea in Japanese on OCC experimental student's notebook

3 Creativity Measurement Using TTCT

Although initial verbal creativity tests were conducted at OCC, we carried out figural tests for groups of students the next year to obtain more detailed creativity factors related to IMS training. To empirically evaluate the creative effect of IMS education on OCC students, we used the Torrance Tests of Creative Thinking (TTCT) Figural tests. There are two types of TTCT Figural tests (Type A as the Pretest and Type B as the Posttest) which were used to measure the creativity effects of continuous training of Idea-Marathon for the 3-month period.

TTCT Figural tests consist of the following norm-referenced indicators [16]:

(1) Fluency: The number of expressed ideas which meaningfully utilize the stimulus. Fluency is the gatekeeper of TTCT analysis.
(2) Originality: The uniqueness of the ideas in terms of statistical infrequency.
(3) Elaborations: The imagination and exposition of detail in the pictures.
(4) Abstractness of Titles: The level of synthesis and organization evidenced in the titles in order to express the essence of the pictures.

Table 1 12 times ETS (e-Training System) data of experimental group students and comments given to students

Student ID	IMS started	1st Report day	Number of ideas	Teachers' comment	2nd Report day	Number of ideas	Teachers' comment		8th Report day	Number of ideas	Teachers' comment		12th Report day	Number of ideas	Teachers' comment
1	2012/4/12	2012/4/18	16	Your IMS Position is	2012/5/2	42	Well done		2012/6/21	95			2012/7/19	138	Hard working
2	2012/4/12	2012/4/18	6	Your IMS Position is	2012/5/2	25	Hard working		2012/6/21	82	Reviewing your idea numbers		2012/7/19	115	Quite good
3	2012/4/12	2012/4/18	1	Your IMS Position is	2012/5/2	22	Hard working		2012/6/21	90	Reviewing your idea numbers		2012/7/19	132	Reviewing your idea numbers
4	2012/4/12	2012/4/18	6	Your IMS Position is	2012/5/2	28	Try harder	From 3rd to7th Reports data	2012/6/21	93	Reviewing your idea numbers	From 9th to11th Reports data	2012/7/19	125	Quite good
5	2012/4/12	2012/4/18	10	Your IMS Position is	2012/5/2	23	Quite good		2012/6/21	86	Reviewing your idea numbers		2012/7/19	98	Hard working
6	2012/4/12	2012/4/18	7	Your IMS Position is	2012/5/2	48	Hard working		2012/6/21	181	Reviewing your idea numbers		2012/7/19	254	Try harder
7	2012/4/12	2012/4/18	7	Your IMS Position is	2012/5/2	19	Going on well		2012/6/21	170	Reviewing your idea numbers		2012/7/19	270	Hard working
8	2012/4/12	2012/4/18	16	Your IMS Position is	2012/5/2	80	Reviewing your idea numbers	Abbreviated due to Space	2012/6/21	119	Reviewing your idea numbers	Abbreviated due to Space	2012/7/19	152	Very high
9	2012/4/12	2012/4/18	7	Your IMS Position is	2012/5/2	43	Very high		2012/6/21	119	Reviewing your idea numbers		2012/7/19	140	Hard working
10	2012/4/12	2012/4/18	12	Your IMS Position is	2012/5/2	44	Hard working		2012/6/21	107	Reviewing your idea numbers		2012/7/19	139	Quite good
11	2012/4/12	2012/4/18	10	Your IMS Position is	2012/5/2	33	Try harder		2012/6/21	97	Reviewing your idea numbers		2012/7/19	111	Good!

(continued)

Table 1 (continued)

12	2012/4/12	2012/4/18	8	Your IMS Position is	2012/5/2	39	Quite good				2012/6/21	116	Reviewing your idea numbers			2012/7/19	146	Very high
13	2012/4/12	2012/4/18	12	Your IMS Position is	2012/5/2	77	Hard working				2012/6/21	224	Reviewing your idea numbers			2012/7/19	276	Very high
14	2012/4/12	2012/4/18	17	Your IMS Position is	2012/5/2	50	Hard working				2012/6/21	90	Reviewing your idea numbers			2012/7/19	118	Hard working
15	2012/4/12	2012/4/18	9	Your IMS Position is	2012/5/2	50	Hard working				2012/6/21	178	Reviewing your idea numbers			2012/7/19	277	Very high
16	2012/4/12	2012/4/18	42	Your IMS Position is	2012/5/2	104	Going on well				2012/6/21	200	Reviewing your idea numbers			2012/7/19	248	Very high
17	2012/4/12	2012/4/18	8	Your IMS Position is	2012/5/2	64	Going on well				2012/6/21	96	Reviewing your idea numbers			2012/7/19	127	Hard working
18	2012/4/12	2012/4/18	9	Your IMS Position is	2012/5/2	74	Hard working				2012/6/21	169	Reviewing your idea numbers			2012/7/19	229	Hard working
19	2012/4/12	2012/4/18	7	Your IMS Position is	2012/5/2	33	Hard working				2012/6/21	90	Reviewing your idea numbers			2012/7/19	124	Quite good
20	2012/4/12	2012/4/18	16	Your IMS Position is	2012/5/2	43	Quite good				2012/6/21	87	Reviewing your idea numbers			2012/7/19	117	Reviewing your idea numbers
21	2012/4/12	2012/4/18	3	Your IMS Position is	2012/5/2	25	Going on well				2012/6/21	74	Reviewing your idea numbers			2012/7/19	102	Try harder
22	2012/4/12	2012/4/18	7	Your IMS Position is	2012/5/2	57	Hard working				2012/6/21	116	Reviewing your idea numbers			2012/7/19	165	Hard working
			236			1,023						2,679					3,603	
			1.60			2.32						1.79					1.62	

(5) Resistance to Premature Closure (RPC): The ability to consider all relevant information and resist impulsive, premature conclusions.

In addition to the above-mentioned five norm-referenced indicators, there are 13 more Creative Strengths as Criterion-referenced measures in TTCT analysis to check existence or nonexistence of: Emotional Expressiveness, Storytelling Articulateness, Movement and Action, Expressiveness of Titles, Synthesis of Incomplete Figures, Synthesis of Lines and Circles, Unusual Visualization, Internal Visualization, Extending or Breaking Boundaries, Humor, Richness of Imagery, Colorfulness of Imagery, and Fantasy.

To make the inter-rater scoring, the first author obtained the official certificate for scoring TTCT tests Figural Type A test and Type B test at a scoring seminar held by the TTCT Center of the University of Georgia, from October 1 to October 3, 2012.

3.1 TTCT Figural Pretest Type A and Posttest Type B in OCC

Two TTCT Figural tests of the Pretest (Type A test) on April 19, 2012, and Posttest (Type B test) on July 26, 2012 were administered to two groups of OCC students at once. One group of freshmen IMS Experimental Group students ($N = 21$) had completed 3 months of IMS training and another group of sophomore students were Non-IMS Control Group Students ($N = 19$). Both groups had TTCT Figural Pretest and Posttest on the same day.

3.2 Effects of IMS on TTCT Scores

For the experimental group, a statistically significant difference was found on the norm-referenced indicators: Total Score $t(20) = -4.399$ $p < 0.01$**, on Fluency $t(20) = -5.468$ $p < 0.01$**, Originality $t(20) = -3.261$ $p < 0.01$**, and Resistance to Premature Closure $t(20) = -2.878$ $p < 0.01$**. No significant difference was found for the component of Elaborations and Abstractness of Titles (Table 2). Figures in Table 2 depict average scores.

Table 2 TTCT scores of IMS experimental group students with 3-month IMS at OCC 2012

Total Score		Fluency		Originality		Elaborations		Abstractness of titles		Resistance to Premature Closure	
Pre	Post	Pre	Post	Pre	Post	Pre	Post	Pre	Post	Pre	Post
518	590	86	111	100	120	141	150	109	110	82	99
	**		**		**						**

Significance *<5 %, **<1 %

Table 3 TTCT scores of Non-IMS control group students OCC 2012

Total score		Fluency		Originality		Elaborations		Abstractness of titles		Resistance to Premature Closure	
Pre	Post	Pre	Post	Pre	Post	Pre	Post	Pre	Post	Pre	Post
571	578	103	106	118	112	153	146	106	104	92	110
											**

Significance *<5 %, **<1 %

On the TTCT Figural tests for control group students, a statistically significant difference was found only for Resistance to Premature Closure $t(18) = -3.412$ $p < 0.01$** in five norm-referenced indicators, and no significant difference was found for the measures of Total Score, Fluency, Originality, Elaborations, and Abstractness of Titles (Table 3). The figures inside Table 3 are all average scores.

This TTCT experiment was performed at OCC between freshmen (experimental Group) and sophomore students (control group). Between these two grades, the creativity tests at Pretest and Posttest were studied.

The Pretest between freshmen students (experimental Group) and sophomore students (control Group) showed significance for Total Score $t(38) = -2.231$ $p < 0.05$*, Fluency $t(38) = -2.981$ $p < 0.01$**, Originality $t(38) = -2.488$ $p < 0.05$*, and Elaborations $t(38) = -2.217$ $p < 0.05$* (Table 4 Pretest).

From the results of the Pretest, we conclude that the sophomore students showed higher creativity scores than the freshmen students. The bold underlined figures in Table 4 are higher average scores.

However, as a result of 3 months of IMS practice for the freshmen experimental students, at Posttest, all the above-mentioned significant items were canceled out due to the score improvement of freshmen students, while the sophomore students scores remained almost the same or decreased (Table 4 Posttest).

3.3 *Influence of ETS on Creativity*

The increase of total idea numbers during these 3 months are shown in Fig. 2.

The average daily idea numbers per person are shown in Fig. 3, in which the students produced 1.6–1.7 ideas per day per person in average. This is higher than that of the students of University of Electro-Communications (UEC) 2007–2010, which was 1.3 ideas per day [6].

The correlation of the idea numbers recorded in ETS was tested to each of six TTCT norm-referenced indicators, including Total Score, Fluency, Originality, Elaborations, Abstractness of Titles, and Resistance to Premature Closure. It was found that ETS data was moderately correlated in Fluency, with only $r = 0.419$, while all other norms were not correlated.

Table 4 Comparison and test of TTCT norm-referenced indicators' scores between IMS experimental group students and Non-IMS control group students

	Total Score			Fluency			Originarity			Elaborations			Abstractness of titles			RPC		
	Pre	Post	%	Pre	Post	%	Pre	Post	%	Pre	Post	%	Pre	Post	%	Pre	Post	%
IMS experimental group freshmen	518	590	114	86	111	129	100	120	120	141	150	106	109	110	101	82	99	121
Non-IMS control group sophomore	571	578	101	103	106	103	118	112	95	153	146	95	106	104	98	92	110	120
Significance	*			*			**			*								

Significance *<5 %, **<1 %

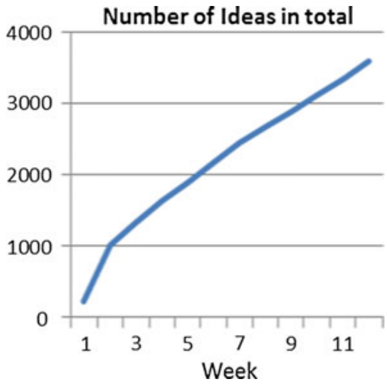


Fig. 2 Accumulated total numbers of ideas shown in ETS in OCC, 2012

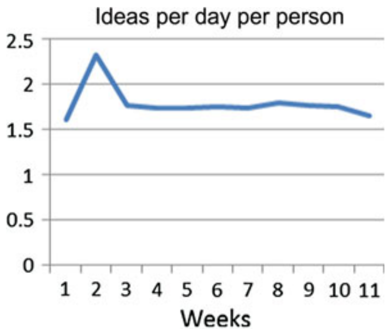


Fig. 3 Average numbers of ideas per day per student in OCC, 2012

3.4 Results of Analysis of Top, Middle, and Low of TTCT Figural Pretest Compared with Figural Posttest

The mutual relation among Top, Middle, and Low scores in Pretest and Posttest was analyzed. The Top, Middle, and Low scores of the Pretest were divided according to the table of National Percentile in TTCT Norms-Technical Manual Score Table [17]:

(1) Top (more than 90 % of National Percentile in TTCT Score Table) 7 persons
(2) Middle (90 % and less, and more than 50 % of National Percentile of TTCT Score Table) 8 persons
(3) Low Score (50 % and less of National Percentile of TTCT Score Table) 6 persons

Top/Middle/Low scores were tested first by Nonparametric Kruskal–Wallis test and then by multiple comparison test (Nonparametric), Steel-Dwass (Table 5).

In Total Score, Middle got closer to Top, Low closer to Middle. In Originality, Low got closer to Top. In Elaborations, Low got closer to Top. In Abstractness of Titles, Middle and Low got closer to Top. In RPC, Low got closer to Top and Middle.

Table 5 Significant position among Top, Middle, and Low of Pretest and Posttest

		Total Score			Fluency			Originality						Abstractness of titles			RPC		
		TM	ITL	IML	TM	ITL	IML	TM	ITL	IML	TM	ITL	IML	TM	ITL	IML	TM	ITL	IML
Pretest A-type	Kruskal–Wallis	**			Nil			*			**			**			**		
	Steel-Dwass test	*	**	*				Nil	*	Nil	Nil	**	*	*	**	Nil	Nil	*	*
Posttest	Kruskal–Wallis test	Nil			Nil			Nil			Nil			Nil			Nil		
B-type	Steel-Dwass test																		

p value *<0.05, **<0.01

4 Discussion

The discussion from the above-mentioned results is as follows:

(1) Judging from the result of ETS (Table 1) of OCC students for a 3-month period, 3,803 ideas in total were recorded in their notebooks, which is 1.6 ideas per day per student written in their notebooks, it can be understood that the OCC students practiced IMS basically according to the rules (defined in 2.1 Basic Rules of Idea-Marathon). The steadiness of keeping IMS rule of daily thinking and writing into notebooks is the base for self-improvement of creativity and can become important factor for maintaining longer motivation. If students continue IMS longer and more and more ideas are kept stocked in their notebooks, they get more self-satisfaction from the accumulated idea stock.

(2) Between the ETS idea numbers and TTCT Fluency score, there was a moderate correlation that if students write as many ideas as possible in their notebooks, their score of TTCT Fluency will probably improve. Paul E. Torrance said *"Fluency is the gatekeeper of creativity"* [17].

(3) IMS Experimental Group for 3 months supported by ETS and weekly supply of Thinking Hints has shown significance for the Total Score, Fluency, Originality, and RPC. All the TTCT scoring process starts from scoring Fluency. Therefore, increase of Fluency is directly related to Originality and other norm-referenced indicators of TTCT Figural tests. Or without Fluency score, other norm-referenced indicators will not be counted [18]. In this experimental group of students, Elaborations and Abstractness of Titles were not found significant and not activated during 3 months.
As the Non-IMS control group of students over 3-month interval did not show any statistical significance on Fluency, Originality, Elaborations, Abstractness of Titles, Total Scores except Resistance to Premature Closure (RPC), the creativity level of Non-IMS control group remained same as before.

(4) From the result of analysis of Top, Middle, and Low, all norm-referenced Indicators showed the improvement of Middle and/or Low to Top, which prove that 3 months Idea-Marathon training is effective to Middle and Low scorers.

5 Conclusions

TTCT Figural tests were proceeded for the first time in Ohtsuki City College in 2011 for both IMS experimental group students and Non-IMS control group students with a 3-month interval. The result of TTCT Figural test showed significance in Total Score, Fluency, Originality, and RPC for IMS experimental group students. The e-Training System was found to have correlation with Fluency in this experiment. The Middle and Low scores were more improved than Top scorers in all indicators except Fluency.

In future experiment, we plan to check whether the creativity scores can be improved more if the students continue to practice IMS even after 3 months period with or without ETS and Thinking Hints. Experimental period of 6 months and 9 months for IMS with TTCT Figural tests can be tried.

Acknowledgment We thank Professor Bill Holden of Japan Advanced Institute of Science and Technology (JAIST) for his full support of detailed proofreading to this paper in such a short period during his extreme tight schedule.

References

1. Cramond, B.: Creativity in the Future, Encyclopedia of Creativity, vol. 1, pp. 423. Academic Press, San Diego (1999)
2. EUA Executive Summary of Creativity in Higher Education, Report on the Creativity Project 2006–2007. European University Association, Belgium (2007)
3. Heslin, P.: Better than brainstorming? Potential contextual boundary conditions to brainwriting for idea. J. Occup. Organ. Psychol. The British Psychological Society (2009)
4. Higuchi, T.: You can get affluent ideas from this book (in Japanese). Diamond Inc, Tokyo (1992)
5. Higuchi, T.: Ideas in Action, Digital Achievement of Idea Marathon System. Adarsh Books (2001)
6. Higuchi, T.: Application of Idea-marathon for career design class at the university of electro-communications. In: 32nd Conference of Japan Creativity Society, Tokyo (2010)
7. Higuchi, T.: Notebook Technology for Business (in Japanese). Toyo-Keizai Shinposha, Tokyo (2011)
8. Higuchi, T. et al.: Creativity Improvement by Idea-Marathon Training, Measured by Torrance Tests of Creative Thinking (TTCT) and Its Applications to Laboratories, KICSS2012. Melbourne Australia (2012)
9. Higuchi, T.,Yuizono T., Miyata K., Sakurai K. Kawaji T.: Effects of Idea-Marathon System (IMS): Torrance tests of creative thinking (TTCT), figural tests for college students In: Skulimowski, A.M.J. (ed.), Looking into the Future of Creativity and Decision Support Systems: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems, Kraków, Poland, Nov 7–9, 2013, Advances in Decision Sciences and Future Studies, vol. 2, pp. 279–290. Progress & Business Publishers, Kraków (2013)
10. Higuchi, T.: Idea-Marathon System, Encyclopedia of Creativity, Invention, Innovation, and Entrepreneurs. Springer, UK (2013)
11. Livingston, L.: Teaching creativity in higher education. Arts Educ. Policy Rev. **111**, 59–62 (2010)
12. NCEE: Executive Summary, National Center on Education and the Economy, p. 8. Washington DC (2007)
13. Osborn, A.F.: Applied Imagination. Charles Scribners Sons (1957)
14. Rhodes, M.: An analysis of creativity. The Phi Delta Kappan, USA **42**(7), 305–310 (1961)
15. Starko, A.J.: Creativity in the Classroom, Schools of Curious Delight. Taylor & Francis, New York (2012)

16. Torrance, E.P.: Torrance Tests of Creative Thinking: Directions manual and scoring guide, Verbal test booklet A. Bensenville, Il. Scholastic Testing Service (1974)
17. Torrance, E. P.: Torrance Tests of Creative Thinking Norms-Technical Manual Figural (streamlined) Forms A & B, Scholastic Testing Service, Inc. (1998, 2008)
18. UGA Georgia Center: TTCT Figural Workshop Manual, Mar 2012

A Color Extraction Method from Text for Use in Creating a Book Cover Image that Reflects Reader Impressions

Takuya Iida, Tomoko Kajiyama, Noritomo Ouchi and Isao Echizen

Abstract The image on a book cover gives potential buyers not only an impression of the book's contents but also a clue for search and browsing before or after buying the book. We propose using a color extraction method as the first step in automatically creating book cover images that reflect readers' impressions. We constructed a database expressing the relationships between adjectives and colors and extracted colors from text such as sentences in the book and user reviews. In an experiment with 20 participants who were tasked with reading a book, writing a review of the book, and drawing an image of the book cover, we demonstrated that the colors extracted using this method were more consistent with the colors in the images drawn by the participants than the colors in the actual cover, especially for novels, regardless of the amount of text in the book.

Keywords Color extraction · Adjective · Book cover image · User review · Color psychology

1 Introduction

The use of electronic devices for reading digital books is quickly spreading, and the market for digital books is growing rapidly [7]. There are two basic types of digital books: printed books that have been digitalized and books prepared only in digital form. The use of electronic devices for reading has changed not only how people

T. Iida · T. Kajiyama (✉) · N. Ouchi
Aoyama Gakuin University, 5-10-1 Fuchinobe, Chuo-ku, Sagamihara-Shi, Kanagawa 252-5258, Japan
e-mail: tomo@ise.aoyama.ac.jp

I. Echizen
National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-Ku, Tokyo 101-8430, Japan

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_16

read books but also how they select books. Moreover, some digital books come without a cover, unlike printed books.

A book cover serves an important function—it gives potential buyers an impression of the book. Therefore, digital books sold online [2, 3] that have been in the public domain or are original digital books and are provided without book cover images. They are often given images containing only the book title with a standard design. The colors in the image are defined on the basis of the genre or are simply random. As a result, the cover image is of little use in searching for a book and browsing the book contents both before purchasing a book and when it is on the purchaser's virtual bookshelf. Several services and applications have been developed for browsing a book's contents. A service has been developed for designing an image for the cover of a digital book [10], and an application has been developed that enables users to design such an image by themselves [8]. However, using this service or application is costly or time consuming.

To support a user's search for a digital book, it is important to create book cover images automatically. In general, the higher the person's expectations before reading, the lower the level of satisfaction after reading, and the lower the expectations before reading, the greater the chances of opportunity loss [5]. It is thus important to reduce the gap between the impression obtained from a book cover image and the impression after reading the book. To realize a function that can do this, we have to consider not only the contents of the book but also the impressions of its readers. Reader impressions can be found in the user reviews commonly found on sites selling digital books. Users can read the reviews and use them to make a purchase decision. However, a user's emotions and latent comments about a book are most likely to be represented in images, not text [1]. In addition, the meaning of something can generally be understood more quickly from an image than from characters [15].

We are developing a method for automatically creating book cover images that reflect reader impressions [6]. As the first step, we focused on defining colors because colors in images are generally considered to be the most significant aspect in studies on color psychology [13] because colors create various kinds of associations and affect wits such as imagination and fantasy. To develop a method for extracting color from texts, we focused on the relationships between colors and adjectives as adjectives typically represent the inner emotional state. We constructed a database showing the correspondence between adjectives and colors and created a method for extracting color from texts such as the text in a digital book and the text in user reviews. To evaluate this method, we performed a usability test in which 20 participants were tasked with reading a book, writing a report about it, and drawing an image of the book's cover.

2 Color Extraction Method

2.1 Overview

Colors reflecting the reader impressions of a digital book are extracted using the text in the book and reader reviews of the book. To visualize the emotions described in the book and the feelings of the reader, it is necessary to extract the words describing them. There are two types of emotion: emotion that can be observed externally by others and described objectively, such as “surprised” and “fired up,” and emotion that represents an inner state based on emotion, such as “terrible (story)” and “moving (event).” In general, the former type is described using verbs, and the latter type is described using adjectives [12].

In the method we developed, colors are extracted using adjectives because our objective is to reflect reader impressions, i.e., to reflect the inner emotions of readers. Figure 1 shows the flow of this method. Morphological analysis is performed on the input text (sentences in the book and reader reviews) to extract the adjectives. Scores are then calculated for the adjectives. A score is calculated for each adjective on the basis of the number of occurrences of adjectives and other words. A score is calculated for each color using a score for each adjective and a color database expressing the relationship between adjectives and colors. Colors with higher scores are then extracted for use in the cover image.

2.2 Color Database

We constructed the color database using a color image scale [9] expressing the relationships between adjectives and colors. This scale defines 180 adjectives representing basic emotions for 130 colors that capture experiences psychologically. This database has three attributes: color (RGB), adjective, and frequency of use. The frequency of use is defined on a five-star scale—the greater the number of stars, the

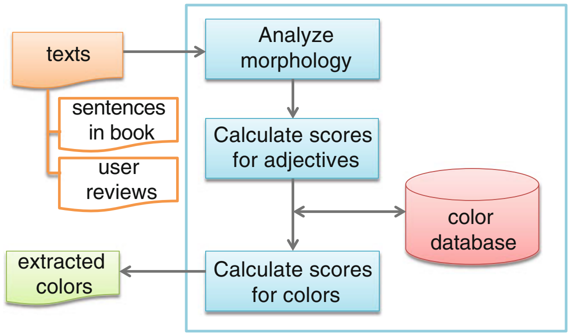


Fig. 1 Flow of proposed method

more strongly the color represents the image of the adjective [9]. Each color corresponds to more than 1 but less than 25 adjectives. The 180 adjectives were extended to 3184 using a thesaurus [16].

2.3 Scores for Adjectives

The morphological analysis extracts the adjectives from the input text, and a score is calculated for each one. The total number of occurrences of adjectives in the book $(a_{11}, a_{12}, a_{13}, \ldots, a_{1n})$ is defined as $(x_{11}, x_{12}, x_{13}, \ldots, x_{1n})$, and the total number of occurrences of adjectives in the reader reviews $(a_{21}, a_{22}, a_{23}, \ldots, a_{2m})$ is defined as $(x_{21}, x_{22}, x_{23}, \ldots, x_{2m})$. The total number of words in the book is n_1, and the total number of words in the reader reviews is n_2. Weight w_{1i} for an adjective in book a_{1i} is calculated using $w_{1i} = a_{1i}/n_1$, and weight w_{2i} for an adjective in user review a_{2i} is calculated using $w_{2i} = a_{2i}/n_2$.

The set of adjectives $(b_1, b_2, b_3, \ldots, b_\mathrm{r})$ is created by eliminating duplication between $(a_{11}, a_{12}, a_{13}, \ldots, a_{1n})$ and $(a_{21}, a_{22}, a_{23}, \ldots, a_{2m})$. The score $(y_1, y_2, y_3, \ldots, y_r)$ for $(b_1, b_2, b_3, \ldots, b_\mathrm{r})$ is calculated using

$$\mathrm{y}_i = \begin{cases} w_{1j} + w_{2k} & (\mathit{if}\ \mathrm{b}_i = a_{1j} = a_{2k}) \\ w_{1j} & (\mathit{if}\ \mathrm{b}_i = a_{1j}) \\ w_{2k} & (\mathit{if}\ \mathrm{b}_i = a_{2k}) \end{cases}$$

2.4 Scores for Colors

The colors in the color database correspond to various numbers of adjectives, so we normalized the frequency of use because the sums of the frequency of use are different. If a given color C_j has defined adjectives $(d_{j1}, d_{j2}, d_{j3}, \ldots, d_{jp})$ and frequency of use $(t_{j1}, t_{j2}, t_{j3}, \ldots, t_{jp})$, the weights for the adjectives $(z_{j1}, z_{j2}, z_{j3}, \ldots, z_{jp})$ are calculated using

$$z_{jk} = t_{jk} / \textstyle\sum_{m=1}^{p} t_{jm}.$$

The score for C_j $\mathrm{S_j}$ is calculated using the adjective scores and the weight for each adjective:

$$S_j = \sum y_i \times z_{jk} \quad \left(\mathit{if}\ b_i = d_{jk}\right).$$

The calculated S_j are arranged in ascending order, and the colors with higher scores are extracted on the basis of a threshold.

3 Evaluation

3.1 Overview

An experiment was performed to determine how well the extracted colors reflect the reader impressions and to determine the effects of the book features. The participants were 20 university students, and five digital books were used. Three books (A–C) were originally printed books and had covers; the two other books (D and E) did not have covers. Table 1 shows the features of each book.

The participants were tasked with reading one of the books, writing a report about it, and drawing an image of the book's cover after reading it. The drawing task was open-ended in regards to PC or paper, size, and color. All of the books were presented to the participants without a cover to prevent them from drawing an image with the original cover in mind. We defined the images drawn by the participants as p-cover images.

3.2 Method

To determine the degree of coincidence between the colors extracted using our method and the colors extracted from the p-cover images, we defined the threshold mentioned in Sect. 2.4 as an upper value of 15 %. All books with a cover were consisted of three colors. The ratio of three high score colors for each book was variable, so that we defined the average ratio for the top three colors as a threshold. The colors extracted from the p-cover images were defined using the following procedure:

(1) Set to zero the number of occurrences for all colors in the color database.
(2) Calculate RGB value for a given pixel in each p-cover image.
(3) Select the color in the color database that is the closest to the value calculated in step (2).
(4) Add 1 to the number of occurrences for the selected color in step (3).
(5) Repeat steps (2) to (5) until all pixels have been scanned.
(6) Sort color scores in ascending order on the basis of the number of occurrences.

Table 1 Features of each book

Book	Cover	Genre	No. of letters	No. of words
A	Y	Novel	123,728	25,349
B	Y	Novel	87,569	17,294
C	Y	Novel	91,773	17,605
D	N	Novel	6,648	1,260
E	N	Review	4,461	940

(7) Calculate the percentage for each color by dividing the number of its occurrence by the total number of pixels.
(8) Calculate the number of colors for the top 15 % of the extracted colors.
(9) Extract the colors having the same number of colors in step (8).

To determine the degree of coincidence between the colors extracted using our method and the colors extracted from the p-cover images, we integrated each color on the basis of hue. We then defined 13 colors (ten chromatic colors and three monotone colors). For the p-cover images drawn with color pencils that had insufficient region fill, we performed region filling.

3.3 Results

3.3.1 Reflection of Reader Impressions

To determine how well our method reflects reader impressions, we made a comparison for two cases: using only the sentences in the digital books as input text (case 1) and using the sentences in both the digital books and reviews as input text (case 2). In case 1, we used the number of occurrences for the extracted adjectives as the score for each adjective as described in Sect. 2.3.

Figure 2 shows the degree of coincidence between the colors extracted with our method and the colors extracted from the p-cover images. As shown by the average for the five books, the coincidence was 8.8 % higher for case 2. The results show that our method reflects the reader impressions by adding reviews.

Table 2 shows the colors extracted for book A, which had the biggest difference between cases 1 and 2, from three participants. Table 3 shows adjectives with high score for the adjective for three participants in Table 2. The colors extracted from the p-cover images of participants (a) and (b) were mainly yellowish and reddish, which coincided with the colors extracted with our method in case 2. The common adjectives extracted from the participants were 'pretty' and 'active.' All adjectives were positive impression for reading and they seemed to enjoy reading.

Fig. 2 Degree of coincidence between extracted colors

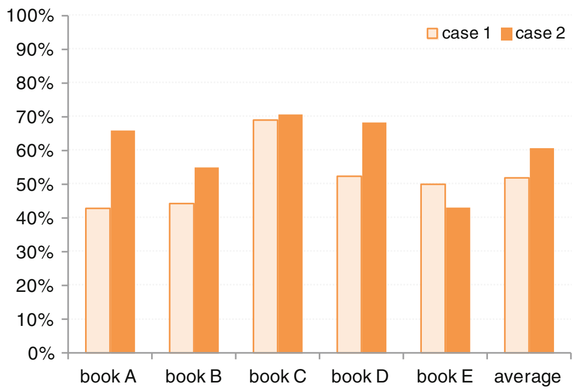


Table 2 Colors extracted for book A

p-cover images			Our method		Actual cover
images		colors	case 2	case 1	
a					
b					
c					

Table 3 Extracted adjectives with high score by our method for book A

Case 1		Quiet, beautiful, lonely, archaic, unlovely, heavy
Case 2	a	Enjoyable, dynamic, pretty, tough, active
	b	Pretty, active, attractive, flaming, bustling
	c	Quiet, archaic, beautiful, awful, old

In contrast, the colors extracted from the p-cover image of participant (c) were mainly yellowish and black. These colors matched the colors of the actual cover and did not coincide with the colors extracted with our method in case 2. Since the results for participant (c) were the same for both cases, there were apparently some adjectives in the review of participant (c) that were not registered in the color database such as 'chummy' and 'celebrity-like.' As the results, many adjectives with high scores in case 2 were the same as adjectives extracted in case 1.

3.3.2 Effect of Book Features

Figure 3 shows the results of the book features experiment. The "long" corresponds to the average number of letters for books A, B, and C, and the "short" corresponds to the number for book D. The "novel" corresponds to book D, and the "review" corresponds to the results for book E. The average number of words for the long

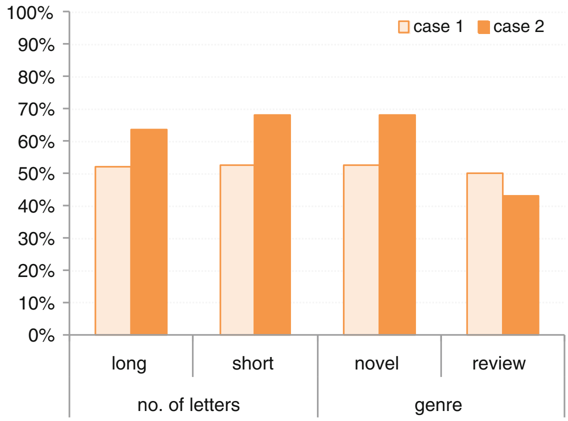


Fig. 3 Effect of book features

books was 20,083, and they contained 972 adjectives on average. There were 1,260 words in the short book, and it contained 62 adjectives. As the number of words increased, various kinds of readers' impressions occurred and different kinds of p-cover images are drawn by participants. Our method thus provides almost the same degree of coincidence between long books and short books. The results show that our method reflects the reader impressions regardless of the number of words in the book.

The average number of words and adjectives for genre was about the same as for the short book. The review contained 1,260 words and 26 adjectives. The degree of coincidence was higher for the novel in case 2 while it was higher for the review in case 1. Table 4 shows the colors extracted for book B from three participants, and

Table 4 Colors extracted for book E

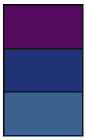
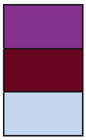
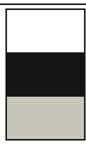
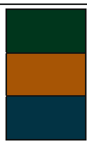
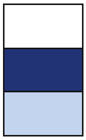
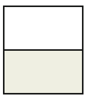
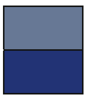
p-cover image			Our method	
images		colors	case 2	case 1
a				
b				
c				

Table 5 Extracted adjectives with high score by our method for book B

Case 1		Complex, innovative, doubtful, serious, difficult
Case 2	a	Intelligent, smart, mechanical, discreet, arid
	b	Dry, mechanical, antique, old, acerbic
	c	Artificial doubtful, serious, difficult, rustic

Table 5 shows adjectives with high score for the adjective for three participants in Table 4. The reviews for genre review were mostly expositive, and there were few adjectives representing the impressions of the participants. Therefore, the degree of coincidence in case 2 is lower than case 1. However, all the extracted colors were dark tone or monotone as shown in Table 4, which are colors extracted for book E whose genre is review. The degree of coincidence would likely be higher if the threshold value was changed or more colors were used to calculate the coincidence.

3.4 Discussion

3.4.1 Readers Impression

For book A, the colors extracted from actual cover were defined 'wild' and 'tough' as the attribute values in the color database described in Sect. 2.2. This novel was the story of school life in girl's high school and described girls who confront problems and live strongly. The colors extracted from an actual cover symbolize the story; however, the most impressed thing for this story for the participants (a) and (b) in Table 2 was cheerful and energetic girls. Although this method was proposed to reflect reader impression, the intention of author has also to be respected. To solve the situation when the intention of author was different from the reader impression, one solution is that colors extracted by our method are displayed instead of the reviews by co-exist with an actual cover image.

The color database constructed by our method was expressing relationships between colors and emotions for Japanese adults. It was not considered the difference of impressions for colors for age and nationality. Our method could apply more people by replacing other database adapting the target users.

3.4.2 Possible Application

There are various kinds of methods to support browsing books in virtual world, i.e., a graphical search interface replicating bookshelf in real world by capturing the spine of books [14] and automatically capturing a book cover in library counter [11]. These approaches are focused on recreating environment for browsing paper books on virtual world. By creating book cover images applying our method, it is considered that effective browsing and searching books can be provided because users could catch the impressions after reading intuitively.

Possible application of our method includes a real-time book recommendation system. Book recommendation system in e-commerce sites is provided based on texts such as information of the books purchased or browsed by users and reviews. It is difficult to assess users' present emotions and preferences because they continue shifting with time, and therefore, wrong recommendation sometimes happened [4]. By displaying a cover image applying our method during book search, it is considered that users can assess the impressions after reading the book and verify whether they fit with users' present feelings. It is considered that analysis of browsing history in real time provides effective recommendation reflecting user impressions.

4 Conclusions

We have developed a method for automatically extracting colors from text for use in creating a book cover image that reflects the reader impressions. We constructed a color database showing the correspondence between adjectives and colors and developed a method for extracting colors from texts such as digital books and user reviews. We evaluated this method experimentally by tasking 20 participants to read a book, write a report about it, and draw an image of the book's cover. We found that (1) the degree of coincidence between colors extracted from our method and colors extracted from p-cover images was higher when the text was extracted from both books and reviews than when it was extracted from only the books, (2) there is sometimes a difference between the actual cover and the readers' impressions after reading it, (3) our method can extract colors related to the reader impressions regardless of the number of letters in the book, and (4) our method works better for novels than for reviews. We have demonstrated that our method can extract colors for use in creating book cover images that reflect the reader impressions.

To create effective cover image reflecting reader impressions, it is necessary not only to improve the accuracy of the color extraction by enhancing the word analysis but also to optimize the arrangement of the extracted colors, to extract symbolic objects from body text, and to represent bibliographic information (title, authors, etc.).

Acknowledgments This research was supported in part by a Japan Society for the Promotion of Science Grant-in-Aid for Young Scientists (B) (No. 20454085) and the Telecommunications Advancement Foundation. We greatly appreciate the help of the participants in the experiment.

References

1. Akamatsu, N.: The effects of first language orthographic feature on second language reading in text. Lang. Learn. **53**(2), 207–231 (2003)
2. Aozora Bunko. http://www.aozora.gr.jp/ (2014). Accessed 27 Sept 2014
3. Calibre. http://calibre-ebook.com/ (2014). Accessed 27 Sept 2014
4. Choice Stream Survey Reports. http://www.choicestream.com/who/pressrelease.php?id=95 (2014). Accessed 27 Sept 2014
5. Dawes, R.M., Singer, D., Lemons, F.: An experimental analysis of the contrast effect and its implications for intergroup communication and the indirect assessment of attitude. J. Pers. Soc. Psychol. **21**(3), 281–295 (1972)
6. Iida, T., Kajiyama, T., Ouchi, N., Echizen, I.: A color extraction method from text for use in creating a book cover image that reflects reader impressions. In: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems (KICSS2013), pp. 223–232 (2013)
7. ICT Research & Consulting. http://www.ictr.co.jp/topics_20110714.html (2014). Accessed 27 Sept 2014
8. Interwred Timely Research. http://www.dims.ne.jp/timelyresearch/2009/090202/ (2014). Accessed 27 Sept 2014
9. Kobayashi, S.: Color Image Scale (in Japanese). Kodansha (2001)
10. Libura. http://libura.com/ (2014). Accessed 27 Sept 2014
11. Miyagawa, T., Oomori, Y., Ikeda, D.: Automation of back cover image generation in virtual bookshelf. Digital Libr. **30**, 25–38 (2006)
12. Ohso, M.: Verbs and adjectives of emotion in Japanese. Stud. Lang. Cult. **22**(2), 21–30 (2001)
13. Oyama, T.: Introduction of Color Psychology (in Japanese). Chuokoron-shinsha (1994)
14. Shinsho-map. http://bookshelf.shinshomap.info/ (2014). Accessed 27 Sept 2014
15. Simazaki, K.: Psychological Color (in Japanese). Bunka-shobo-hakubunnsha (1990)
16. Weblio Thesaurus. http://thesaurus.weblio.jp/ (2014). Accessed 27 Sept 2014

On the Role of Computers in Creativity-Support Systems

Bipin Indurkhya

Abstract We report here on our experiences with designing computer-based creativity support systems over several years. In particular, we present the design of three different systems incorporating different mechanisms of creativity. One of them uses an idea proposed by Rodari to stimulate imagination of the children in writing a picture-based story. The second one is aimed to model creativity in legal reasoning, and the third one uses low-level perceptual similarities to stimulate creation of novel conceptual associations in unrelated pictures. We also introduce a fourth study, which explores the role of surface-level similarities in stimulating creative ideas. We discuss lessons learnt from these approaches, and address their implications for the question of how far creativity can be tamed by algorithmic approaches.

Keywords Algorithmic creativity · Creativity support systems · Conceptual similarities · Legal reasoning · Perceptual similarities · Stimulating creativity

1 Introduction

Even though the last few decades have seen a steady progress in the development of computer systems that produce artifacts in the domain of visual art [8, 51], music [7, 48, 52], literature [46, 58]; and so on, generally they have received a negative press as regard to their creativity: computers cannot have emotions, programs do not have intents, creativity cannot be algorithmic, etc. [4, 65]. Even designers of computational creativity systems seem to take an apologetic tone when it comes to ascribing creativity to their systems. For example, Colton [9] argues that it is not enough to generate an interesting or creative artifact, but one must also take into

B. Indurkhya (✉)
Department of Computer Science, AGH University of Science and Technology, Cracow, Poland
e-mail: bipin@agh.edu.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_17

account the process by which the artifact was generated. Krzeczkowska et al. [45] took pains to project some notion of purpose in their painting tool so that it might be perceived as creative. Such views blatantly expose the implicit assumptions underlying creativity: namely that it crucially needs a creator with emotions, intentions, and such. A consequence of this view is that creativity is considered an essentially human trait, and cannot be ascribed to computer programs or AI systems (or to animals like elephants and gorillas).

We critically examine this traditional view in the light of our previous experiences in designing creativity support systems and modeling creativity. We present four such case studies here, each of which incorporates a different mechanism of creativity. These systems are based on our previous research, and we will mention here only the main ideas behind each of the systems and the results. After a brief discussion of these systems, we will present our views on the role of computers in supporting and modeling creativity. These ideas were presented at a conference KICSS'2013 held in Kraków, Poland [38]. This chapter is a revised and extended version of that conference paper.

2 Some Case Studies of Computer-Based Creativity Support Systems

We have been studying and modeling different aspects of creativity for over 20 years [30–42]. During this time, we have also explored various computational approaches to creativity, and have developed some computational systems that stimulate imagination and emergence of novel ideas and associations in the users, or model such processes. We present here three such systems in order to provide some concrete examples of how computers can play a crucial role in supporting creativity.

2.1 Stimulating Creativity in Generating Stories

We implemented a system *Story Telling from Pictures* [42] inspired by an idea *Little Red Riding Hood in a helicopter* from Rodari's fascinating book *The Grammar of Fantasy* [62]. In this technique, children are given a list of five or six words, and are asked to make a story that involves all of them. Rodari's idea was that if all but one words in the list are chosen so that they remind the children of some familiar story, and one unrelated but familiar word is thrown in with them, children's imagination is stimulated in incorporating the unrelated concept in the familiar story. Children enjoy this activity, and produce a great many imaginative variations of the original story.

For example, suppose children are given the words 'grandmother', 'wolf', 'forest', 'cape', and 'helicopter'. The first four words may remind the children of the story of *Little Red Riding Hood.* However, the last word is completely unrelated. We must emphasize here two necessary conditions for this technique to work. The first is that the children must be familiar with the story of *Little Red Riding Hood.* If they do not know the story, or if the words used in the list do not remind them of the story, for there are several versions of the story, then this technique is not so effective in terms of stimulating imagination. The second condition is that the children must be familiar with the unrelated word as well. If they have no idea what a helicopter is, then the technique does not work either.

In this technique, the children find it interesting and challenging to make a story that incorporates the strange but familiar element (the helicopter) in the familiar story. Each child tackles this task in her or his own way. Their imagination is stimulated, and they enjoy the activity. They listen to each other's stories, and react to them enthusiastically.

Our creativity support system based on this technique was implemented in three stages. In the first stage, we showed a number of pictures to the children and asked them to describe each picture. Our aim here was to find out the concepts with which our target user group (the children) was familiar. In the second stage, we created a library of *picture elements*, where each picture element depicted an object familiar to the children. Picture elements were also organized in a semantic association network to reflect which ones of them are related and which ones are not. In the third stage, we composed a picture by combining picture elements such that all the picture elements were associated except one that was semantically distant or unrelated. So, for example, the system might generate a picture of a cow in a classroom by combining some classroom-related picture elements like desks, blackboard, notebooks, children, and teacher, and add the unrelated picture element cow. On evaluating the system, we found that the children found writing about such pictures more interesting, and they wrote longer stories.

This system was a straightforward implementation of the Rodari's technique. The point we would like to emphasize is that it is not so difficult for a computer program to add an unrelated object in a scene. However, this task is harder for people, for as soon as a concept or word is given, all the associated concepts and their corresponding words get automatically activated—it is difficult to suppress these activations and to look beyond them to find an unrelated word or concept.

The T-puzzle (Fig. 1) provides an excellent example of how our past experiences, and perceptual and conceptual associations constrain us. The puzzle has four simple wooden pieces, and the objective is to arrange them in the form of letter T. However, people have a very hard time solving this puzzle because their prior perceptual experiences keep them trying the same combinations over and over again [70].

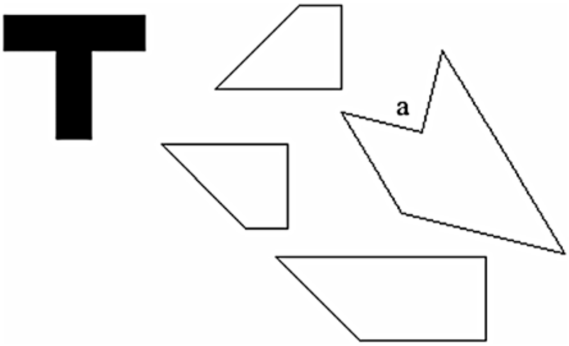


Fig. 1 The T-puzzle

In fact, if we analyze many of Rodari's techniques to stimulate creativity and imagination, they are essentially elaborate ways to get some unrelated combination of words or concepts, which then can become the stimulus for creative reflection. We have argued at length elsewhere how at least this aspect of creativity is easily modeled in computers algorithmically [37].

2.2 Modeling Creativity in Legal Reasoning Computationally

In another piece of previous research, we studied creativity in legal reasoning, and sought to model it computationally [31]. The main idea behind this approach was that creative insights often come from applying the high-level structure (or gestalt) of one situation to the low-level details of another situation. The distinction between the levels (high and low levels) is important, for if both the situations are considered from a high level, then only traditional analogy can result, which, as far as creativity is concerned, is counterproductive (see [30, 35] for detailed arguments with examples). We explain this approach with an example below.

We focused on the situations where new categories are brought into analogize or distinguish between prior cases and the new case in order to argue for a particular resolution of the new case. Our domain was a particular tax law in the US, which allowed taxpayers to deduct their home office expenses from their taxable income under certain conditions. These exception conditions were based on factors such as whether the home office was the *principal place of business* (e.g., when a doctor saw patients regularly at a home clinic), whether the employer provided office space for the employee, and so on.

In particular, among the precedents, there was a case of a high-school teacher, who claimed home office deduction, but the courts denied him because the school provided him a *suitable space:* a classroom where he could teach, an office equipped with a phone and office supplies, and so on. Let us refer to this as *Case 1.*

Another case among the precedents concerned a concert violinist. He claimed tax deduction for a studio he maintained at home where he practiced regularly. The courts allowed him the deduction arguing that for musicians, their principal activity is rehearsal, and the employer did not provide any space where the musician could rehearse. Let us call this *Case 2.*

Now consider the case of a college professor, who claims tax deduction for an office he maintained at home. In his case, the college provided him a shared office. But the taxpayer argued that because many staff members shared the office, he could not leave his books and other material safely there. Let us call this *New Case.*

With this background, *Case 1* is very similar to the *New Case,* and supports a decision against the taxpayer. When *Case 2* is applied to the facts of the *New Case,* we also get a decision against the taxpayer, because for each task that the college professor had to perform as part of his duties, there was some place (the shared office) provided by the employer.

However, when *Case 1* is activated, the category *suitable space* comes into play. Now we can reinterpret *Case 2* using this category to argue that the decision favored the taxpayer because the employer did not provide any *suitable space* to carry out the activity. With this reinterpretation, *Case 2* is rendered similar to the *New Case,* and supports a decision in favor of the taxpayer. (See [31] for details.)

It may seem a small semantic quibble to some readers, but legal arguments often hinge on such quibbles. There is another example discussed there that hinges on introducing the term *substantial.* We should also add that all the analyses and representations used in the examples were derived from actual opinions written by the judges when rendering their decisions.

The implications of this for computational systems is that we need to be able to model the process of reinterpretation, by which concepts and categories are applied to different data (for which they may not have been intended) in novel ways. It is interesting to point out that the ability to get a new insight or perspective was one of the advantages claimed for case-based reasoning when it was promoted by Riesbeck and Schank ([61], pp. 9–14). They compared and contrasted three modes of reasoning: (1) reasoning with ossified cases (rules or abstract principles), (2) reasoning with paradigmatic cases (cases with a given interpretation), and (3) reasoning with stories (cases with many possible interpretations and capable of reinterpretations). They argued that it is the third mode of reasoning that displays the most flexibility and power of having a knowledge base containing cases. However, most approaches to case-based reasoning in the 1990s and early part of the 2000s worked largely with indexed cases, which preclude this reinterpretation step. (See, for instance, [6].) But in recent years, advances in data mining and unsupervised learning techniques provide us with many new approaches to model the reinterpretation process. (See, for example, [27, 73, 74].)

2.3 Role of Low-Level Perceptual Similarities in Stimulating Novel Conceptual Associations

One of the research problems we have been working on is to assess the role of low-level perceptual similarities—namely similarities with respect to shape, color, texture, etc.—on emergent features when two images are juxtaposed. A feature related to a metaphor is considered *emergent* if it is not normally related to either of the two terms of the metaphor alone. For example, in "Her gaze, a flash of diamond", 'seduction' is an emergent feature as it is not normally related to 'gaze' or 'diamond' [19]. A major methodological problem in working with images is in determining the degree of low-level perceptual similarities between two given pictures. One alternative is to ask the participants to rate the degree of perceptual similarities between pairs of pictures, but the drawback is that when we look at a picture, conceptual and perceptual features interact heavily, and it is difficult to be certain that only perceptual features were used in determining the degree of similarity. To address this problem, we turned to image-processing programs.

In the field of machine vision, a number of algorithms have been developed for low-level visual processing. These algorithms extract features (like color, shape, texture, and so on) of images, which are analogous to features found in the early stages of visual processing in humans. So a similarity measure based on these features would reflect perceptual similarity.

We used one such image-based search system called fast image search in huge database (FISH), which compares two images based on low-level perceptual features like color, shapes, texture, etc., to get a similarity index for them [72]. We refer to this as *algorithmic perceptual similarity*. For example, consider the pair of images shown in Fig. 2. The image on the left is of the world-famous marble mausoleum *Taj Mahal* that was built by the Moghul emperor Shah Jahan in the seventeenth century. The image on the right is of wine bottles. These two images were given a high perceptual similarity index by the FISH system. In fact, the wine bottles image was retrieved by the system as a *similar* image when queried by the *Taj Mahal* image. If we examine them carefully, we can see the perceptual similarities: the tall slender minarets of the *Taj Mahal* are analogous to the shape of the

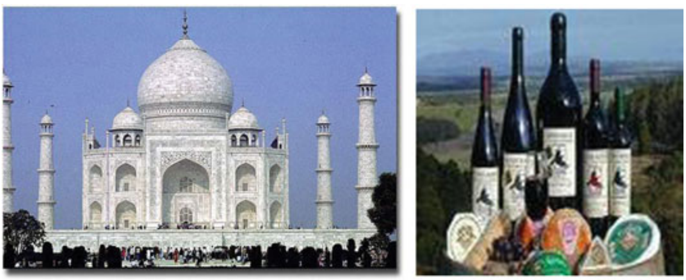


Fig. 2 An example of algorithmic perceptual similarity

wine bottles. However, when people look at these two images, they tend to focus on conceptual similarities, if they find them similar at all.

Using such stimuli, we experimentally studied how perceptual similarities correlate with people's ability to interpret pairs of images metaphorically, and with emergence of new features that are not a part of either image [55]. Our results show that a pair of perceptually similar images (in terms of color, shape, etc.) is more likely to be given a metaphorical interpretation. Here are some examples of the interpretations given to the pair of images in Fig. 2 by the participants: 'Becomes better as it grows old', 'Standing pillars of tradition', 'Beauty in taste', 'Taste of history', 'Taj for eyes, wine for tongue', 'What a waste of time'. We also found that perceptual similarity correlates positively with emergent features.

An implication of these results is that they provide yet another way in which computational approaches can aid creativity. Or, we can go as far as to claim that if this hypothesis is correct—namely that low-level perceptual similarities facilitate novel conceptual associations (among people)—then a computational system based on algorithmic approach to perceptual similarity will be quite effective in stimulating creative imagination in the viewer. Systems based on such approaches can be used for creating persuasive ads, intuitive educational material, aesthetic pleasing art, and so on [40].

2.4 Thinking like a Child: Surface Similarities and Creativity

In popular psychology, an oft-touted *mantra* for increasing creativity is: *think like a child* [10, 14, 24, 47]. There is also empirical research demonstrating the effectiveness of this technique [76]. We examine here one particular aspect of child-like thinking, namely the focus on surface similarities. There are other aspects of thinking like a child, such as functional fluidity and pretense play, which we will not consider here.

There is a large body of existing research showing that younger children tend to focus on perceptual or surface-level (for example, shape or color) similarities for categorization or for giving meaning to new words, and it is only as they get older they start to use functional, structural or other semantic similarities [17, 18, 23, 29, 53, 59, 68]. There seems to be a general agreement on this, and the researchers have emphasized, time and again, how it is the functional and structural similarities that are useful for reasoning and categorization, and surface similarities are often thought to be distracting [15].

In our collaborative work with a visual artist [41], we focused on the creative process involved in connecting two pictures by painting another picture in the middle in such a way that the trio of pictures forms one smooth portrait. This technique was involved in four *Infinite Landscape* workshops conducted at Art Museums in Japan and Europe by Mr. Ogawa over the last 5 years. Based on the

Fig. 3 A trio of pictures from *Infinite Landscape* workshop

artist's verbal recollection of the ideas that occurred to him as he drew each of the connecting pictures, we identified the microprocesses and cognitive mechanisms underlying the genesis of these ideas, and surface similarities with respect to shading, texture, and shape were found to play a key role in it.

One such trio of pictures is shown in Fig. 3. Here the pictures (9) and (10) were drawn by participants, and the Artist drew the middle picture S9. The Artist recorded the following thoughts on how he came up with the idea for S9 (this is the translation of the Artist's original comments from Japanese to English): "These two had completely different atmosphere from each other. Sketch 9, drawn by an adult participant, is a scene set at dusk; a person looking at the artist is drawn wearing a sad expression. Sketch 10 has a bright atmosphere with flowers, fountains, buildings on a hill, and a horse. Moreover, each picture had an important character in the bottom left. The idea for connecting these sketches came to me while looking at the wonderful horse in 10. I thought of putting a parent horse running nearby. Because the background color of 9 and the body color of the horse in 10 was the same, I transformed the background of 9 into the parent horse in S9, which became a nested image structure. Then I extended the baby horse and the hill with the buildings."

Here the same shading for the horse's body in 10 and the background in 9 led to the idea that the background in 9 can be morphed into the mother horse in S9, which resulted in an Escher-like nesting of pictures. Thus, this study illustrates the role of surface similarities in generating new ideas.

We now briefly discuss how surface similarities exemplify one aspect of *think-like-a-child* maxim for stimulating creativity. If we look at the perspective from developmental psychology, the progression from surface features to structural features can be explained as follows. As children grow older, they acquire more knowledge of the world and, more importantly, more knowledge of the social norms and conventions. This knowledge takes the form of semantic structures and relationships, and over time, they rely more on these relationships, and surface similarities take a back seat. But as a child gets habituated to some semantic structures, a horde of alternate possible semantic structures get lost. A major part of creativity consists in reclaiming some of these alternate semantic structures—or what Nelson Goodman [21] would call 'worlds'—that might have been. (See also [39].) Focusing on surface similarities provides one mechanisms to go back to the prestructure stage, so that alternate structures can be found. Moreover, we can extend the arguments made earlier in Sect. 2.3 to suggest that surface similarities can be naturally exploited to design computational creativity system.

3 Is Creativity Computational?

We now return back to the question we raised in the introduction: Can creativity be algorithmic? Or is it essentially a human quality? To examine such questions, let us consider two different characterizations of creativity. The first one focuses on the process by which a human being engages in a creative pursuit. If we try to think of creative people, who comes to mind? Perhaps Einstein, Mozart, Michelangelo, or Leonardo da Vinci. In the modern times, we might think of Steve Jobs. But what do we mean when we say that they are creative?

Perhaps music came naturally to Mozart. In a letter to his father on Nov. 8, 1777, he wrote: "I cannot write in verse, for I am no poet. I cannot arrange the parts of speech with such art as to produce effects of light and shade, for I am no painter. Even by signs and gestures I cannot express my thoughts and feelings, for I am no dancer. But I can do so by means of sounds, for I am a musician." Perhaps one could say that his brain was structured in a certain way that generated musical patterns naturally. Of course, what makes his work great is because of the way people have responded to his music over more than two centuries. (See also [44, 56].)

Or consider mentally different people, like the case of Stephen Wiltshire, discussed in Sacks [63]. Sir Wiltshire has an amazing ability to draw a landscape from memory after seeing it only once. Though he is diagnosed with autism, his work is highly regarded both by critics and general population. He was awarded *Member of the Order of the British Empire* for services to art in 2006. So he is no doubt a very creative person, no matter which criterion one chooses to apply.

But let us think about it a minute. What do we mean by saying that he is creative? His work has a certain style, level of details that most people cannot reach, aesthetic appeal, and all that. As with Mozart, we can go further and say that perhaps this is the way he expresses himself naturally: just like you and I might describe what we did on our last summer vacation, he draws fantastic landscapes.

We can now throw in here examples of people with schizophrenia or brain damage, savants or manic-depressive people, and so on [65]. When these people produce work that is considered creative, often this is their mode of being, and it could not have been otherwise. (See also [1, 20].) Many times the intention is missing as well.

Einstein's brain was preserved after his death so that people can study it to get any clues about the biological basis for creativity. But it is not like he was creative every day of his life. It is the impact of his theory of relativity, and its eventual acceptance by the scientific community that was a key factor in him becoming an icon of scientific creativity of the twentieth century. Moreover, Einstein was also dogmatic at times, perhaps the most famous case being his rejection of Alexander Friedmann's expanding universe hypothesis [69].

If we were to model Einstein's creative process, what would we model? There have been some computational models of scientific discovery, but they almost always greatly simplify the process by putting a number of assumptions in place as to what is significant and what is not. At that point, it is not clear at all if they are modeling the actual mental process of the creative person at the time of creative act. (See also [6].)

Such examples suggest that the so-called creative humans use a variety of heuristics, some of them consciously and some subconsciously, for creating artifacts or for problem solving. Many of these heuristics can be mechanized, and in principle there seems to be no reason to consider any of them non-algorithmic.

The second characterization of creativity focuses on the nature of creative artifacts. It takes only the audience's perspective, so the creator is not even mentioned. We refer to Barthes' [3] articulation: "We know that to restore to writing its future, we must reverse its myth: the birth of the reader must be ransomed by the death of the Author," though he traced this view to even earlier scholars. Though not everyone subscribes to this extreme position, most accounts of creativity do acknowledge the role of audience [11, 12, 28, 49].

In the audience-based view of creativity, it is generally accepted that in order for an artifact to be deemed creative, it must be *novel* and *useful*. We have argued above (Sect. 2.1) that *novelty* is cognitively difficult for people because we are constrained by our previous conceptual associations. Researchers who study creativity have come up with various techniques to overcome this difficulty. However, computers and AI systems, which do not have any such associations, have a great advantage here, for they can search the unchartered areas of novel concepts and conceptual combination more systematically [25, 37].

However, to automatically assess the *usefulness* of created artifacts is a different cup of tea altogether. As the usefulness is necessarily from a human point of view, the question becomes: Can an algorithm capture usefulness to humans? Here, we can distinguish two different aspects of usefulness. One is aesthetics, which relates to artistic creativity. In this regard, there has been some research to suggest that at least some of our aesthetic values are hardwired in the structure of the brain [60, 77]. Moreover, machine-learning techniques have been applied to *learn* about the cultural preferences of an audience based on the past data. For instance, Ni et al. [54] trained their program with the official UK top-40 singles chart over the past 50 years to learn as to what makes a song popular. A program like this might successfully predict, for instance, the winner of the future Eurovision competitions. However, a limitation of these approaches is that they cannot predict drastic changes in the aesthetic values and tastes: for example, atonal music or abstract art. Moreover, creativity is not the same as popularity. So to be able to predict whether a song, or a book, or a video will become popular [71] is not the same thing as evaluating their creativity.

This problem becomes more severe when we move beyond arts, and consider creativity in problem solving, and in science and technology. Here the usefulness of a novel and creative idea comes down to simply whether it works. This clearly has an objective component, for in a sense it is the reality that determines whether the idea works or not. History of science and technology is full of many interesting and novel ideas that did not work. Prehistory of flight [26] is a rich domain of examples where many novel ideas that were based on numerous observations, experimentations, and in which inventors had complete faith, did not work at all. The following examples provide further support for this argument (see also [34, 57]):

1. Schön ([67], p. 259) noted that a product-development team working on the synthetic fiber paintbrush problem considered *painting as masking a surface* metaphor, which was quite a novel idea, but it led to no useful insight.
2. Yolanda Baie, a food stand operator and owner, petitioned to have the kitchen of her house, where she prepared food sold at the food stand, qualify as a *home office* for the purpose of tax deduction (Baie vs. C.I.R., 74 T.C. 105), using the argument that her kitchen was a *manufacturing facility* for her business. The judges, while finding the argument 'ingenious and appealing,' ruled it 'insufficient' nonetheless.
3. John Casti ([5], pp. 7–10) comments on the fate of Immanuel Velikovsky's theory as outlined in Worlds in Collision, which hypothesized Earth's encounters with a large comet expelled from Jupiter and provided explanations for many biblical events. Velikovsky's theory proposes a novel understanding of the Solar system, but the scientific community has not accepted it.

Considering such examples, we suggest that this *usefulness* aspect of creativity remains essentially non-algorithmic, but not because humans are special, and cognitive processes cannot be computational, but because nature is not bounded by the limits of our cognitive models. Creativity, in this view, represents the open-endedness of our interaction with the environment, and cannot be captured in a cognitive or computational model. Nonetheless, we can have computational models of creativity in limited domains, and computational systems can be designed to stimulate and enhance general creativity in people.

4 Conclusions

To summarize the main arguments of this paper, we would like to rearticulate them in another way. It is generally accepted that the two main characteristics of creativity are originality and intelligibility: the product must be novel or the process must generate a new perspective; and the product or the generated perspective must be intelligible in order to be useful for at least some audience [2, 64, 75]. For novelty, research on real-world creativity shows that it is difficult for people to step out of their conventional and habitual conceptual associations. To overcome this inertia, several methods like making the familiar strange [22], concept displacement [66], bisociation [43], lateral thinking [13], estrangement [62], conceptual blending [16], and so on, have been proposed in the literature. However, computers do not have this inertia, and so they can be very effectively used to generate novel ideas. This argument has been presented in more detail elsewhere [37]. Our experience in developing creativity-assistive systems (reviewed in Sect. 2 above) lends supports to this hypothesis.

However, when it comes to incorporating usefulness of the generated perspective or idea, we have argued that, in general, it is not possible to capture this aspect of creativity algorithmically. The reason is simply that when a new object or style is introduced, people react to it in different ways. Sometimes they adapt to it right away; at other times they do not find it interesting or useful at first, but the same object or style introduced at a later time becomes a big success; and sometimes they do not find it useful at all, in spite of the efforts made by the creators to convince them otherwise.

Nevertheless, one cannot rule out the possibility that in limited domains we might be able to characterize usefulness algorithmically, and to design and implement computer systems that can generate statistically a larger number of useful and interesting artifacts and ideas. So combining this with novelty-generating systems, we can have computer systems that are creative. Systems like Aaron exemplify this approach.

However, even in a limited domain, once usefulness is characterized algorithmically, it loses its novelty, and gradually ceases to be creative. (See, for instance, the model of literary style change proposed by Martindale [50].) So while, we may be able to model some aspect of creativity within a style (with respect to usefulness), it remains doubtful whether creative changes in styles can be modeled successfully in a universal way. Again, to emphasize, novelty can be modeled—it is relatively easy to computationally generate new styles, but the problem is to incorporate which styles will be successful (meaning people will adapt to them and find them useful). Therefore, we claim that this usefulness aspect of creativity will always remain the last frontier for computational modeling techniques.

References

1. Abraham, A., Windmann, S., McKenna, P., Güntürkün, O.: Creative thinking in schizophrenia: the role of executive dysfunction and symptom severity. Cogn. Neuropsychiatry **12**(3), 235–2581 (2007)
2. Amabile, T.M.: Creativity in context. Westview Press, Boulder (1996)
3. Barthes, R.: The Death of the Author. In: Barthes, R. Image Music Text (trans. S. Heath), Fontana, London (1977). Original work published 1967
4. Boden, M.A.: Computer models of creativity. AAAI AI Mag. **30**(3), 23–33 (2009)
5. Casti, J.L.: Paradigms Lost: Images of Man in the Mirror of Science. William Morrow & Co., New York (1989)
6. Chalmers, D.J., French, R.M., Hofstadter, D.R.: High-Level perception, representation, and analogy: A critique of artificial intelligence methodology. J. Exp. Theor. Artif. Intell. **4**(3), 185–211 (1992)
7. Chordia, P., Rae, A.: Tabla Gyan: an artificial tabla improviser. In: Proceedings of the International Conference on Computational Creativity: ICCC-X, Lisbon, Portugal (2010)

8. Cohen, H.: On Modeling of Creative Behavior. Rand Corporation Report Series, P. 6681 (1981)
9. Colton, S.: Creativity versus the perception of creativity in computational systems. In: Proceedings of the AAAI Spring Symposium on Creative Systems (2008)
10. Corner, L.: How to think like a child. The Independent, (26 October, 2010). http://www.independent.co.uk/life-style/health-and-families/features/how-to-think-like-a-child-2116291.html (2012). Accessed 15 Oct. 2012
11. Cropley, D.H., Kaufman, J.C., Cropley, A.J.: Measuring creativity for innovation management. J. Technol. Manag. Innov. **6**(3), 13–30 (2011)
12. Csikszentmihalyi, M.: Creativity: Flow and the psychology of discovery and invention. Harper Collins, New York (1996)
13. de Bono, E.: New Think: The Use of Lateral Thinking in the Generation of New Ideas. Basic Books, New York (1975)
14. Ewedemi, W.: 6 Blocks to creative thinking and how to solve them. http://thenetng.com/2011/08/6-blocks-to-creative-thinking-and-how-to-solve-them/ (2011). Accessed 15 Oct. 2012
15. Faries, J.M., Schlossberg, K.R.: The effect of similarity on memory for prior problems. In: Proceedings of the 16th annual conference of the Cognitive Science Society, pp. 278–282 (1994)
16. Fauconnier, G., Turner, M.: The Way We Think: Conceptual Blending and the Mind's Hidden Complexities. Basic Books, New York (2002)
17. Gentner, D.: Metaphor as structure mapping: The relational shift. Child Dev. **59**, 47–59 (1988)
18. Gentner, D., Rattermann, M. J.: Language and the career of similarity. In: Gelman, S.A., Brynes, J.P. (eds.) Perspectives on Language and Thought: Interrelations in Development. Cambridge University Press, pp. 225–277 (1991)
19. Gineste, M.-D., Indurkhya, B., Scart, V.: Emergence of features in metaphor comprehension. Metaphor Symbol **15**(3), 117–135 (2000)
20. Glicksohn, J.: Schizophrenia and Psychosis. In: Runco, M.A., Pritzker, S.R. (eds.) Encyclopedia of Creativity, 2nd edn. Academic Press, New York (2011)
21. Goodman, N.: Ways of worldmaking. Hackett, New York (1978)
22. Gordon, W.J.J.: Synectics: The Development of Creative Capacity. Harper & Row, New York (1961)
23. Gottfried, G.M.: Using metaphors as modifiers: children's production of metaphoric compounds. J Child Lang. **24**(3), 567–601 (1997)
24. Greenwood, C.: Thinking like a child. Success. http://www.success.com/articles/890-think-like-a-child, 4, Nov. 2009. Accessed 01 Oct. 2014
25. Harry, H.: On the role of machines and human persons in the art of the future. Pose **8**, 30–35 (1992)
26. Hart, C.: The prehistory of flight. University of California Press (1985)
27. Hofstadter, D.: The Fluid Analogies Research Group: Fluid Concepts and Creative Analogies. Basic Books, New York (1995)
28. Horn, D., Salvendy, G.: Consumer-based assessment of product creativity: a review and reappraisal. Hum. Factors Ergon. Manuf. Serv. Ind. **16**(2), 155–175 (2006)
29. Imai, M., Gentner, D., Uchida, N.: Children's theories of word meaning: the role of shape similarity in early acquisition. Cogn. Dev. **9**, 45–75 (1994)
30. Indurkhya, B.: Metaphor and Cognition. Kluwer Academic Publishers, Dordrecht (1992)
31. Indurkhya, B.: On modeling creativity in legal reasoning. In: Proceedings of the Sixth International Conference on AI and Law, pp. 180—189. Melbourne, Australia (1997)
32. Indurkhya, B.: An Algebraic Approach to Modeling Creativity in Metaphor. In: Nehaniv, C.L. (ed.) Computation for Metaphors, Analogy and Agents. LNAI, vol. 1562, pp. 292–306. Springer, Berlin (1999)
33. Indurkhya, B.: Emergent representations, interaction theory, and the cognitive force of metaphor. New Ideas Psychol. **24**(2), 133–162 (2006)
34. Indurkhya, B.: Rationality and reasoning with metaphors. New Ideas in Psychol. **25**(1), 16–36 (2007)

35. Indurkhya, B.: On the role of metaphor in creative cognition. In: Proceedings of the International Conference on Computational Creativity: ICCC-X, Lisbon, Portugal (2010)
36. Indurkhya, B.: Thinking like a child: the role of surface similarities in stimulating creativity. In: Stojanov, G., Indurkhya, B. (eds.) Proceedings of the AAAI Spring Symposium on Creativity and (Early) Cognitive Development, pp. 38–43. AAAI Press, Palo Alto, CA (2013)
37. Indurkhya, B.: Computers and creativity. In: Veale, T., Feyaerts, K., Forceville, C.J. (eds.) Creativity and the Agile Mind, pp. 61–79. De Gruyter Mouton, Berlin (2013)
38. Indurkhya, B.: On the role of computers in creativity-support systems. In: Skulimowski, A.M. J. (ed.) Proceedings of KICSS'2013, pp. 233–244, Progress and Business Publishers, Kraków (2013)
39. Indurkhya, B. A perspective on cognition, experience and creativity. In: Manjaly, J., Indurkhya, B. (eds.) Cognition, Experience and Creativity. Orient Blackswan, New Delhi (2014)
40. Indurkhya, B., Kattalay, K., Ojha, A., Tandon, P.: Experiments with a creativity-support system based on perceptual similarity. In: Fujita, H., Zualkernan, I. (eds.) New Trends in Software Methodologies, Tools and Techniques, pp. 316–327. IOS Press, Amsterdam (2008)
41. Indurkhya, B., Ogawa, S.: An empirical study on the mechanisms of creativity in visual arts. In: Miyake, N., Peebles, D., Cooper, R.P. (eds.), Proceedings of the 34th Annual Conference of the Cognitive Science Society, pp. 1727–1732. Cognitive Science Society, Austin, Texas (2012)
42. Ishii, Y., Indurkhya, B., Inui, N., Nose, T., Kotani, Y., Nishimura, H.: A system based on Rodari's 'Estrangement' Principle to promote creative writing in children. In: Proceedings of Ed-Media & Ed-Telecom 98, pp. 1685–1686, Freiburg, Germany (1998)
43. Koestler, A.: The Act of Creation. Hutchinsons, London (1964)
44. Kozbelt, A.: Factors affecting aesthetic success and improvement in creativity: a case study of the musical genres of mozart. Psychol. Music **33**(3), 235–255 (2005)
45. Krzeczkowska, A., Colton, S., Clark S.: Automated collage generation—with intent. In: Proceedings of the International Conference on Computational Creativity: ICCC-X, Lisbon, Portugal (2010)
46. Kurzweil, R.: Cybernetic Poet. http://www.kurzweilcyberart.com/poetry/ (2001). Accessed 29 Jan 2012
47. Lehrer, J.: Childish creativity. http://scienceblogs.com/cortex/2010/03/25/childish-creativity/, 25 Mar 2010. Accessed 15 Oct. 2012
48. López, A.R., Oliveira, A.P., Cardoso, A.: Real-time emotion-driven music engine. In: Proceedings of the International Conference on Computational Creativity: ICCC-X, Lisbon, Portugal (2010)
49. Maher, M.L.: Evaluating creativity in humans, computers, and collectively intelligent systems. In: Proceedings of DESIRE'10: Creativity and Innovation in Design, Aarhus, Denmark (2010)
50. Martindale, C.: The Clockwork Muse: the Predictability of Artistic Change. Basic Books, New York (1990)
51. McCorduck, P.: Aaron's Code: Meta-art, Artificial Intelligence, and the Work of Harold Cohen. W. H. Freeman, New York (1991)
52. Monteith, K., Martinez, T., Ventura, D.: Automatic generation of music for inducing emotive response. In: Proceedings of the International Conference on Computational Creativity: ICCC-X, Lisbon, Portugal (2010)
53. Namy, L.L., Gentner, D.: Making a silk purse out of two sow's ears: young children's use of comparison in category learning. J. Exp. Psychol. Gen. **131**(1), 5–15 (2002)
54. Ni, Y., Santos-Rodriguez, R., Mcvicar, M., De Bie, T.: Hit song science once again a science?. In: Proceedings of the 4th International Workshop on Machine Learning and Music: Learning from Musical Structure, Sierra Nevada, Spain (2011)
55. Ojha, A., Indurkhya, B.: Perceptual vs. conceptual similarities and creation of new features in visual metaphor. In: Kokinov, B., Holyoak, K., Gentner, D. (eds.) New Frontiers in Analogy Research. New Bulgarian University Press, Sofia (2009)

56. Painter, K.: Mozart at work: biography and a musical aesthetic for the emerging German bourgeoisie. Music. Q. **86**(1), 186–235 (2002)
57. Palmer, S.: Human ontology and rationality. Aldershot, Avebury (1992)
58. Pérez y Pérez, R., Negrete, S., Peñalosa, E., Ávila, R., Castellanos, V., Lemaitre, C.: MEXICA-Impro: a computational model for narrative improvisation. In: Proceedings of the International Conference on Computational Creativity: ICCC-X, Lisbon, Portugal (2010)
59. Pierce, K.A., Gholson, B.: Surface similarity and relational similarity in the development of analogical problem solving: isomorphic and nonisomorphic transfer. Dev. Psychol. **30**(5), 724–737 (1994)
60. Ramachandran, V.S., Hirstein, W.: The science of art: a neurological theory of aesthetic experience. J. Conscious. Stud. **6**(6–7), 15–51 (1999)
61. Riesbeck, C.K., Schank, R.C.: Inside case-based reasoning. Lawrence Erlbaum & Associates, Hillsdale (1989)
62. Rodari, G.: The Grammar of Fantasy (J. Zipes, Trans.) Teachers & Writers Collaborative, New York (1996)
63. Sacks, O.: An Anthropologist on Mars: Seven Paradoxical Tales. Alfred A. Knopf, New York (1995)
64. Sarkar, P., Chakrabarti, A.: Development of a method for assessing design creativity. In: Proceedings of the International Conference on Engineering Design ICED'07, pp. 28–31, Cite des Sciences et de L'Industrie, Paris, France (2007)
65. Sawyer, K.: Explaining Creativity. Oxford University Press (2006)
66. Schön, D.A.: Displacement of Concepts. Humanities Press, New York (1963)
67. Schön, D.A.: Generative metaphor: a perspective on problem-setting in social policy. In: Ortony, A. (ed.) Metaphor and thought, pp. 154–283. Cambridge University Press, Cambridge (1979)
68. Siltanen, S.A.: Effects of three levels of context on children's metaphor comprehension. J. Genet. Psychol. Res. Theory Hum. Dev. **150**(2), 197–215 (1989)
69. Singh, S.: Big Bang. Harper Collins, New York (2004)
70. Suzuki, H., Hiraki, K.: The constraint relaxation process as theory change: toward theorizing the process of insight. (In Japanese.) In: Proceedings of the Japanese Cognitive Science Society's Special Interest Group on Language & Learning 97 (1), pp. 33–42 (1997)
71. Szabo, G., Huberman, B.A.: Predicting the popularity of online content. Commun. ACM **53** (8), 80–88 (2010)
72. Tandon, P., Nigam, P., Pudi, V., Jawahar, C.V.: FISH: A practical system for fast interactive image search in huge data bases. In: Proceedings of 7th ACM International Conference on Image and Video Retrieval (CIVR'08), Niagara Falls, Canada (2008)
73. Veale, T.: Re-Representation and creative analogy: a lexico-semantic perspective. New Gener. Comput. **24**(3), 223–240 (2006)
74. Veale, T., Hao, Y.: A fluid knowledge representation for understanding and generating creative metaphors. In: Proceedings of COLING 2008, pp. 1–5. Manchester (2008)
75. Yong, K., Lander, M.W., Mannucci, P.V.: Novelty x usefulness: actor-level effects and cultural influences on creativity in organizations. In: Proceedings of the 35th DRUID Celebration Conference, Barcelona, Spain, 17–19 June (2013)
76. Zabelina, D.L., Robinson, M.D.: Child's play: facilitating the originality of creative output by a priming manipulation. Psychol. Aesthet. Creativity Arts **4**, 57–65 (2010)
77. Zeki, S.: Inner Vision: An Exploration of Art and the Brain. Oxford University Press, Oxford (UK) (2000)

Visualization of N-Gram Input Patterns for Evaluating the Level of Divergent Thinking

Taro Tezuka, Shun Yasumasa and Fatemeh Azadi Naghsh

Abstract Creative thinking is often considered to be difficult, one reason because humans tend to be trapped in the same patterns of thinking and cannot easily come up with a totally new combination of concepts. In other words, humans are not talented at evenly exploring combinatorial space. In order to visualize how strong this tendency is, we implemented a system that asks the subject to type in a long sequence of numbers. The system then counts the frequency of the appearance of the same subsequences using n-grams. We called it "the Creativity Test." It measures one's efficiency of exploring a wider part of a combinatorial space without being caught in few patterns. The result is assumed to be related to the ability of divergent thinking, which is considered to be important in creative thinking. When we tested the system on a group of subjects, we discovered that, for most of them, surprisingly long n-grams appeared frequently, making the subjects realize how inefficient they were at coming up with new combinations.

Keywords Creativity · Divergent thinking · Patterns · Heuristics · Visualization · N-gram

T. Tezuka (✉)
Faculty of Library, Information and Media Science,
University of Tsukuba, Tsukuba 305-8550, Japan
e-mail: tezuka@slis.tsukuba.ac.jp

S. Yasumasa · F.A. Naghsh
Graduate School of Library, Information and Media Studies,
University of Tsukuba, Tsukuba 305-8550, Japan
e-mail: yasumasa@slis.tsukuba.ac.jp

F.A. Naghsh
e-mail: atena@slis.tsukuba.ac.jp

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_18

1 Introduction

Creativity, or the ability to create new ideas, is a fundamental mechanism that drives the advancement of human society. It is not easy to actually define creativity, and many theories have been proposed. If we could concisely define what it is to make an idea new, it would be easier to implement a computer program that comes up with new ideas. In an abstract sense, however, a new idea is nothing but a new combination of existing concepts. This is because unless it can be expressed using existing concepts, it cannot be communicated to others. Therefore, we put forth that creative thinking is the process of searching through combinations to find ones that are useful but have not been utilized. We can formalize this mathematically as follows.

Let Ω be a space of existing concepts. The space of combinations, or the combinatorial space, can be expressed as 2^{Ω}. We can see why searching through combinatorial space and finding a useful combination of concepts is difficult, since 2^{Ω} is much larger than Ω.

The size of combinatorial space is not the sole reason that inhibits humans from coming up with novel ideas. Humans may be quite inefficient in searching through the combinatorial space to find useful ones. For example, in various techniques that aim to nurture creative thinking, "divergent thinking" is considered a fundamental phase that must be carried out intentionally [12, 14]. Here, the goal is to expand the line of thought to try out as many combinations of concepts as possible. The importance of divergent thinking points out that it is not easy for humans to think freely, without intentionally trying: our thinking is often constrained to existing combinations of concepts. Searching evenly through the space of combinations seems to be difficult for humans to do, and that is what we are trying to demonstrate in this paper.

There are two primary reasons for difficulties with creative thinking.

1. The combinatorial space is very large.
2. The search steps of the combinatorial space is likely to check the same combinations repeatedly, thus reducing search efficiency.

While the first reason is intrinsic to the nature of combinatorial problems, the second one has its roots in the nature of the human mental process. In this paper, we show that humans are inclined to repeat the same patterns. In other words, we are not evenly searching through combinatorial space. Certain combinations are more likely to be visited than others, probably because humans are heavily dependent on heuristics when exploring combinatorial space rather than performing sequential exhaustive searches. Heuristics is too strong in that it inhibits humans from creating random combinations evenly. Of course, heuristics play an important role in finding worthwhile ideas when the combinatorial space is too big. There are cases, however, in which a heuristic turns out to be inappropriate for the problem. In such cases, searching thorough the combinatorial space without overlap is probably more efficient. Our work is an attempt to evaluate heuristics based on its efficiency.

2 Motivation

Knowing one's inadequate tendency is the first step to correcting it. Obtaining feedback is often a very effective means of control. In this paper, we describe a system that shows how likely humans are to become constrained in a small subset of combinatorial space and how inefficient they are in finding a new combination of ideas. The proposed system helps users realize the strength of the tendency to repeat patterns and shows why it is necessary to think freely with intention rather than just assuming that even searching can be easily reached. It is shown that "random input" is not actually so.

In some cases, exhaustive checking might be preferable to heuristics because our "assumedly random" searches are not efficient and might not even be able to find a new combination. Indeed, truly random numbers are not easily created by computers, as evidenced in the tremendous efforts of computer scientists to create better pseudorandom numbers. Perhaps in nature (as in quantum physics), randomness is created without any cost, but for deterministic systems (such as classical physics), even pseudorandomness is not easily obtained.

Although listing numbers may seem not directly relevant to creativity, we believe that it is a useful abstraction of a part of creative thinking. Exploring a wider part of a combinatorial space, without being caught in few patterns, is important for divergent thinking, which is often considered as an integral part of creativity [12, 14].

It is true that the number listing task does not capture all aspects of human creativity. In ill-defined problems that occur in real life, the rules themselves are difficult to define, being dependent to the context [15]. In such a case, the task is much more difficult than just following the rule of listing numbers. However, breaking down a large problem into smaller parts and examining each component is a common practice in scientific research. We therefore pursue this research to examine whether listing numbers in a random order itself is difficult for humans to do.

3 Method

We propose a method for visualizing how humans are likely to repeat the same pattern that uses a sequence of numbers generated by an individual. We ask participants to create a long sequence of decimal numbers, e.g., integers from 0 to 9, and to type them using a keypad-like interface run on a Web browser.

In the following, the "pattern" we discussed previously corresponds to a subsequence within a sequence of numbers typed by the participant. If numbers appear randomly, it means they are independent from the preceding numbers. This is sometimes called a "Markov property." For example, "2" should appear at the probability of $\frac{1}{10}$ after "1". If it appears much more frequently, we can assume that it is not random. Also, there can be longer patterns: for example, following "82", "4" is much more likely to appear than other numbers.

Fig. 1 Example n-grams

54335534876405455335543

2-gram	frequency
54	3
43	2
33	2
35	1
55	2
…	…

3-gram	frequency
543	2
433	1
335	1
355	1
553	2
…	…

In general, let x be a probabilistic vector representing a sequence, with x_i as its ith component. a_j represents an integer from 0 to 9. The probability of the occurrence of numbers conditioned by the previous numbers can be expressed as

$$p(x_i = a_i | x_{i-1} = a_{i-1}, ..., x_{i-n+1} = a_{i-n+1}, \theta) \tag{1}$$

We can estimate the conditional probability of the occurrence of numbers using the frequencies of n-grams. An n-gram is a data structure that stores the frequency of each subsequence of length n [9]. For example, in the sequence *ABAAB*, the 2-gram consists of (*AA*: 1), (*AB*: 2), (*BA*: 1), and (*BB*: 0). In other words, an n-gram stores the frequency of sequential patterns. Figure 1 shows the n-grams of a sequence for various values of n.

Using n-grams, the conditional probability can be estimated as follows:

$$\bar{\theta} = \frac{m(x_i = a_i, x_{i-1} = a_{i-1}, \ldots, x_{i-n+1} = a_{i-n+1})}{m(x_{i-1} = a_{i-1}, \ldots, x_{i-n+1} = a_{i-n+1})} \tag{2}$$

When there are s symbols, the chance level that a specific subsequence of length n appears in a sequence of length m is $(m - n + 1)s^{-n}$. Therefore, we can use n-grams to determine if the probability that a certain pattern occurs is above the chance level.

In our system, n-grams with different n values are calculated and visualized. By repeatedly using the system, users are expected to learn to identify when they are thinking “freely”. Such a feeling is often difficult to capture without the aid of a system like ours, since humans constantly repeat the same patterns without realizing it. Our method can be considered a feedback device for controlling the mental state associated with divergent thinking.

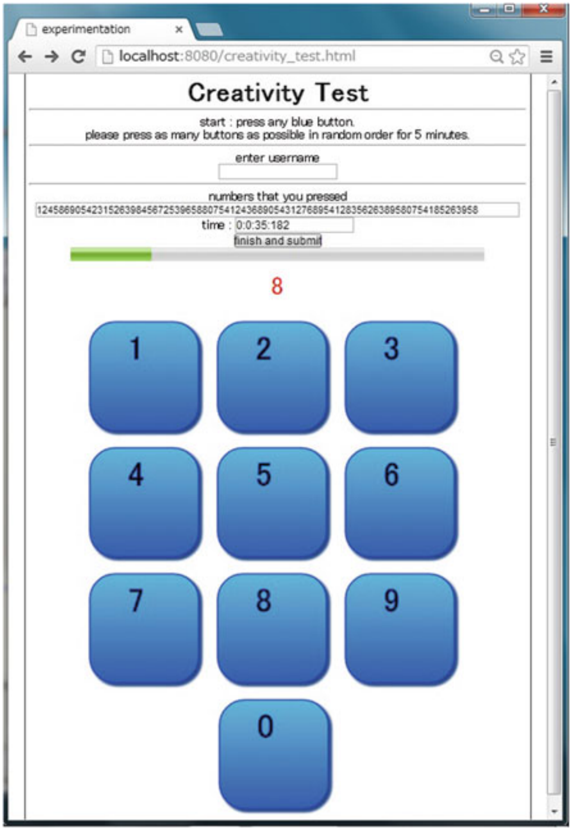


Fig. 2 The input interface

4 Implementation

The proposed system, "Creativity Test," evaluates user tendencies to repeat the same patterns [19]. We implemented it using JavaScript and Ruby, employing the client–server architecture. Participants could take part in the experiment on a Web browser. All user input was recorded in the server.

Each trial consists of three phases: the input phase, the analysis phase, and the visualization phase. The input interface consists mainly of a numerical keypad. The user is asked to type in numbers by pressing buttons on the Web browser as many times as possible in as random an order as they can. While the user is typing numbers, the sequence that has already been typed appears on the interface, but no n-gram is visualized yet. The user types a sequence for a set amount of time. In our experiment, we set it to 3 min. Figure 2 shows the interface for the input phase.

After the time is up, the system analyzes the sequence and obtains n-grams. This can be done by scanning the sequence from the beginning to the end once. If the

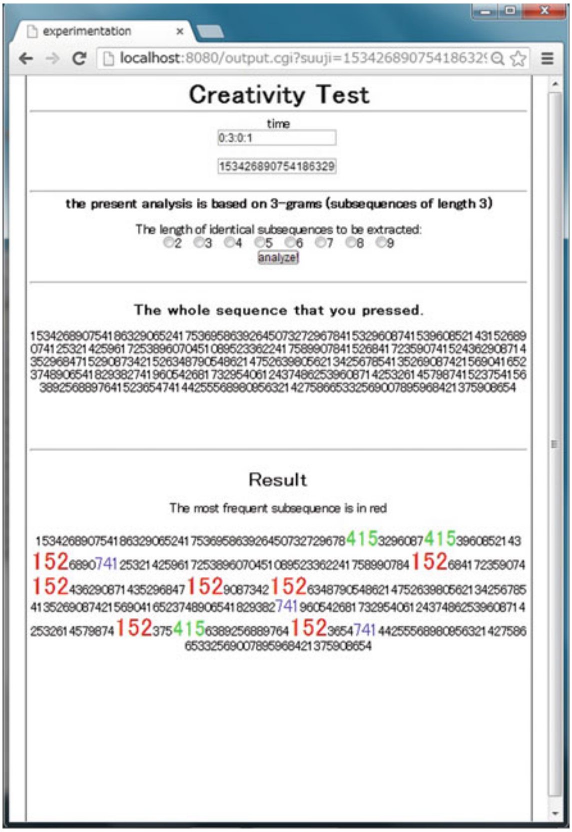
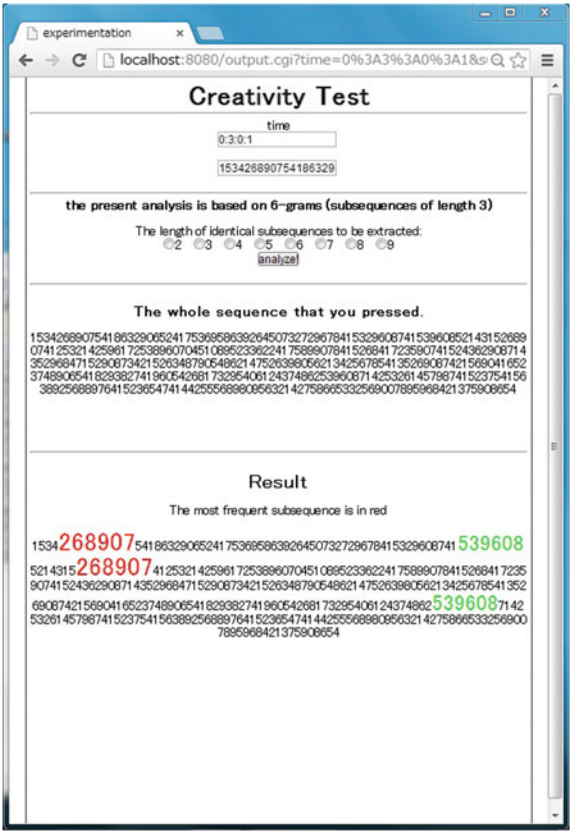


Fig. 3 The visualization interface, showing 3-gram

length of the sequence is m, the computation time is of order $O(m)$. The analysis phase therefore does not take much time.

In the visualization phase, the original sequence is presented with emphasis on the extracted n-grams. The most frequent subsequence (n-gram) is indicated by red, the second most frequent one by green, and the third by blue. They are also presented using a larger font. The n value for an n-gram can be changed by using the toggle buttons at the top of the interface. Figures 3 and 4 show the interface for the visualization phase. They show the result for 3-gram and 6-gram, respectively. The user's account, the time length of the trial, the start time, and the sequence itself are recorded in the database run on the server to enable further analysis.

Fig. 4 The visualization interface, showing 6-gram



5 Experiments

We performed experiments to see if the same patterns actually occur frequently in the sequences of numbers created by the participants. Eight individuals, four males and four females, all of whom were graduate and undergraduate students, participated in the experiment.

The participants were asked to type as many numbers as possible as randomly as they could. Tables 1 and 2 summarize the results and show the frequency of the most frequent n-gram for different values of *n*.

Table 1 Frequency of n-grams (1)

User ID	N for n-gram	Subseq.	Freq.
1	1	5	49
1	2	59	10
1	3	524	5
1	4	5246	3
1	5	30542	2
1	6		0
2	1	2	60
2	2	98	15
2	3	123	5
2	4	5412	4
2	5	14526	2
2	6	456982	2
2	7		0
3	1	5	35
3	2	14	7
3	3	321	3
3	4	3698	2
3	5		0
4	1	1	38
4	2	13	12
4	3	413	6
4	4	1326	5
4	5	41326	4
4	6	541326	3
4	7	1326490	2
4	8		0
5	1	5	48
5	2	54	10
5	3	254	4
5	4	2547	2
5	5	24856	2
5	6		0

Table 2 Frequency of n-grams (2)

User ID	N for n-gram	Subseq.	Freq.
6	1	5	55
6	2	53	17
6	3	253	8
6	4	2536	4
6	5	25369	3
6	6	425369	2
6	7	4253690	2
6	8	60751432	2
6	9	607514329	2
6	10		0
7	1	5	60
7	2	80	17
7	3	235	8
7	4	9562	4
7	5	45123	3
7	6	095623	3
7	7	8095623	2
7	8	85987415	2
7	9	859874154	2
7	10		0
8	1	5	44
8	2	07	17
8	3	807	11
8	4	5807	7
8	5	58074	4
8	6	580743	3
8	7	4123690	2
8	8		0

5.1 Discussion

There could be various reasons why humans tend to repeat the same patterns. It may come from the individual's familiarity with certain number sequences. The semantic meanings of certain numbers may affect as well.

We should also consider the effect of the interface. Since most participants used a mouse to move the pointer over the numerical keypad displayed on a Web browser, patterns may result from a tendency to repeat the same arm movements. Some order of pressing the buttons might be easier than others. For example, those that result from circular movement of the arm might occur more frequently. To determine if the effect comes purely from the mental process, we must separate various factors and perform further experiments. We plan to conduct experiments using different input interfaces. We should also compare for each subject the results from different days, and see if the same patterns appear.

In our test, computers can perform very well, using pseudorandomly generated sequences of numbers. It does not mean, however, that computers are creative by itself. In creating thinking, different combinations of ideas must be selected and filtered to obtain useful outcomes. In other words, convergent thinking is necessary, in addition to divergent thinking. It therefore shows the importance of human–computer collaboration in producing creative outputs.

Based on this result, we propose several usages of the Creativity Test. First, it can be used to train a person to perform better in creative tasks. Having feedbacks to one's action is often very effective for improving performance. By visualizing how random the output sequence was, the user might be able to learn to intentionally produce a more random sequence. This "feeling" of being random could be applied to other tasks as well. Whether such effect can be observed waits further experiments, but is nevertheless interesting to explore.

Second, it can be used for experiments to measure how environmental factors may affect the performance of creative activity. There are already various tests intended to measure the level of creativity, based on solving puzzles or problems. However, many of them are time-consuming and not practical. It also takes much time to create such puzzles and problems too. Since our test requires only a small amount of time to perform, we expect this to be useful for conducting research on measuring environmental factors on creativity.

It would also be interesting to analyze notes taken by creative individuals, for example, scientists and engineers. Whether they also have tendency to repeat the same patterns in combining concepts when they are trying to create a new idea. This would involve much of semantic processing and cannot be implemented so easily, but extending our method to higher order concepts is one direction we would like to pursue. Since not all creative individuals are taking organized notes, collaboration with techniques developed in research on lifelogs may become necessary [17].

6 Related Work

6.1 Defining Creativity

The conventional idea of creativity as the production of something novel and useful does not sufficiently accommodate the complexity of this concept. The 4-Ps model that was originally proposed by Rhodes [13] provides a more detailed structure for understanding creativity [3, 16]. This model assumes four specific perspectives on creativity: creative persons, creative processes, creative products, and creative press or environments. Research on creative persons focuses on the abilities of individuals, personality factors, and biographical backgrounds. Research on creative processes focuses on questions related to creative problem solving. Research on creative products explains the nature of the outcome of the creative process and discusses the measurement thereof. Research on creative press investigates the relationship between humans and their environment. In other words, this type of research typically focuses on the organizational level.

Generally, creativity or creative performance is evaluated using one of these criteria: fluency, originality, and flexibility [20]. Fluency refers to the generated number of nonredundant ideas, insights, problem solutions, or products. Originality, one of the defining characteristics of creativity, refers to the infrequency of the ideas, insights, problem solutions, or products that are being invented [6]. Flexibility, a core component of creative performance [6], manifests itself in the use of different cognitive categories and perspectives, and the use of broad and inclusive cognitive categories [2, 11].

6.2 Quantification of Creativity

Durand and Van Huss [4] compared idea-generating performance using a software package called Idea Generator with a process that used no software. Participants were asked to read two cases and suggest suitable solutions, using the software for one case and nothing for the other. Dependent measures were the quantity of alternatives generated, originality, depth, detail, and an overall assessment of creativity. The results indicated that the software helped generate more alternatives, but the ideas so generated were less creative than when no software was used. The effect of software generating more alternatives but less creativity was observed for both high-creative and low-creative individuals.

Malaga [8] conducted a laboratory experiment to validate a theoretical model, based on the dual coding theory and the theory of associative creativity, which suggests that combining word stimuli with picture stimuli may be effective for improving idea generation and creativity. Picture and word stimuli were used to elicit ideas in response to the task of listing as many new delicious ice cream flavors as possible. Subjects completed the task using the software package developed by Microsoft Access ’97e. The number of ideas generated and the creativity of those ideas was analyzed by two judges who were ice cream lovers. Each judge was asked to rate the novelty and feasibility of each flavor based on a 5-point Likert scale. While the theoretical model did not provide enough support, the results indicate that the stimulus mode plays a role in the efficacy of individual creativity support systems (ICSSs). Specifically, the use of picture stimuli, either by itself or in conjunction with word stimuli, leads to ideas with a higher level of creativity than the use of just word stimuli.

6.3 Novelty and Usefulness of Ideas

Massetti [10], who has a theoretical model based on an extensive analysis of the creativity literature, used a controlled laboratory experiment entailing a 1   4 design where participants completed the same task using one of four treatments: generative ICSS (IdeaFisher), exploratory ICSS (Ideatree), conventional software (Harvard Graphics), and no software. The participants devised solutions to the homeless problem and were rated on a scale of one (no merit for the factor) to 10 (maximum merit for the factor) by experts in the task domain field. The results suggested that responses generated with software support are significantly more novel and valuable than responses generated by pen and paper.

It has been argued that humans are likely to fall into certain patterns that inhibit the generation of new insights [18]. TRIZ, a famous method of systematically eliciting innovative thinking, points out that inventions are actually combinations of ideas that already exist [1]. Duran-Novoa et al. [5] presented a strategy based on dialectical negation in which TRIZ and evolutionary algorithm (EA) approaches converge, creating a new conceptual framework for enhancing computer-aided problem solving. The two basic ideas presented are the inversion of conventional EA selection and the incorporation of new dialectical negation operators in evolutionary algorithms based on TRIZ principles. Two case studies are introduced to discuss the results of using a “Dialectical Negation Algorithm.”

6.4 Importance of Divergent Thinking

Felix Muller-Wienbergen et al. [12] introduced a design theory that describes the development of IT systems that enhance an individual’s creative performance by supporting divergent and convergent thinking in relation to knowledge provision. This research is grounded in different kernel theories and focuses on the simultaneous support of convergent and divergent thinking. It addresses all components of a design theory, presents research designs for an empirical evaluation, and a way for designers to propose creative design schemes with high efficiency and quality during the conceptual design phase.

Lopez-Ortega [7] proposed a holistic framework to describe creative people or to discover regions that are activated during creative endeavors and their interaction. To represent the interplay of cognitive processes around creativity, models were developed in the agent unified modeling language (AUML). Two modules of the resultant system are exemplified: opus planning and divergent exploration. The domain in which the software system is tested is the creation of musical pieces. Using the CSSs, mixed-gender groups generated ideas that were as novel and as creative as those of same-gender groups. innovation within organizations and, through facilitating effective management, enhanced the innovative capabilities of organizations and implementation in decision-making.

He also discussed how an IT supported environment can facilitate software innovation by allowing changes and adaptations late in the development process.

7 Conclusions

We have implemented a system that visualizes the tendency of humans to repeat the same patterns when asked to give a random sequence of numbers. Results showed that almost all participants repeated subsequences, some of which were surprisingly long. This demonstrates the strong human tendency to repeat the same patterns.

In our future work, we will compare our system with other tests that are said to measure creativity and see if there is any correlation. We plan to further explore this tendency in human thinking and also develop a way to overcome it and help humans become more creative.

We also plan to implement different interfaces and to test them so that the effects from the characteristics of our present interface can be avoided. For example, we are planning to build a system that makes users speak out numbers in a sequence and then converts them into a sequence of number symbols using automatic speech recognition. This will reduce the effect from uneven difficulty in performing different arm movements.

We have not shown that training by this system can raise the level of practical creativity. It is still unclear if unevenness in the appearance of number sequences is related to the unevenness of searching through combinatorial space in terms of practical creative behavior. To make the system more connected to the actual process of creative thinking, we would like to build a system that presents the sequence of objects or concepts belonging to a certain category, rather than using numbers. For example, we are considering building a system that asks chemists to list the names of reactions or molecules in random order, which may make them realize that they are likely to fall into a certain pattern, therefore restricting the generation of new combinations and potential inventions.

We would also like to explore the differences among individuals and if the ability to generate random patterns is actually relevant to the level of creativity the individual has. We plan to compare our system with existing tests that are said to measure creativity and see if there is any correlation. We want to see what factors affect scores in our system: for example, if the time of day would affect the level of creativity. Are individuals more likely to fall into the same pattern in the morning, or late at night? Also, we would like to compare scores when the subject is in different mental states: for example, when the subject is either elated or depressed, would the result be different?

Acknowledgments The work was supported in part by JSPS KAKENHI Grant Numbers 21700121, 25280110, and 25540159.

References

1. Altshuller, G.: Creativity as an Exact Science: The Theory of the Solution of Inventive Problems. Gordon and Breach Science Publishing, London (1984)
2. Amabile, T.M.: Creativity in Context: Update to the Social Psychology of Creativity. Westview Press, Boulder (1996)
3. Bostrom, R.P., Nagasundaram, M.: Research in Creativity and GSS, In: 31st Hawaii International Conference on System Sciences, pp. 391–405. IEEE Computer Society (1998)
4. Durand, D.E., VanHuss, S.H.: Creativity software and DSS: cautionary findings. Inf. Manag. **23**, 1–6 (1992)
5. Duran-Novoa, R., Leon-Rovira, N., Aguayo-Tellez, H., Said, D.: Inventive problem solving based on dialectical negation, using evolutionary algorithms and TRIZ heuristics. Comput. Ind. **62**, 437–445 (2011)
6. Guilford, J.P.: The Nature of Human Intelligence. McGraw-Hill Education, New York (1967)
7. Lopez-Ortega, O.: Computer-assisted creativity: emulation of cognitive processes on a multiagent system. Exp. Syst. Appl. **40**, 3459–3470 (2013)
8. Malaga, R.A.: The effect of stimulus modes and associative distance in individual creativity support systems. Decis. Support Syst. **29**, 125–141 (2000)
9. Manning, C.D., Schutze, H.: Foundations of Statistical Natural Language Processing. The MIT Press, Cambridge (1999)
10. Massetti, B.: An empirical examination of the value of creativity support systems on idea generation. MIS Q. **20**, 83–97 (1996)
11. Mednick, S.: The associative basis of the creative process. Psychol. Rev. **69**(3), 220 (1962)
12. Muller-Wienbergen, F., Muller, O., Seidel, S., Becker, J.: Leaving the beaten tracks in creative work—a design theory for systems that support convergent and divergent thinking. J. Assoc. Inf. Syst. **12**, 714–740 (2011)
13. Rhodes, M.: An analysis of creativity. The Phi Delta Kappan **42**, 305–310 (1961)
14. Russ, S.W., Fiorelli, J.A.: Developmental approaches to creativity. In: Kaufman, J.C., Sternberg, R.B. (eds.) The Cambridge Handbook of Creativity, pp. 233–249. Cambridge University Press, Cambridge (2010)
15. Suchamn, L.: Plans and Situated Actions: The Problem of Human-machine Communication. Cambridge University Press, Cambridge (1987)
16. Seidel, S., Muller-Wienbergen, F., Becker, J.: The concept of creativity in the information systems discipline: past, present, and prospects. Commun. Assoc. Inf. Syst. **27**, 14 (2010)
17. Sellen, A., Whittaker, S.: Beyond total capture: a constructive critique of lifelogging. Commun. ACM **53**(5), 70–77 (2010)
18. Storm, B.C., Angello, G.: Overcoming fixation: creative problem solving and retrieval-induced forgetting. Psychol. Sci. **21**(9), 1263–1265 (2010)
19. Tezuka, T., Yasumasa, S., Naghsh, F.A.: A system using n-grams for visualizing the human tendency to repeat the same patterns and the difficulty of divergent thinking. In: 8th International Conference on Knowledge, Information and Creativity Support Systems (2013)
20. Torrance, E.P.: The nature of creativity as manifest in its testing. In: Sternberg, R.J. (ed.) The Nature of Creativity: Contemporary Psychological Perspectives, pp. 43–75. Cambridge University Press, Cambridge (1988)

Knowcations—Positioning of a Meme and Cloud-Based Personal Second Generation Knowledge Management System

Ulrich Schmitt

Abstract The first generation of organizational knowledge management (OKM) focused on the capturing, storing, and reusing of existing knowledge. To be classed as second generation, systems need to facilitate the creation of new knowledge and innovation which requires creativity and the awareness that old knowledge becomes obsolete. Recent suggestions also urged to advance personal knowledge management (PKM) as an overdue support tool for knowledge workers in the rising creative class and knowledge societies. Based on the assumption of creative conversations between many individuals' PKM devices, the autonomous systems are supposed to enable the emergence of the distributed processes of collective extelligence and intelligence, which in turn feed them. With a PKM prototype system pursuing these qualities, the paper illustrates the interaction between a user and external information-bearing hosts and vehicles. The resulting feedback loop incorporates Boisot's I-Space Model, Dawkins' Memes, Probst's KM Building Blocks and Pirolli's Sensemaking Model for Intelligence Analysis and supports the Nanatsudaki (Seven Waterfall) Model.

Keywords Personal knowledge management systems · Information space · Extelligence · Memes · Business genes · Sensemaking · Creative class · Knowledge worker · Nanatsudaki (Seven Waterfall) model

1 First Versus Second-Generation Knowledge Management

In the "Complete Guide to Knowledge Management", Pasher and Ronen describe the focus of the first generation of knowledge management (KM) as the capturing, storing, and reusing of existing knowledge including "systems of managing

U. Schmitt (✉)
University of Stellenbosch Business School, PO Box 802, Gaborone, Botswana
e-mail: schmitt@knowcations.org

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_19

knowledge like company yellow pages, experts outlining processes they are involved in, creating learning communities where employees/customers share their knowledge, creating information systems for documenting and storing knowledge, and so on. These first-generation KM initiatives were about viewing knowledge as the foremost strategic asset, measuring it, capturing it, storing it, and protecting it. They were about treating knowledge as an asset, recognizing how it influences strategy, and wanting to make the most of it by managing it properly" [18].

The second generation of KM needs to focus, they argue, on creating new knowledge and innovation, a process which starts with the "reuse or new use of existing knowledge, adding an invention, and then creating a new product or service that exploits this invention." This process requires creativity and the awareness that old knowledge becomes obsolete. For reaping the appropriate rewards, it is essential to systematically exploit the knowledge captured and created [18].

In meme[1] theory, this innovative process is attributed to two factors: recombination and mutation. As Distin points out in "The Selfish Meme": "In recombination, existing memes are appropriately recombined in new situations, creating new ways of thought and novel effects, perhaps as the result of previously recessive memes' 'effects' being revealed in the reshuffle". As a result of mutations, "copies will not always be exact, and the idea or skill in question may change in some way en route". In science, for example, "existing theses are reshuffled perhaps in the light of new evidence—and this may lead to unforeseen consequences, or even to a fresh hypothesis" [10].

The distinction between the first and second KM generation is by no means trivial, in particular, since the lack of an established KM definition as well as the lack of an agreed scope of KM is blamed for the failure of too many knowledge management initiatives [11].

Hence, this paper takes account of how the concept of the system-in-progress accommodates the features of the first as well as the second KM generation. The second KM generation capabilities are closely related to the memetic context and the meme life cycle steps presented and referenced in the information-space model.

[1]Memes are (cognitive) information-structures that evolve over time through a Darwinian process of variation, selection, and transmission. They are able to self-replicate utilizing mental storage in human hosts and to influence their hosts' behavior to promote further replication. Memes are virtual, and have no intentions of their own, they are merely pieces of information in a feedback loop which are encoded in vehicles for transmission between human hosts; this loop facilitates their continued replication as mental copies with their longevity being determined by their environment [4, 7, 9].

2 Organizational Versus Personal Knowledge Management

In comparison to organizational KM (OKM), personal KM (PKM) has been positioned historically in a narrow individualistic confinement [6]. In limiting its scope, it has been labelled as sophisticated career and life management with a core focus on personal enquiry [19] or as a means to improve some skills or capabilities of individuals, negating its importance relating to group member performance, new technologies, or business processes [8].

Wiig's view differs and considerably amplifies its status: "The root objective of PKM is the desire to make citizens highly knowledgeable. They should function competently and effectively in their daily lives, as part of the workforce and as public citizens. In a society with broad personal competences, decision-making everywhere will maximize personal goals, provide effective public agencies and governance, make commerce and industry competitive, and ensure that personal and family decisions and actions will improve societal functions and Quality of Life. […] Competence of the individual is particularly important in small-to-medium-sized enterprises (SMEs) where variability of work requires workers to be more agile than in larger enterprises" [35].

Levy's vision puts PKM even at the center stage of knowledge management: "Without denying the importance of collective strategies and the shared visions that support them, I believe that social knowledge management should be thought of as an emergent level based on the creative conversation of many individuals' Personal Knowledge Management. One of the most important functions of teaching, from elementary school to the different levels of university, will therefore be to encourage in students the sustainable growth of autonomous capacities in PKM. And this personal management should be conceived from the outset as the elementary process that makes possible the emergence of the distributed processes of collective intelligence, which in turn feed it" [15].

Since Levy's notion has to enable continuous feedback loops between individuals and communities or institutions, PKM and OKM systems have to share compatible models and structures. In 'Practical Knowledge Management: A Model That Works' [22], Probst introduces eight building blocks of OKM[2] which form a logical sequence of activities to be carried out continuously. Since a Personal KMS has to facilitate the creation of new documents (e.g., thesis, article, report, presentation) with subsequent release to audiences, this model is also fully applicable

[2]Probst's eight building blocks of OKM: knowledge goals, knowledge identification, knowledge acquisition, knowledge development, knowledge distribution, knowledge use, knowledge preservation, and knowledge measurement [22].

to PKM (see paper Sect. 5), although the motivation, emphasis, scope, and scale of organizational KM systems differ.[3, 4]

Wiig's and Levy's views support the notion that "knowledge and skills of a knowledge worker are portable and mobile. Unlike manual workers, they have numerous options on where, how, and for whom they will put their knowledge to work. As Drucker so often noted, this is a fairly recent phenomenon, and many people have not yet come to grips with its implications for the present or the future." [23].

Both views are also at the heart of a concept and system-in-progress to aid teamwork, lifelong-learning, resourcefulness, and creativity of individuals throughout their academic and professional life and as contributors and beneficiaries of organizational performance.

3 Knowcations—A Personal KM System-in-Progress

The idea originated during the author's Ph.D. studies in the early 1990s and the resulting prototype has been continuously expanded to integrate additional uses and functionalities. Due to advances in standards-based rapid database development platforms as well as cloud/hosting services, the transformation into a marketable application across multiple platforms and environments has become a viable opportunity for capacity development and innovation. Recent papers published, accepted, or currently under review have concentrated on the underlying relevant concepts and features.

Attending to the general troubles of individuals managing their knowledge, four distinctive challenges at acquisition/preservation, collaborative, capacity development, and conceptual level (referred to in paper, Sect. 7) have been discussed [32]. It was shown that they can be constructively addressed by assimilating recognized models and approaches from other disciplines, e.g., Probst's eight building blocks of KM [22], Gratton's changing pattern of work [13], Andrew's four components of a strategy [1], and Kerwin's domains of ignorance [2, 25, 34]. Taking the ignorance matrix as a point of departure (given prominence by Rumsfeld's knowns and unknowns [24]), its personal learning cycles were analyzed by identifying seven wastes to be minimized with the assistance of the proposed PKM [28].

A further presentation and papers [29, 30] assessed the envisaged scope of the PKMS concept proposed by positively comparing it to a set of applied competences (the philosophical underpinning of a Higher Education Qualifications Framework

[3]"The goal of [organizational] KM is a practical one: To improve organizational capabilities through better use of the organization's individual and collective knowledge resources. These resources include skills, capabilities, experience, routines, and norms, as well as technologies." [22].

[4]Organizational KM's "overall objective is to maximize the enterprise's Intellectual-Capital-related effectiveness and returns in all its forms." [35].

[26]) as well as to Pirolli's and Card's Notional Model of the Sensemaking Loop for Intelligence Analysis [20].

To put Levy's notion into action, an integrated view of Boisot's 'Agent and the World' [4] and Stewart's and Cohen's 'Extelligence' [33] assisted in identifying seven present-day barriers which novel PKM approaches have to overcome by catering for five vital provisions[5] [27].

An earlier short paper/poster [30] also visualized the general knowledge formation process to be supported by a PKMS by presenting it within Boisot's I-Space Model [4]. This paper considerably advances this approach by positioning the PKM system itself and by demonstrating how its design elements are interacting with the user and the outside world and by adding the perspectives of the Nanatsudaki Model and Nonaka's Concept of 'Ba' to the initial paper [31].

4 The Visualization of a PKM System in Information-Space

The 'Information Space' or 'I-Space' Model [4] entails a three-dimensional matrix formed by the axes of codification, abstraction, and diffusion. Whereas the original model depicts the dynamic flow of knowledge assets following a 'Social Learning Cycle' through six phases, the matrix in Fig. 1 visualizes the PKMS's key constituents and their process pathways which consists of Pirolli's and Card's [21] foraging loop {steps 1–2} and sensemaking loop {steps 3–8} as well as added PKMS support functions {steps a–k}. The information-space has been divided into five layers:

1. The bottom first layer represents the part of the meme pool[6] which inhabits hosts as uncodified and tacit knowledge. A host must be able to possess at least the potential capacity to elaborate on a meme and to perform those cognitive tasks connected to the meme that we normally refer to as "understanding [12]."

[5]
1. Digital personal and personalized knowledge is always in possession and at the personal disposal of its owner or eligible co-worker, residing on personal hardware and/or personalized cloud-databases.
2. Contents are kept in a standardized, consistent, transparent, flexible, and secure format for easy retrieval, expansion, sharing, pooling, reuse and authoring, or migration.
3. Information and functionalities can continually be used without disruption and independent of changing one's social, educational, professional, or technological environment.
4. Collaboration capabilities have to be mutually beneficial to facilitate consolidated team and enterprise actions that convert individual into organizational performances.
5. The PKMS design and its complex operations are based on a concept, functionalities, and interventions which are clearly understood and are painlessly applied in practice.

[6] "Meme Pool: The full diversity of memes accessible to a culture or individual. Learning languages and traveling are methods of expanding one's meme pool" [12].

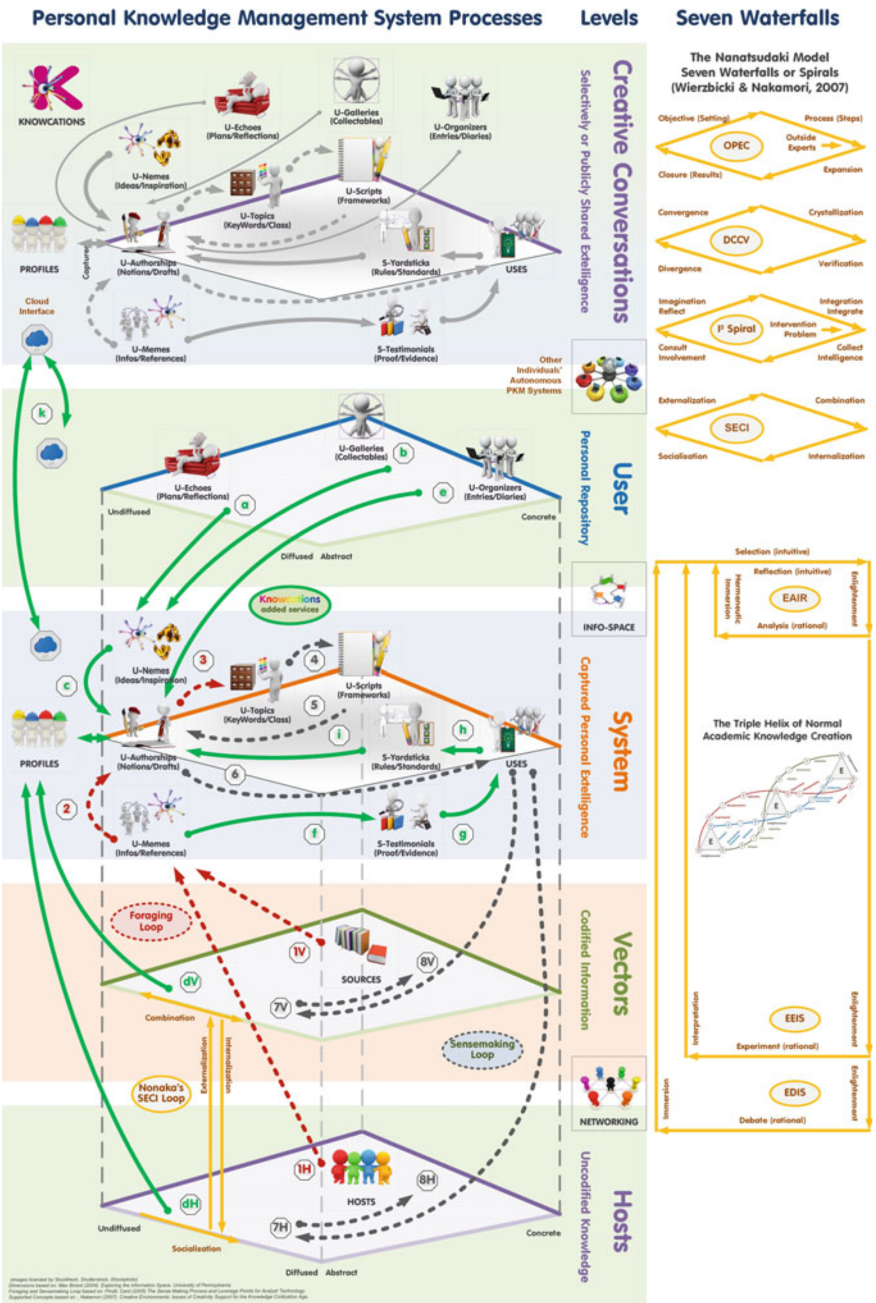


Fig. 1 Memes flow within PKM system depicted in I-Space [4, 16, 21, 36]

2. The second layer is occupied by memes and memeplexes[7] codified in vectors forming part of the private and shared world extelligence.[8] A vector is anything that transports the meme between hosts without the capacity to reflect on the meme, e.g., books, spoken message, observed behavior, CD, pictures, or artefacts [12].
3. The third layer characterizes the personal extelligence captured in the user's PKM system including the intellectual and social capital at his/her disposal.
4. The fourth layer depicts the user's individual intelligence and emotional capital.
5. The top layer symbolizes the combined shared knowledge bases of networked users. Three cloud interface symbols are shown on the left to indicate the links to other compatible autonomous PKM systems resulting in shared repositories, collaborative opportunities, and creative conversations (as further exemplified in Fig. 3).

Section 2 mentioned the need to bridge the gap between OKM and PKM by sharing compatible models and structures and Probst's eight building blocks [22] were cited as an example. These blocks correspond to the following steps (Fig. 1): knowledge identification {1}, acquisition {2}, preservation {3}, goals {4}, development {5}, distribution {6}, knowledge use {7 and 8}, and knowledge measurement {1–8}. Knowledge measurement takes place on a meta-level, when users and peers engage in feedback or when outstanding 'work to be carried out' (meme gaps to be filled) is evaluated against progress using dedicated comment fields.

4.1 The PKMS Foraging Support in the Information-Space

(For the ensuing deliberations and references to the icons in the Fig. 1 *{as indicated by square brackets} the following applies: in any link referred to between constituents of the individual's PKMS, either the creating or the receiving counterpart can be—in a shared and collaborative environment—substituted by the compatible constituent(s) of other cooperating autonomous PKMSs* {k}. *Furthermore, since the PKMS concept introduced incorporates Pirolli's Sensemaking Model for Intelligence Analysis* [20, 21] *the associated terms are co-used whenever applicable.)*

To succeed in becoming part of the body of socially transmitted information, a meme has to be encoded in an information-carrying medium and can then be passed

[7]Memeplexes are groups of memes mutually supporting each other and replicating together to benefit from competitive advantage [3].

[8]Stewart and Cohen introduced the term 'Extelligence' for externally stored information; it represents the cumulative archive of human cultural experience and know-how accessible and augmentable by any individual who knows how. In their concept, Extelligence forms the external counterpart to the intelligence of the human brain/mind and deals in information whereas intelligence deals in understanding; together they are driving each other in a complicit process of accelerating interactive co-evolution [33].

between peers from brain to brain via learning and imitation {Hosts & Vectors}. In the meme world, information 'consumes' the attention of its recipient/host and is either multiplied or lost in the process (but not consumed like physical goods).

A knowledge worker, referred to in the following as user (of the PKMS), foraging for data and information collects relevant material via field research {1H} or desk research {1V} from sources accessible to him/her and stores it in a case file or shoe box {U-Memes}. The outcome will resemble a disorganized pile of memeplexes, which are packaged or absorbed in complex ways inside larger vectors {Vectors} and recorded conversations or observations {Hosts}. The user is acting like any host with a limited attention span and memory, but with the potential cognitive capacity to single out and understand a meme, to elaborate on it, and to create groups of memes from diverse sources, mutually supporting each other for further replication. However, this activity does not only take place mentally but is supported by the PKMS functionalities and the more reliable storage and retrieval facilities of the underlying knowledge base and engine.

So, based on his/her interests, knowledge and editorial literacy, the user consciously selects suitable materials gathered, captures their memes' intended original 'messages' {2} and stores them in a knowledge base {U-Authorships}. To ease reusability any meme captured and codified should be—ideally—in an atomic state of an information-structure, perfectly understandable alone by itself but being able to be used at any later time in combination with other memes without piggybacking irrelevant or potentially redundant information. The latter requires the variation and replication of an original meme in a creative manner. As Koch reminds us, not the physical script matters but the ideas it contains; "It must be valued, either for its own intrinsic appeal or because it can help to deliver other things that people want, or help to deliver them at a higher quality level or using fewer resources" [14]. In an iterative process, emerging gaps in an authoring project mean that further suitable memes to be found or already captured have to be added or, alternatively, new memes have to be self-authored or re-purposed from personal repositories {a, b, c}. Depending on the required context, the contents, purpose, and format (text, sound, visual, message) of memes might have to be adjusted, resulting in copying and variation.

The references of memes (e.g., origins, authors, publishers, contact details, titles, formats, licenses) are also stored {dH, dV} in the knowledge base {Profiles} by linking meme-relevant hosts (individuals, teams, communities, and organizations as actors in research/project-related, industrial, service-oriented, and geographic settings) to the meme-accommodating vectors (knowledge sources and uses). The capturing of host-to-hosts/sources/uses relationships assures the comprehensive recording of intellectual capital, but also provides suitable 'scaffoldings' for social (bullets 1 and 2) and emotional capital, in particular, regarding reflection and confidence building (bullets 4 and 5). These assets are key targets of an organizational KMS, but equally relevant to a personal KMS.

1. Structural Relations and Networks Bases depict interdependencies of an actor in regard to any other actor (e.g., memberships, relatives, superior, or advisor).

2. Formal Education and Professional Experience Bases capture past and current organizational ties (e.g., employers or universities) typically chronologically ordered in a CV, supported usually by qualifications and references.
3. Exploration Base lists the access to existing Sources and the particular role (e.g., author, publisher, or owner); Innovation Base references the user's occupation with Uses which includes work-in-progress and planned activities.
4. Research Activities and Leadership/Achievements Bases describe past and current ties with a more task/project-specific orientation, evidenced usually by publications, project proposals, or reports as well as peer assessments.
5. Outcomes/Results Bases consider the wider benefits of activities for society (e.g., Environment or Development Agenda) evidenced by impact studies/testimonials.

Supporting evidence of any relationships is stored separately {f – S-Testimonials – g} and attached to the user's output {Uses} at the end for dissemination to back up any statements or claims made.

Codified information to any of the entities (e.g., directories, lists of ingredients) can be uploaded and integrated for convenience. Current datasets used include: world's countries, regions, and cities; international universities; ERA journals and conferences; standard research as well as industrial classification codes; higher education standards and audit criteria; food, cocktail, and genealogical datasets.

4.2 The PKMS Sensemaking Support and Knowledge Diffusion

In order to facilitate future recovery, replication, and repurposing, the searchability of the memes is assured via the references to the source of origin. To further enhance their accessibility, the user qualifies a meme by linking it to a multidimensional classification system, made up of pre- or user-defined abstract Meme Types (e.g., area, concept, process, tool) or already created topics and memeplexes (e.g., decision methods, logistics, ecology) {3} to be stored in a Topics and Schema Base {U-Topics}. Thus, memes and/or sub-topics are only a few clicks away and can be more easily and speedily retrieved and repurposed. Because any entity with a focus is displayed on a user's screen with its immediate entity neighborhood, the more qualified memes stocked impart constant mental refreshers about their usage and are better memorized.

During the authoring process, the accumulated meme pool is scanned to activate appropriate 'candidates' for composing a desired Script (e.g., article, lecture, or presentation) {4} to be stored in a Scripts and Hypotheses Base {U-Scripts}. Any gaps will be filled with a provisional 'known-unknown' meme as a reminder for work to be carried out in order to direct the iterative processes alluded to {5}. Additionally, dedicated fields allow for comments and to-do-lists regarding any entity in the knowledge base, and silent memes linked to particular memeplexes can be added which do not show up during publication, but contain, for example, annotations, further ideas, feedbacks from colleagues or supervisors. The significance of recombination and mutation, as emphasized by Distin, [10] has already been pointed out in Sect. 1.

Any finalized script can be converted into a presentation, pdf or paper version for publication and wider diffusion {6} and stored in a Report or Presentation Base {Uses} in order to become part of the shared world extelligence {Hosts &Vectors}.

Any finalized script can also be left in the knowledge base and shared with other compatible external PKMS. It provides a knowledge asset with all reference links kept intact and instant access to the underlying information-rich contributing memes. This type of digital document—if the content is appropriate for the purpose—is ideal for storage in a Benchmarking and Standards Base {S-Yardsticks} whose contents is feeding back {h & i} to related later projects and activities by providing, for example, templates, samples, best-practice methods, proven heuristics, regulations, tutorials, evaluation criteria, or trial assessments. Unfortunately, in a pdf or paper copy these information-rich causative references have to be given up due to its one-dimensionality.

After publication and diffusion, the novel insights are shared by oral presentation from person-to-person {7H} or via newly codified vectors {7V} and lead to shared new learning experiences and behaviors. As a consequence, knowledge and memes are absorbed by the human brain and can become 'intangible', 'uncodified', or 'tacit' knowledge {Hosts} as well as personal or organizational extelligence {Vectors}. Eventually, the abstract knowledge absorbed makes an impact by becoming embedded in concrete practices, either in codified formats such as documents or products {8V} or uncodified formats such as unwritten rules or patterns of behavior {8H}.

5 Human Capital Development—The Impact of PKM Solutions

To build personal capacities for career progress in the propagated Knowledge Economies and their increasingly complex multi-disciplinary problem spaces, PKM systems for knowledge workers and their diverse roles and career portfolios are critical. Existing solutions address PKM needs only partially; they concentrate on more specialized as well as wider unrelated tasks and, accordingly, are usually

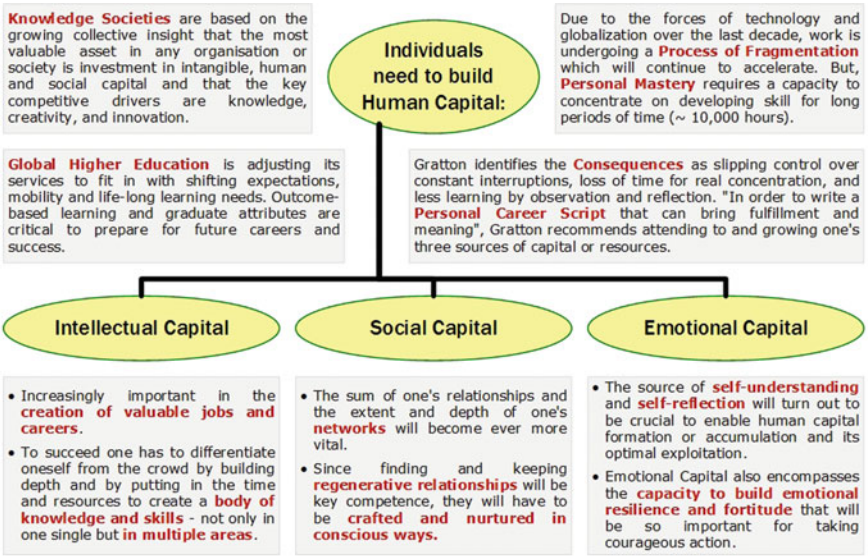


Fig. 2 Three sources of human capital or resources [13]

grouped into categories such as Office Suites, Document and Bibliographic Management, Contact and Relationship Management, Group and Collaboration Software, Web Databases, and the Organizational Knowledge Management Systems referred to.

With dedicated PKM systems, stacks of time and attention currently lost due to redundant findings, mundane tasks, and rework could to be mobilized for concentrating—instead—on the creative or innovative targets set and for facilitating consolidated actions that convert individual into organizational performances. Novel PKM approaches catering for the five provisions[5] made are geared toward continuous life cycle support from trainee, student, novice, or mentee to professional, expert, coach, or leader. They empower individuals to keep their accumulated intellectual, social, and emotional capitals (Fig. 2) in compatible recorded formats to benefit continuous modes of maintenance and learning and to uphold a basis for sustainable development, innovative authorship, and mutually beneficial collaborations (Fig. 3). They also imply a departure from the current heavyweight, prohibitive, centralized, top-down, institutional developments with preference given to grass roots, bottom-up, lightweight, affordable, and personal applications across a multitude of information and communication platforms.

In “Creative Environments” [36], Wierzbicki and Nakamori affirm: “In the new knowledge civilization era, given the systemic methods and tools of intercultural and interdisciplinary integration of knowledge, we shall also need computerized creativity support”, but they also maintain: “Before we think of constructing systems of tools to support creativity, we should reflect upon what prescriptive conclusions can be derived from descriptive theoretical considerations.”

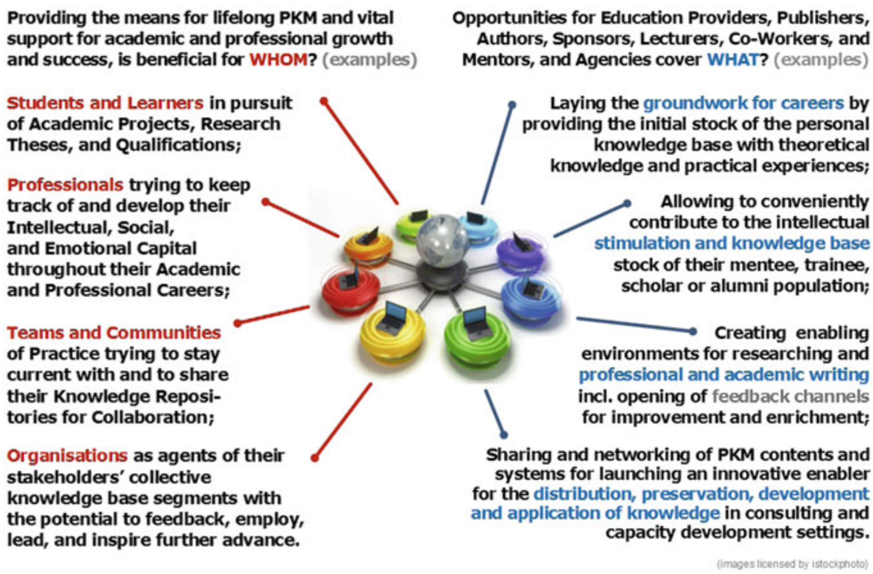


Fig. 3 PKMS conversation clusters exemplifying beneficiaries and benefits [28]

The system-in-progress introduced agrees with this notion; it provides a pragmatic novel solution based on established concepts and theories, some of which have already been alluded to.

The knowledge diffusion process is vital for today's knowledge economies and societies and has attracted widespread academic and professional attention. Hence, the features of the system can be further verified against the prescriptive notions of the JAIST Nanatsudaki Model with its sequence of seven waterfalls or seven creative spirals (depicted in Fig. 1 on the right) which describe academic/individually-oriented as well as practice/group-oriented knowledge creation processes [36].

Three spirals form a Triple Helix of normal academic knowledge creation. First, the Implementation or EEIS Spiral guides individual research by the {User} interacting with {Vectors} for desktop research or with {Hosts} for field research. Second, the Debate or EDIS Spiral stands for the questioning of assumptions and debating of findings resulting from the {User} interacting with other {Hosts}. Third, the Hermeneutics or EAIR Spiral correlates with the {User} interacting with his/her PKM {System} assuming the relevant information and knowledge gathered from outside sources have been appropriately captured during the activities of the two first spirals.

Four further spirals are practice and group oriented and, thus, represent the interaction of the {User} with other individuals' autonomous but networked PKM devices allowing for partially or publicly shared extelligence and {Creative Conversations}. They resemble the Objectives or OPEC Spiral for setting objectives and targets, the Brainstorming/writing or DCCV Spiral for generating ideas, and the Roadmapping or I^5-Spiral covering detailed process planning and controlling [36].

Special consideration needs to be given to the Socialization Spiral equivalent to Nonaka's SECI Loop Model [16] which addresses a major objective of organizational KM. Its aim is to make the tacit knowledge (gained only experientially and difficult to articulate, explain, share; as opposed to formal or explicit knowledge) of knowledge workers explicit, so it can be measured, captured, stored, protected, and further utilized in a 'spiral' of knowledge creation for the benefit of the organization and its stakeholders and independent of the availability of the persons concerned.

Its four knowledge conversion modes are also shown in the {Vectors} and {Hosts} layers (Fig. 1): Socialization {tacit-to-tacit: Hosts, undiffused to diffused; 1H sharing, 8H}, Externalization {tacit-to-explicit: Hosts to Vectors; 1H recording}, Combination {explicit-to-explicit: within Vectors and System; all steps with the exception of 1H, 7H, and 8H}, and Internalization {explicit-to-tacit: Vectors to Hosts; 7H}.

All seven spirals are, as described, supported by the PKM system-in-work. All entities and relationships share a common data structure which adds to the system's flexibility as memes pass through the learning cycles. The meme flow represented within the I-Space also integrates Nonaka's concept of 'ba' [17] including its phases and spaces for originating and socialization, exercising and internalization, interacting and externalization, and the Cyber Ba guiding the combination phase.

6 Conclusions and the Road Ahead

This applied research paper has introduced a novel approach to Personal Knowledge Management supported by a prototype system which responds to the operational or needs analyses of Dawkins, Destin, Koch, Stewart, Cohen, Gratton, Pasher, and Ronen as well as integrates the prescriptive notions or design recommendations of Armour, Schamanek, Boisot, Wiig, Levy, Probst, Pirolli, Russell, Card, Nonaka, Takeuchi, Noboru, Wierzbicki, and Nakamura. In consequence, it corresponds closely to Bush's, so far, unfulfilled 69-year-old vision of the 'Memex' [5].

It is planned to transform the Prototype to a commercially viable PKM application within 2 years. In parallel, a training seminar will be set up as well as a curriculum and study guide for the wider multi-disciplinary PKM contexts and methodologies.

Further papers in progress or planned will feature technical aspects, case studies, and demonstrations. The latter will document the system's knowledge management and authoring capabilities based on the memes and references of prior publications to result in a paper detailing the steps of its production including relevant screenshots.

References

1. Andrews, K.R.: The Concept of Corporate Strategy. Irwin, University of Michigan (1987)
2. Armour, P. G.: The five orders of ignorance. Commun. ACM **43**(10), 17–20 (2000)
3. Bjarneskans, H., Grønnevik, B., Sandberg, A.: The lifecycle of memes. http://www.aleph.se/Trans/Cultural/Memetics/memecycle.html (1999). Accessed 29 Sept 2014
4. Boisot, M.: Exploring the Information Space, pp. 5–8. University of Pennsylvania (2004)
5. Bush, V.: As we may think. Atlantic Mon. Issue **176**(1), 101–108 (1945)
6. Cheong, R.K., Tsui, E.: From skills and competencies to outcome-based collaborative work. Knowl. Process Manage. **18**(3), 175–193 (2011)
7. Collis, J.: Introducing Memetics. http://meme.sourceforge.net/docs/memetics.php (2003). Accessed 11 Feb 2003
8. Davenport, T.H.: Personal knowledge management and knowledge worker capabilities. In: Pauleen, D.J., Gorman, G.E. (eds.) Personal Knowledge Management, pp. 167–188 (2011)
9. Dawkins, R.: The Selfish Gene. Paw Prints (1976)
10. Distin, K.: The Selfish Meme. Cambridge University Press (2005)
11. Frost, A.: Two reasons why knowledge management fails. YouTubeVideo. http://www.youtube.com/watch?v=pF9M5ezkxQs (2011). Accessed 29 Sept 2014
12. Grant, G.: Memetic Lexicon. http://www.aleph.se/Trans/Cultural/Memetics/meme_lex.html (1990). Accessed 29 Sept 2014
13. Gratton, L.: The Shift—The Future of Work is Already Here. Harper Collins (2011)
14. Koch, R.: The Power Laws of Business. Nicholas Brealey (2001)
15. Levy, P.: The Semantic Sphere 1. Wiley (2011)
16. Nonaka, I., Takeuchi, H.: The Knowledge-Creating Company. Oxford University Press (1995)
17. Nonaka, I., Noboru, K.: The concept of "ba": Building a foundation for knowledge creation. Calif. Manag. Rev. **40**(3), 40–55 (1998)
18. Pasher, E., Ronen, T.: The Complete Guide to KM. Wiley (2011)
19. Pauleen, D.J., Gorman, G.E.: The nature and value of personal knowledge management. In: Pauleen, D.J., Gorman, G.E. (eds.) Personal Knowledge Management, pp. 1–16 (2011)
20. Pirolli, P., Russell, D.M.: Introduction to this special issue on sensemaking. Human-Comput. Interact. **26**(1–2), 1–8 (2011)
21. Pirolli, P., Card, S.: The sensemaking process and leverage points for analyst technology. In: Proceedings of International Conference on Intelligence Analysis, vol. 5 (2005)
22. Probst, G.J.B.: Practical Knowledge Management. Prism, A.D. Little, pp. 17–29 (1998)
23. Rosenstein, B.: Living in More Than One World. Berrett-Koehler Publishers (2009)
24. Rumsfeld, D.H.: Knowns & Unknowns. http://www.youtube.com/watch?v=QaxqUDd4fiw (2002). Accessed 29 Sept 2014
25. Schamanek, A.: Taxonomies of the Unknown. http://andreas.schamanek.net/w/phi/not-knowing/taxonomies_of_the_unknown (2012). Accessed 29 Sept 2014
26. SAQA: Level Descriptors for the South African National Qualification Framework (NQF). South African Qualification Authority (2012)
27. Schmitt, U.: Overcoming the seven barriers to innovating personal knowledge management systems. In: International Forum on Knowledge Asset Dynamics (IFKAD), 11–13 June 2014, Matera, Italy, pp. 3662–3681 (2014)
28. Schmitt, U.: Making sense of e-Skills at the dawn of a new personal knowledge management Paradigm. In: Proceedings of the e-Skills for Knowledge Production and Innovation Conference, 17–21 Nov 2014. Cape Town, South Africa (2014)
29. Schmitt, U.: Making personal knowledge management part and parcel of higher education programme and services portfolios. J. World Univ. Forum **6**(4), 87–103 (2014)
30. Schmitt, U.: Knowcations—a meme-based PKM system-in-progress. In: 8th International Conference on E-Learning (ICEL), 27–28 June 2013, pp. 523–527. Cape Town, South Africa (2013)

31. Schmitt, U.: Knowcations—conceptualizing a meme and cloud-based personal second generation knowledge management system. In: Skulimowski, A.M.J. (ed.) Looking into the Future of Creativity and Decision Support Systems. Proceedings of the 8th Conference on Knowledge, Information and Creativity Support Systems (KICSS), 07–09 Nov 2013, Krakow, pp. 587–598, Poland (2013)
32. Schmitt, U.: Knowcations—the quest for a personal knowledge management solution. In: i-Know, 12th International Conference on Knowledge Management and Knowledge Technologies, 05–07 Sept 2012, Graz, Austria (2012)
33. Stewart, I., Cohen, J.: Figments of Reality. Cambridge University Press (1999)
34. University of Arizona Health Sciences Center: Q-cubed Programs—What Is Ignorance. http://www.ignorance.medicine.arizona.edu (2012). Accessed 29 Sept 2014
35. Wiig, K.M.: The importance of personal knowledge management in the knowledge society. In: Pauleen, D.J., Gorman, G.E. (eds.) Personal Knowledge Management, pp. 229–262 (2011)
36. Wierzbicki, A.P., Nakamori, Y.: Creative Environments: Issues of Creativity Support for the Knowledge Civilization Age. Springer Publishing Company (2007)

Comparing Different Implementations of the Same Fitness Criteria in an Evolutionary Algorithm for the Design of Shapes

Andrés Gómez de Silva Garza

Abstract Evolutionary algorithms (EAs) have been used in varying ways for design and other creative tasks. One of the main elements of these algorithms is the fitness function used by the algorithm to evaluate the quality of the potential solutions it proposes. The fitness function ultimately represents domain knowledge that serves to bias, constrain, and guide the algorithm's search for an acceptable solution. In this paper, we explore the degree to which the fitness function's implementation affects the search process in an evolutionary algorithm. To perform this, the reliability and speed of the algorithm, as well as the quality of the designs produced by it, are measured for different fitness function implementations. These measurements are then compared and contrasted.

Keywords Evolutionary algorithms · Fitness function · Evolutionary design

1 Introduction

Evolutionary algorithms (EAs) are general-purpose search methods that proceed by performing operations on the individuals in a population of potential solutions to a problem [5]. An EA will evolve these potential solutions through a series of generations until some convergence criterion is reached. EAs embody such a general-purpose search method that it is no surprise that EA's have often been used in supporting design and other creative tasks. Two compendia of example systems and applications are [1] and [2]. In many of the examples described in these books, the fitness function is not automated. Instead, in some of the example approaches a user is asked to determine the fitness function by tweaking the values of different

A.G. de Silva Garza (✉)
Computer Engineering Department, Instituto Tecnológico Autónomo de México (ITAM), Río Hondo #1, Col. Progreso-Tizapán, 01080 Mexico D.F., Mexico
e-mail: agomez@itam.mx

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_20

parameters provided through system interface controls. In some of the other examples, votes are gathered from a large set of users to rank the solutions proposed by the systems, thus deciding which solutions will survive and eliminating the need for a fitness function to be explicitly programmed. Instead, in our work we are concerned with the traditional EA approach of having a completely preprogrammed, autonomous, fitness function integrated into the EA's functioning.

The fitness function analyzes the characteristics of the solutions generated by the EA according to how well they match the desired characteristics, which are usually specified when the EA search is initiated. These desired characteristics are usually expressed in general terms, and the fitness of a particular solution is determined by seeing the degree to which the specific solution matches with the general pattern. Like any computational algorithm, the fitness evaluation function can be implemented in multiple ways. All of them are equivalent in the sense of describing the same general type of algorithmic solution. However, the specific way in which the fitness function is expressed or implemented could affect the performance of the EA. This paper describes a set of experiments designed to measure this influence. It is assumed that the reader has a passing knowledge of how a generic EA operates. If this is not the case, the reader is referred to [4], which describes the characteristics of EAs.

The EA used in the experiments described in this paper is called ShEvolver (a contraction of "shape evolver"). Here, we describe ShEvolver briefly to set the context for the description of the experiments we report in this paper. A more complete description of ShEvolver can be found in [3], from which a few sections of this paper were taken or adapted (with the permission of the Association for Computing Machinery, ACM, the copyright holder). The domain of ShEvolver is the evolution of shapes that consist of configurations of colored unit squares. This kind of domain and algorithm can be applied, for example, to the design of bathroom/kitchen tile patterns.

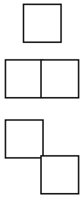
The genetic alphabet used by ShEvolver is such that each gene represents the execution of a "move-and-place" action, moving from an initial location, by a distance of one unit square, to a destination location in any one of eight possible directions, and placing a colored unit square there (in one of four possible colors). A genotype is a sequence of such moves that, when followed in succession, end up producing a configuration of colored unit squares on the "canvas" (initially empty) that the system is working on.

2 Fitness Evaluation

There are 36 fitness evaluation functions (criteria) that have been implemented in ShEvolver, labeled *c1–c36*. Some of them focus on measuring geometric characteristics of shapes, some of them focus on color-related features, and others combine aspects of both. An example of the first type of function is *c4*, which measures the bumpiness of a shape (where bumpiness is taken to mean that the outer edges

Fig. 1 The method for calculating the bumpiness of a shape (evaluation criterion *c4*) and three examples

```
// Calculates the degree of bumpiness
// of a shape s:
public double c4(Shape s) {
    return (double) numberOfAngles(s)/
           (double) perimeter(s);
}
```



(a) Bumpiness: 4/4=1

(b) Bumpiness: 4/6=1/3=0.67

(c) Bumpiness: 8/8=1

of a shape are jagged, caused by diagonal adjacencies between the unit squares forming the edges, rather than smooth, caused by horizontal or vertical adjacencies between the unit squares forming the edges). An example of the second type of function is *c12*, which measures the greenness of shapes (the percentage of unit squares in a shape that have a green color). An example of the third type of function is *c17*, which measures the degree to which x-shaped sub-shapes of a given color occur within a shape.

As an example, Fig. 1 shows the Java code for the method used to measure the bumpiness of a shape (evaluation function *c4*) and shows how it applies to three small shapes, a, b, and c. In all three cases, a fitness value of 1 is achieved only if the shape has the maximum possible bumpiness given the number of unit squares it is composed of. Otherwise, a fitness value between 0 and 1 is assigned to the shape reflecting its degree of bumpiness. Specifically, the degree of bumpiness is calculated by counting the number of angles along the edges of a shape and dividing by the perimeter of the shape.

Each of the evaluation functions in ShEvolver produces a fitness value between 0 (meaning the absence of whatever is being measured) and 1 (meaning the maximum possible amount of whatever is being measured). ShEvolver is designed so that many of these fitness evaluation functions can be applied in parallel to the shapes produced by the system before assigning a global fitness value. When the user decides to use multiple evaluation functions, each one of them is given the same weight/importance, i.e., the global fitness function consists of a linear combination of individual evaluation functions.

As an example, let us assume that in a given scenario the fitness function consists of a linear combination of two evaluation criteria, *c1* and *c2*. Let us also assume that each of these evaluation criteria was used to evaluate a shape *s*. Thus, if *c1*(*s*) is the fitness value of shape *s* when evaluating it using *c1*, and *c2*(*s*) is its fitness value when evaluating it using *c2*, then the global, normalized, fitness value *F* for *s* would be calculated as

$$F(s) = (c1(s) + c2(s))/2.$$

If three evaluation criteria (*c1*, *c2*, and *c3*) had been used to evaluate *s*, then the global, normalized, fitness value *F* for *s* would be calculated as

$$F(s) = (c1(s) + c2(s) + c3(s))/3.$$

And so on. The result is always a value of *F* that falls between 0 and 1, inclusive, calculated in such a way that each of the individual evaluation criteria that make up *F* is given the same importance as the rest. ShEvolver is currently configured so that its EA tries to maximize the global fitness value of the shapes it generates, though it could easily be adapted so that its EA tried to minimize the value. If a shape ends up having a global fitness value of 0.75, this indicates that the shape has a quality of 75 % according to whatever evaluation criteria were being used to measure said quality.

Given this basic framework, two versions of ShEvolver were produced, A and B. In ShEvolver A, each of the individual fitness evaluation functions was enhanced by incorporating into it a verification procedure that checks that the shape meets some minimum and maximum width and height requirements. The enhanced version of the method shown in Fig. 1 is thus the one shown in Fig. 2. The value returned by the method is identical to the value the original version would return, unless the constraints on the dimensions of the shape are not met, in which case the method returns a fitness value of 0. In the experiments described below, when running ShEvolver A only one evaluation function was used at any given time (the enhanced version of one of *c1*–*c36*) to assign a fitness value to each shape produced by the EA.

In ShEvolver B, the fitness evaluation functions were left without modification, but two new functions, *c37* and *c38*, were written and added. In *c37* a 1 or 0 is assigned to a shape depending on whether its dimensions fall within a minimum and a maximum acceptable width or not, respectively. No values between 0 and 1 are allowed. In *c38*, an analogous procedure is followed, but this time analyzing the height instead of the width. When running ShEvolver B in the experiments

Fig. 2 The modified version in ShEvolver A of the method from Fig. 1

```
// Calculates the degree of bumpiness
// of a shape s:
public double c4(Shape s) {
    int w=getWidth(s), h=getHeight(s);
    double result;
    result=(double) numberOfAngles(s)/
           (double) perimeter(s);
    if(w<MIN_WIDTH||w>MAX_WIDTH||
       h<MIN_HEIGHT||h>MAX_HEIGHT)
           result=0;
    return result;
}
```

described below, three evaluation functions were used, and then combined linearly as described above, to assign a fitness value to each shape produced by the EA: one of *c1–c36*, plus *c37* and *c38*.

From the above descriptions it is apparent that ShEvolver A and ShEvolver B are not very different from each other. Because one (enhanced) fitness function was used at a time when running ShEvolver A, satisfying the constraints on the dimensions of a shape contributes 50 % (25 % for width and 25 % for height) of the global fitness value of that shape (the other 50 % being due to whichever original evaluation criterion was used at a given moment). In ShEvolver B satisfying the constraints on the dimensions of a shape contributes 66.7 % of the global fitness value of that shape (33.3 % for width and 33.3 % for height), with the final 33.3 % being due to whichever original evaluation criterion was used at a given moment.

3 Experimental Design and Results

In this section, we describe an experiment that was designed to compare the individual evaluation functions in ShEvolver. The experiment was repeated for each of two versions, one using ShEvolver A and one using ShEvolver B. Each version was run through two stages.

Stage 1 consisted of several runs. In each run the initial population of the EA was generated at random. In each run the convergence criterion used was a combination of two factors: the EA would stop as soon as it produced the first individual with a fitness value of 0.95 (convergence due to achieving a minimum acceptable quality) or as soon as it had reached 5000 evolutionary cycles (convergence due to reaching a maximum acceptable time limit), whichever came first. The limit of 5000 evolutionary generations was designed to halt the EA when no individual with a fitness of at least 95 % was found within a reasonable amount of time, instead of permitting the EA to continue potentially forever, with the possibility of never converging. All other EA parameters such as population size, crossover and mutation rates, etc., also remained fixed across all runs. The minimum and maximum requirements were fixed across all runs and were set at 5 for both width and height (that is, the EA was being forced into producing shapes measuring exactly 5×5 unit squares, in addition to satisfying whatever other evaluation criterion was active—see below).

In Stage 1 of the experiment, we performed 50 runs of ShEvolver using each of the 36 individual evaluation functions one at a time for a total of $50 \times 36 = 1800$ runs. The variables that were measured for each run were the number of generations (G) that the EA went through before convergence and the average fitness (AF) of the individuals in the population at the time of convergence. The best design produced at the end of each run was also obtained.

The reason for performing 50 runs for each experimental condition was to be able to filter out any unusual results that may have been due to the random nature of many of the decision points in an EA, such as the initial makeup of its population or the selection of crossover points. Instead, by obtaining the mean behavior over 50 runs, general trends in the overall set of results can be observed irrespective of any quirks that may have been present in any of the individual runs. For each set of 50 runs corresponding to each evaluation function, the general trends that were focused on were the mean values for G and AF, as well as the reliability of the EA. The reliability is interpreted as being the percentage of times that convergence occurred because a shape of sufficient quality had been produced (quality convergence), as opposed to reaching a maximum allowable number of EA generations (time convergence). Given that the experiment consists of comparing different fitness functions, obtaining different values for the reliability of an EA for the different sets of 50 runs can also be used to measure the relative strictness of the fitness functions. Fitness functions that are easier to satisfy, for whatever reason, will result in higher reliability values than fitness functions for which it is more difficult to produce high-quality shapes.

The observations listed below as a result from the experiments that were run focus on evaluation functions *c4*, *c12*, and *c17*, although the experiments performed took into account all evaluation functions, as mentioned above. Numerical result tables can be consulted in [4]. Here, we include only some observations stemming from the study of said numerical results.

With respect to EA reliability, the intuition that the constraints imposed by some evaluation functions are easier to satisfy than others was demonstrated by the fact that *c12* had a reliability of 1 in both ShEvolver A and ShEvolver B (very easy to satisfy), *c17* had a reliability of 0 in both ShEvolver A and ShEvolver B (very difficult to satisfy), and *c4* had a reliability of 0.96 in ShEvolver A but 1 in ShEvolver B (somewhat easy to satisfy).

With respect to EA convergence, the relative results for ShEvolver A compared to ShEvolver B seem to indicate that, when convergence is in general due to quality (i.e., under high-reliability conditions), ShEvolver B converges much quicker than ShEvolver A (272.42/7.9 ≈ 34 times quicker for *c4* and 186.96/53.08 ≈ 3.5 times quicker for *c12*), whereas under low reliability conditions (*c17*) there does not seem to be any difference between ShEvolver A and ShEvolver B as far as an impact on when convergence occurs. Analyzing the results for ShEvolver A seems to confirm the observation listed in the previous bullet point that *c12* is an evaluation function that is very easy to satisfy, *c17* is very difficult to satisfy, and *c4* is somewhat easy to satisfy. However, in the results for ShEvolver B it would seem that *c4* is easier to satisfy than *c12*. Thus, even tiny differences in comparative experimental scenarios (ShEvolver A vs. ShEvolver B) can cause sets of results that lead to different interpretations.

With respect to mean fitness, the mean fitness of the population for *c17* in ShEvolver A, 0.002, shows that there are at least a few individuals in the population with nonzero fitness values, even though none of them were good enough to make the EA converge due to quality in any of the runs (as seen from the 5000 mean value for convergence of *c17*). The results for ShEvolver A make it difficult to distinguish *c4* from *c12* (the mean population fitness values that the two evaluation functions produced were very similar: 0.896 and 0.900, respectively). However, if we analyze the results for ShEvolver B we could conclude that *c4* is much more difficult to satisfy than *c12* (since the mean fitness of the individuals in the population when convergence occurred was only 0.638 for *c4* but 0.922 for *c12*), and in fact that *c4* is slightly more difficult to satisfy than *c17* (which produced a mean fitness value of 0.681), in direct contradiction with all other indications that *c17* is very difficult to satisfy (both in absolute terms and relative to the other evaluation functions analyzed).

In Stage 2 of the experiment each of the 1800 best designs produced in each of the runs in Stage 1 was evaluated using each of the 36 individual fitness functions in ShEvolver. The purpose of this stage was to get a better feel for how the evaluation functions compare when used to analyze a set of shapes of which most were not "attuned" to the functions. Only 50/1800 = 2.8 % of the shapes that were thus evaluated were produced by an EA using the fitness function that was doing the evaluation. Graphical results of these experiments can be consulted in [4]. Here, we include only some observations stemming from studying said graphical results.

When we compare the results of applying *c1–c36* to 1800 designs using ShEvolver A with the application of the same fitness functions to the same designs using ShEvolver B, the overall behavior is the same, as far as which evaluation functions can be interpreted to be very easy to satisfy, very difficult to satisfy, and somewhat easy to satisfy. This lends further credence to the conclusions drawn earlier with respect to classifying *c4*, *c12*, and *c17* into these three categories. However, the fitness value for each design in the entire set of 1800 is clearly higher (approximately twice as high) when using ShEvolver B compared to its value when using ShEvolver A, for every evaluation function. This shows very clearly that even a slight difference in the way that global fitness evaluation is implemented can cause a great difference in the performance of an EA. We saw it before, in Stage 1, when measuring various EA search parameters such as reliability and speed of convergence, and we see it again here, in Stage 2, when measuring the quality of the designs produced by the EA.

In fact, we actually created a third variation of the system, ShEvolver C. In ShEvolver C we combined the functionality of *c37* and *c38* from ShEvolver B into one new fitness evaluation function, *c39* (which assigns a fitness value of 1 to a shape only if it satisfies minimum and maximum width and height requirements at the same time). We then ran the same experiment as described above using ShEvolver C, always using as a global fitness function a linear combination of two evaluation functions: one of *c1–c36*, plus *c39*. In theory the results should have been similar to those for ShEvolver A, because now we have a scenario in which satisfying the constraints on the dimensions of a shape contributes 50 % (25 % for width and 25 % for height) of the global fitness value (*c39*) of that shape (the other 50 % being due to whichever original evaluation criterion is being used at a given moment), as in ShEvolver A. However, the result was that the experiment could not be completed, as the computer is always out of heap space (memory) while running it, even though several attempts were made. It was decided to try again, but setting the time convergence criterion such that the EA went on for a maximum of 500 generations instead of 5000, assuming that this would use up less memory, but the results were the same. For some reason in ShEvolver C the nature of the shapes being created by the EA was such that they always grew in size, instead of being constrained to the 5 × 5 dimensions imposed by the size-related fitness criteria, whereas in ShEvolver A and ShEvolver B the size constraints were being met by the shapes and thus the system never ran out of memory.

4 Summary and Discussion

In this paper we described ShEvolver, an EA used for evolving shapes that consist of two-dimensional configurations of colored unit squares. In ShEvolver, several fitness evaluation functions have been implemented. Some of these functions focus on geometric features of the designs they analyze, some on color-related features, and some on a combination of both.

A series of experiments was designed and performed in order to evaluate the performance of ShEvolver's evaluation functions in several ways. First, evaluation functions were compared to each other under the same experimental conditions and it was found that they can be classified according to whether they impose constraints that are very easy, somewhat easy, or very difficult to satisfy. Then some of the evaluation functions were combined in slightly different ways and the new ways of combining them were again run under the same experimental conditions. In this way, we found that even small differences in how an EA evaluates the solutions it proposes can cause large differences in the EA's performance. This effect was observed when measuring several EA parameters, specifically its reliability, its mean convergence time, the mean fitness value of its population at convergence, and the quality of the solutions it produced. In one of the experimental scenarios that was tested, a slight difference in how the evaluation functions were combined was also the difference between the EA being able to complete its task (the experiment) or not (because the computer ran out of memory space very quickly). The search paths produced by the slight differences in the implementation of fitness evaluation were widely divergent in this situation.

As we have seen with our experiments, even minor variations in the way that fitness evaluation is implemented can cause large differences in the performance of an EA. What this indicates is that, when using an EA, making a few preliminary trials that evaluate alternate fitness evaluation implementations might enable the construction of more robust, more efficient, more reliable EAs, capable of producing better solutions. Taking into account these lessons learned from our experiments can have a major impact on whether a particular EA will be able to successfully provide support for design or creativity in a given domain or not.

Acknowledgments This work has been supported by Asociación Mexicana de Cultura, A.C.

References

1. Bentley, P. (ed.): Evolutionary Design by Computers. Morgan Kaufmann, San Francisco, CA (1999)
2. Bentley, P., Corne, D.W. (eds.): Creative Evolutionary Systems. Morgan Kaufmann, San Francisco, CA (2002)
3. Gómez de Silva Garza, A.: Exploring the sensitivity to representation of an evolutionary algorithm for the design of shapes. In: Proceedings of the Eighth ACM International Conference on Creativity and Cognition (C&C'11), pp. 259–267, Atlanta, GA (2011). doi:http://dl.acm.org/citation.cfm?doid=2069618.2069661
4. Gómez de Silva Garza, A.: The impact of changing the way the fitness function is implemented in an evolutionary algorithm for the design of shapes. In: Proceedings of the Eighth International Conference on Knowledge, Information and Creativity Support Systems (KICSS 2013), pp. 104–113, Krakow, Poland (2013)
5. Mitchell, M.: An Introduction to Genetic Algorithms. MIT Press, Cambridge, MA (1998)

Human Smart Cities: A New Vision for Redesigning Urban Community and Citizen's Life

Grazia Concilio, Jesse Marsh, Francesco Molinari and Francesca Rizzo

Abstract The Smart City is by now a model for the application of Future Internet services and infrastructure towards radical improvements of urban services in terms of efficiency and sustainability. In recent experiences in this area, however, a new vision is emerging that enriches the original concept with the human perspective, as gained through the application of citizen-centric and participatory approaches to the co-design and development of Smart City services. Indeed, it is becoming clear that "smartness" alone—sensors, metres, infrastructures—risks placing the citizen outside of the process, as a user who never takes the kind of ownership of the services that can only be ensured by engagement in their co-design from the very start. This paper explores the emergent model for Human Smart Cities, its methodological components, and the "softer" features of "smartness"—such as clarity of vision, citizen empowerment, participation, etc.—which can complement the technological drive of the underlying urban infrastructure. Conclusions are set out in terms of implications and potential for EU policy renewal in the context of Europe 2020.

Keywords Smart city · Future internet · Local governance · Urban development · City services · Europe 2020

G. Concilio (✉)
Department of Architecture and Urban Studies, Politecnico di Milano,
Via Golgi 39, 20134 Milan, Italy
e-mail: grazia.concilio@polimi.it

J. Marsh · F. Molinari
Network for Social and Territorial Innovation (NeSTI),
Via M. Bonello n. 31, 90134 Palermo, Italy
e-mail: jesse@atelier.it

F. Molinari
e-mail: mail@francescomolinari.it

F. Rizzo
Design Department, Politecnico di Milano, Via Durando 38 A,
20158 Milan, Italy
e-mail: francesca.rizzo@polimi.it

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_21

1 Introduction

Cities and urban areas of today are complex ecosystems, where ensuring sustainable development and quality of life is an important concern. In the Digital Agenda of the European Commission [11], cities are considered as innovation drivers in areas such as health, environment, inclusion and business.

In such urban environments, people, businesses and public authorities experience specific needs and demands regarding domains such as healthcare, media, energy and the environment, safety and public services. These domains are increasingly enabled and facilitated by Internet-based applications, content management platforms and broadband infrastructures.

Therefore, cities and urban environments are facing challenges to maintain and upgrade the required infrastructures and establish efficient, effective, open and participative innovation processes to jointly create the innovative applications and services that meet the demands of their citizens.

In this context, cities and urban areas represent a critical mass when it comes to shaping the demand for advanced Internet-based services and experimentation in a large-scale open and user-driven innovation environment.

The term Smart City (SC) has attracted a lot of attention in recent years. Since the end of the 1990s many cities have initiated various SC initiatives. The concept captures different meanings and we must look beyond a superficial use of the term for pure city marketing purposes. In this paper, we aim to focus in particular on the defining role of the Internet and user-driven innovation.

A useful definition to start with is to call a city "smart" when "investments in human and social capital and traditional (transportation) and modern (ICT-based) infrastructure fuel sustainable economic growth and a high quality of life, with a wise management of natural resources, through participatory government" [4]. To this, the notion of empowerment of citizens and "democratized innovation" should be added [24]. Other definitions have tried to identify key domains of SC, such as smart economy, smart mobility, smart environment, smart living, smart people, smart governance, and have defined rankings based on measurable underlying indicators [12]. Smart cities can be also understood as places generating a particular form of spatial intelligence and innovation, based on sensors, embedded devices, large data sets, and real- time information and response [13].

While these definitions are relevant for benchmarking or for placing emphasis on specific development aspects, they are merely based on technology-led views. There is a need for research on effective strategies of cities to become smarter, taking into account the particular socio-economic context and urban development objectives, and on approaches mobilizing the participation and intelligence of citizens, businesses and societal organizations.

In recent experiences in the area, however, a new vision is emerging that enriches the original concept with the human perspective, as gained through the application of citizen-centric and participatory approaches to the co-design and development of SC services. Indeed, it is becoming clear that "smartness" alone—

sensors, metres and infrastructures—risks placing the citizen outside of the process, as a user who never takes the kind of ownership of the services that can only be ensured by engagement in their co-design from the very start.

This paper is structured as follows: Sect. 2 explores the emergent model for Human Smart Cities (HSC) and its methodological components; Sect. 3 explores (some of) the "softer" features of "smartness"—such as clarity of vision, citizen empowerment, participation, etc.—which can complement the technological drive of the underlying urban infrastructure. Conclusions are set out in Sect. 4 in terms of implications and potential of the HSC concept for EU policy renewal in the context of Europe 2020.

2 Towards a HSC Model

Figure 1 maps the existing SC visions along two axes

- *Innovation Strategies*, with those holding a predominance of technological innovation to the left and those with a shift towards social innovation to the right;
- *City Strategies*, with the lower extreme representing infrastructure and platform investments and the upper quadrants a greater emphasis on network building, citizen empowerment, and stakeholder engagement.

In particular, there are four options at hand

- **Smart Grids and Sensors**, to the lower left with the pair technical innovation—infrastructure policy. This approach characterizes the large-scale Smart City initiatives such as U-City Songdu (South Korea) but also the first Commission initiatives such as SETIS (Strategic Energy Technologies Information System) and FIRE/Smart Santander as well as the general IERC approach to the SC problematic, even within the CIP (ICT-PSP) context;

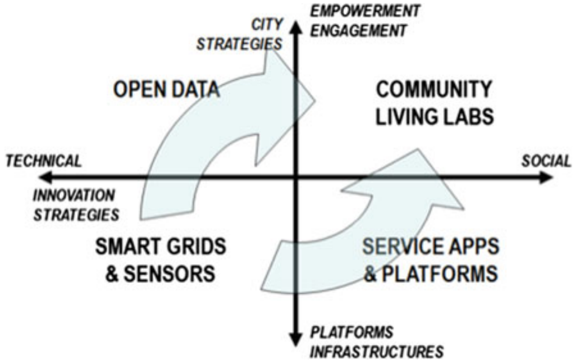


Fig. 1 Transitioning towards a HSC

- **Open Data**, to the upper left with the pair technical innovation—empowerment policy. This approach is emerging with force across the digital innovation community and with several leading public administrations, and is also the focus of interest of several more recent projects in the EU SC Portfolio. Open Data emphasizes the free publication of public sector information, under the assumption that external developers build applications to access these datasets to provide services, and is essentially a technical approach to citizen empowerment;
- **Service Apps and Platforms**, to the lower right with the pair platform policy—social innovation. This approach is often, but not always, complementary to Open Data strategies, and places the emphasis on the development of open participatory platforms and the development of apps for City services. Although shifting the emphasis to social innovation, especially as regards the development of new services, this approach still relies on technology development as its main strategy;
- **Community Living Labs**, to the upper right with the pair empowerment policy—social innovation [15]. This approach relies on technology as the enabling infrastructure more than the final purpose, while shifting the emphasis to putting citizens and communities at the centre of the SC vision.

What is common to the four quadrants is the firm commitment from the City governments involved, to develop a new generation of public services that are transformational rather than merely responsive to existing or emerging community needs, and impact on the whole relationship between service providers and users/beneficiaries. This trend has been recently structured into the so-called Transformational Government Framework [19], a set of practical, "how to" guidelines and standards for the design and implementation of effective programs of technology-enabled change.

As shown by parallel research [10, 17, 18], however, most of the challenges that City governments face nowadays—in such domains as public safety, climate change, water and energy supply, healthcare, mobility and education—do require far more than a mere deployment of IT service providing infrastructures. Also, in times of financial crisis and tight public budget constraints, this agenda is rendered overly difficult by the additional requirement of "making more with less", while individual and societal expectations continue to grow in relation to lifestyle changes and other external strains (e.g. job losses, family breaks, immigration flows).

Responding effectively to such new challenges requires a "next generation" of city and community services, holding the following characteristics:

- They should generate some degree of change in the behaviour of the citizens involved in technology adoption, service fruition and/or co-production: namely, the various forms of micro-level interaction (human-to-human, human-to-thing, human-to-machine and application) implied by the take-up of Future Internet technologies should make people's attitudes and lifestyles more proactive, not only during each service delivery session, but even afterwards;

- They should activate (mass) imitation, networking or multiplication effects, meaning that their final outcome in terms of innovation, collective action and quality of life shall be more than the sum of individual behavioural changes, because it is self-reinforced by the initial positive achievements and propagated to a far broader audience than the one that has contributed to shaping and launching the service;
- They should be sustainable–not only financially, which would also be good per se, but also from a societal and institutional point of view. Here, sustainability might also be interpreted in its original meaning, since most SC services, such as those related to energy, mobility and transport, are also strictly connected with the (natural and urban) environment–and sustainability thereof.

A key feature of the transition path described by the arrows in Fig. 1 given above is the shift from policies based on large-scale infrastructure projects towards policies that aim to create and nurture socio-technical (or rather "human-digital") innovation ecosystems.

Building an ecosystem is quite a different matter than, say, building a new electricity grid, and the traditional policy instruments generally fail to capture this organic dimension, significantly increasing the risk of failure. As social (or societal) innovation occurs through processes that are more ecosystemic than institutional in nature, HSC policy strategies will need to find a way to promote and guide the birth and development of innovation ecosystems without killing them in the process.

A number of state of the art methodologies can be adopted in support of this effort, such as Participatory Design [7, 14], Open Innovation [20], Living Labs [6, 9], and Design Thinking [3, 22]. These approaches all place the end user at the centre of innovation processes, leading to more effective solutions that more effectively meet peoples needs and in addition enjoy faster uptake.

By this embodiment of "People in Places" in the technologically driven concept of "urban smartness" [5], Future Internet technologies can thus become the platform for the development of new and more sustainable ways of living and working in domains of crucial relevance for local government as well as community life.

While there is no simple formula for making social innovation to happen, the EU PERIPHÈRIA project has made substantial progress in defining methods and tools that have been tested over two years of experimentation in six cities, in fact giving birth to the HSC model. During its final conference in Rome, the project consortium has presented the "Arena Cookbook", an easy-to-use set of guidelines for identifying opportunities, engaging citizens, and co-creating new scenarios and services, as well as the business models for sustainability of innovative city services. PERIPHÈRIA has applied the Arena concept derived from the tradition of participatory software design to a systematic analysis of city structures as a starting model for framing broad-reaching service co-design processes.

3 Five Lessons Learnt

The concept developed in PERIPHÈRIA—like the projects such as SAVE ENERGY it builds upon—is grounded on complementary "softer" features of "smartness" such as clarity of vision, citizen empowerment, participation, etc. The power of these components has been summed up as five lessons learnt in the experience of more than two years of work. These lessons, interpreted within the logic of "people in places", led us to emphasize the anthropological (social and political) reading of the convergent innovation processes occurring in the PERIPHÈRIA Arenas. The five lessons are as follows:

1. *Participatory service co-design processes require institutional framing in order to ensure the required commitment*. The six PERIPHÈRIA cities have worked in parallel and in sequence at the administrative, associational, and citizen levels, first ensuring reciprocal acknowledgment as a prerequisite to the actual initiation of more visible participatory activities. The range of strategies adopted indicates the complexity and its high relevance towards the goal of impact.
2. *The development of HSC ecosystems requires a constant monitoring of the coherence of activity strands*. While the concept of digital innovation ecosystems is the main metaphor for Smart Cities, our understanding of it is primarily based on ex-post analyses rather that experiences of creation of these ecosystems from scratch. The latter is a very complex task indeed, which requires the orchestration of various dynamics to be set into motion, often with parallel or divergent trajectories and following non-linear paths. The experience of PERIPHÈRIA illustrates that even though successful results may be obtained, the lack of a traditional (linear) development or transfer methodology can be disorienting. It is thus necessary to maintain a strong and fluid communication between all parties, with a constantly updated overview of the dynamics in play, so that all actors know "where they are" with respect to the overall ecosystem, and ensure that the systemic co-design interactions continue to develop in a coherent way and move in the direction of desired objectives.
3. *Future Internet service platforms need to combine the technological, social, and spatial dimensions*. The Future Internet has in the first instance been defined by a series of attributes of underlying technologies (in PERIPHÈRIA synthesized as IoT, IoS and IoP—see also [8] for a similar view). These come together through a dedicated platform that however resembles system architectures currently in place (i.e. layered models with middleware services, application modules, etc.). In our experience, a different view is emerging. The Challenges platform developed in PERIPHÈRIA [2] integrates the social and technological dimensions with a spatial relevance, supporting human-driven, place-based convergence and interoperability through co-design rather than mere technical standards compliance.
4. *Engaging stakeholders in the co-design of Future Internet-based services must provide a substantial and relevant motivation and address technology potentials in the context of their behavioural impacts.* As a corollary of the above

statement, many of the technology components of the Future Internet—especially the Internet of Things—can already be said to be available on a mass scale today (with some elasticity in definitions). Further technology R&D will certainly increase their scale, scope and effectiveness, but it is already possible to experiment the interaction of these elements with the social dimension, which is central to sustainability and impact on user behaviour. Paradoxically, however, as Facebook surpasses one billion subscribers and Smartphones are the only product category to flourish in the midst of the economic crisis, it becomes increasingly difficult to engage stakeholders in co-design of these services, as far as people remain in the state of "passive users" rather than "active co-producers". That is why PERIPHÈRIA's cities attempted to align their motivational efforts with other ongoing citizen engagement activities, in order to capture active citizenship or Living Lab processes within which to introduce or reinforce the transformational dimension of Future Internet.

5. *Broad and ambitious re-shaping of Smart City structures and services requires both top-down and bottom-up approaches, with a key role for place-based development policies*. Combining the disruptive potential of Future Internet with a focus on the transformational changes required to attain sustainability and behavioural impact has situated PERIPHÈRIA in a wide open space, where traditional sector and market structures appear increasingly inadequate as reference frameworks for—and become themselves objects of—innovation. Sectorial classifications that structure the IT Smart City market to date (e.g. energy, health, transport) remain valid within goals based on efficiency, but can hinder our understanding of the dynamics of social innovation and service co-production, which instead require more transversal approaches. In this context, a **place-based model** of urban development and territorial cohesion [1] appears to offer a structure within which to shape these dynamics in a multi-level governance scheme, allowing for participatory co-design and open institutionalization processes to take shape. Within this model, in order to actively engage stakeholders, a dialogue between top-down and bottom-up approaches is strictly required. However, the dialectic between top-down and bottom-up to be re-read in terms of roles rather than weights: the role of the top-down—the only possible policy perspective—can no longer be to plan or design social innovation (which is in contradiction with the open nature of innovation itself), but rather to cultivate, align with, and accompany emergent processes, helping them to synergize and scale up towards becoming new societal structures. This means that the top-down perspective must learn to read "weak signals" and to recognize and capture processes at the micro-scale, forming innovation clouds rather than networks, in the sense that the connections are "liquid" and not always readable.

Our experience in PERIPHÈRIA shows that in order to promote an innovation ecosystem, the first step is to define a map of the ecosystem in question, with its key components and dynamics. Actions within this strategy can be initiated from any point on this ecosystem map and can be preferably carried out in parallel as independent strands of development, but the fundamental point is to maintain an overview of the ecosystem in its entirety and how each of the strands is contributing to its systemic evolution.

Drawing up such a map is also a fundamental step for orienting co-design participants and stakeholders, since the process in its development is quite different from the traditional linear fashion, where different actions are related to each other as sequential steps on the same critical path.

This need is further enhanced by considering that most of the urban landscapes selected for the pilots of PERIPHÈRIA were located in peri-urban and rural areas, even of major European cities, therefore including a relevant fraction of world inhabitants whose creativity and innovation potential is often discarded by external analysts.

4 Conclusions

The HSC vision deserves consideration from city governments across Europe as well as in the SC research community, as it more effectively addresses key challenges that require the commitment of large numbers of technologically savvy citizens, such as a low-carbon economy, urban redesign, sustainable mobility, and community-led digital public services, through a more balanced, holistic approach to technology.

In the context of Europe 2020, HSC represents an important opportunity to create a bridge between three cornerstones of the forthcoming programming period 2014-2020, namely:

1. Horizon 2020, to integrate Digital Social Innovation in concrete research and experimentation settings as required;
2. Regional Smart Specialization, one of the key aspects of conditionality being negotiated with DG REGIO within the next Structural Funds, and in particular the Social Challenges as applied to urban settings and the specific topics of Smart Cities and Communities;
3. The Urban Agenda, as currently being developed by the Urban Unit of DG REGIO.

HSC can be thought of as a transversal, ICT driven, "lead market", meeting the requests of the Council and the Parliament for "strengthening synergy between EU support policies in the area of research and innovation"—see [21]. Indeed, some of the instruments through which such an approach could be enacted include:

- Support for the inclusion of HSC approaches within the design of "shared" Regional Smart Specialization Strategies, in particular as regards Digital Social Innovation, promoting specific partnerships and initiatives as a measure of conditionality;

- Coordination of R&D&I initiatives funded by Horizon 2020 with ERDF-supported inter-regional cooperation, under specific ERDF provisions for Integrated Territorial Investment (ex art. 99 of Common Provisions for ERDF & ESF), Community-led Local Development (ex art. 28§1 of Common Provisions), and Inter-regional Cooperation (ex art. 37(6)(b) of the Regulation EC No. 1083/2006 and further integrations for the 2014–2020 period), as well as planned instruments and initiatives such as the "Innovative actions in sustainable urban development" [23], through, e.g. a favoured status for Horizon 2020 proposals coupled to pilot initiatives co-funded by ERDF—and/or vice versa.
- Adoption of Pre-Commercial Procurement as a transversal instrument for the promotion of regional R&D&I, linking cooperative EU-level research (e.g. binding with Horizon 2020 objectives) with the societal challenges raised by City and Regional policies (compare the example of a recent Italian Ministerial PCP call— [16]).

Acknowledgements This paper has been developed in the context of the EU-funded project PERIPHÈRIA, belonging to the CIP (ICT-PSP) Smart Cities cluster. For more information: http://www.peripheria.eu. We also mention the Network of Social and Territorial Innovation (NeSTI) as well as the Budapest Manifesto, which two of the authors have developed in the context of the TC CEE project CENTRALAB: see http://www.centralivinglab.eu/NeSTI. However, none of the statements made here involve or commit any European institution.

References

1. Barca, F.:. An agenda for a reformed cohesion policy. Independent report prepared for the commissioner Ms. Danuta Hubner. http://ec.europa.eu/regional_policy/archive/policy/future/barca_en.htm (2009). Accessed 01.10.2014
2. Barroca, J.: Human smart cities. In: Proceedings of the EISCO 2012 Conference on Open Government in Europe: a challenge for the Digital Local Agenda. Guimaraes, Nov 12–14 (2012)
3. Brown, T., Wyatt, J.: Design thinking for social innovation. In: Stanford Social Innovation Review, Winter (2010)
4. Caragliu, A., Del Bo, C., Nijkamp, P.: Smart cities in Europe. Research Memoranda 0048. VU University of Amsterdam, Faculty of Economics, Business Administration and Econometrics (2009)
5. Concilio, G., De Bonis, L., Marsh, J., Trapani, F.: Urban smartness: perspectives arising in the periphèria project. J. Knowl. Econ. **3**(2) (2011). doi:10.1007/s13132-012-0088-5
6. Concilio, G., De Bonis, L., Trapani, F.: The territorial dimension of living lab approaches: starting the European "participant observatory". In: Cunningham, P., Cunningham, M., (eds.) eChallenges e2011 Conference Proceedings, Florence, 26–28 Oct 2011
7. Concilio, G., Deserti, A., Rizzo, F.: The human approach to smart cities: the periphèria project as a case of complex participatory design. In: 4th Latin American Symposium on Design as a Process, Belo Horizonte, Brazil, 19–22 Sept 2012
8. Cretu, L.G.: Smart cities design using event-driven paradigm and semantic web. Informatica Economică **16**(4), 57–67 (2012)

9. De Bonis, L., Marsh, J., Trapani, F.: Smart cities for real: the territorial living labs approach as engine of spatial innovation. In: Proceedings of the EURA 2012 Conference on Urban Europe —Challenges to Meet the Urban Future. Vienna, 20–22 Sept 2012. ISBN: 978-3-200-02766-4
10. Duarte de Oliveira, A.: The human smart city vision: presentation given at the "smart city exhibition", Bologna, 30 Oct 2012
11. European Commission, Digital Agenda for Europe. A Europe 2020 Initiative. http://ec.europa.eu/digital-agenda/en (2010). Accessed 01.10.2014
12. European Smart Cities, Ranking of European Medium-Sized Cities. Final Report. Centre of Regional Science (SRF), Vienna University of Technology Department of Geography; University of Ljubljana, Research Institute for Housing, Urban and Mobility Studies (OTB); Delft University of Technology (2007). http://www.smart-cities.eu/
13. IBM Intelligent Operations Center for Smarter Cities. http://www-03.ibm.com/software/products/us/en/intelligent-operations-center (2013). Accessed 01.10.2014
14. Manzini, E., Rizzo, F.: Small projects/large changes: participatory design as an open participated process. In: CoDesign, **7**(3–4), 199–215. Taylor & Francis (2011). ISSN: 1571-0882 print/ISSN: 1745-3755
15. Marsh, J.: Living Labs and Territorial Innovation. In: Cunningham, Paul, Cunningham, Miriam (eds.) Collaboration and the Knowledge Economy: Issues, Applications. Case Studies. IOS Press, Amsterdam (2008)
16. MIUR/MISE: Avviso Pubblico per la rilevazione di fabbisogni di innovazione all'interno del settore pubblico nelle regioni convergenza, Prot. N. 437. http://attiministeriali.miur.it/anno-2013/marzo/di-13032013.aspx13 Mar 2013. Accessed 01.10.2014
17. Molinari, F.: Dati aperti, Free Apps, Servizi e Politiche "Smart": Un senso a questa storia. In: Proceedings of the ITN 2012 Expo-Conference on Infrastructures and Technologies for the Smart City. Turin, 27–28 Sept 2012
18. Molinari, F.: Innovative business models for smart cities: overview of recent trends. In: Proceedings of the 12th European Conference on eGovernment—ECEG 2012. Barcelona, 14–15 June 2012
19. OASIS: Transformational Government Framework Primer Version 1.0, Committee Note Draft 01. http://docs.oasis-open.org/tgf/TGF-Primer/v1.0/cnd01/TGF-Primer-v1.0-cnd01.docx (2010). Accessed 01.10.2014
20. Paskaleva Antonova, K.: The smart city: a nexus for open innovation? In: Intelligent Buildings International 3, 153–171 (2011). doi:10.1080/17508975.2011.586672
21. Regional Policy contributing to Smart Growth in Europe 2020. COM—553 Final. European Commission, Brussels (2010)
22. Rizzo, F.: Co-design versus user centred design: framing the differences. In: Guerrini. L. (ed.) Notes on Design Doctoral Research, Franco Angeli, Milano (2011), ISBN: 978-88-568-3513-7
23. Svanfeldt, C.: The Urban Dimension of Future Cohesion Policy. Presentation delivered in Brussels, 30 Jan 2013
24. Von Hippel, E.: Democratizing Innovation. MIT Press, Cambridge (Mass.). http://mit.edu/evhippel/www/democ.htm. Accessed 01.10.2014

The Role of Creativity in the Development of Future Intelligent Decision Technologies

Andrzej M.J. Skulimowski

Abstract This paper presents a methodological background and selected final results of a foresight project concerning the role of creativity in the development of intelligent decision technologies. Technological trends and scenarios have been generated via a simulation of a hybrid system consisting of discrete-time control and discrete-event components. Both form a complex information society model, which describes the evolution of social, economic and scientific factors relevant to the production and adoption of intelligent technologies. The trends and scenarios derived are then discussed and refined during cooperative expert activities. Specifically, we have investigated the development of intelligent decision technologies, with special attention paid to web-based decision support systems, neurocognitive and autonomous systems, as well as artificial creativity aspects. The overall project is outlined in Sect. 2. In Sect. 3, we will present selected trends related to the development of creative technologies in the context of overall progress in information and communication technologies (ICT) and computer science (CS). The discussion of the future role of creativity in the design and implementation of intelligent systems is based on the results of a Delphi study carried out within the recent foresight project SCETIST.

Keywords Artificial creativity foresight · Multi-round delphi · Trend analysis · Creativity support systems · Intelligent decision technologies

---

A.M.J. Skulimowski (✉)
Chair of Automatic Control, Decision Science Laboratory, AGH University of Science and Technology, al. Mickiewicza 30, 30-050 Kraków, Poland
e-mail: ams@agh.edu.pl

A.M.J. Skulimowski
International Centre for Decision Sciences and Forecasting, Progress & Business Foundation, Lea 12B, 30-048 Kraków, Poland

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_22

1 Introduction

The foundations of the research described in this paper go back to a report [13] prepared for FISTERA, an EU 5th Framework Programme Thematic Network, by the International Centre for Decision Sciences and Forecasting, Cracow. The aim of this report was to identify new trends, processes and phenomena regarding the current state of the information society, as well as technological, social and economic trends in the ten EU New Member States that acceded in 2004 and 2007. The project team made a number of unique observations related to the evolution of the information society (IS) and its technologies. This research has continued within the foresight project SCETIST (Scenarios and Development Trends of Selected Information Society Technologies until 2025) [22] resulting in new complex system modelling methods.

This paper provides an overview of the methodological background of SCETIST and the application of its selected results to eliciting future trends related to the role of creativity in the development of future artificial intelligent systems and technologies. The project examined trends in artificial intelligence technologies such as intelligent decision support systems, computer vision, global expert systems [18] as well as neurocognitive and autonomous systems [20] until the year 2025 and beyond. The principal application areas under study were the development of intelligent internet technologies and service systems within e-health and e-learning [19]. In addition, the project included a work package devoted to quantum and molecular computing. The overall research focussed on the evolution of intelligent decision support systems and recommenders in the context of global trends in information and communication technologies (ICT), computer science (CS), the economy and social behaviour.

Unlike the usual approaches to foresight exercises, technological trends and scenarios were generated by combining the outputs of an expert Delphi with bibliometric and patentometric indicators and a simulation of a hybrid socio-econometric system. The latter consisted of discrete-time control and discrete-event components, both describing the information society evolution [16, 17]. Several customised applications were integrated together to form a Foresight Support System (FSS) capable of modelling the IS [16]. This FSS is also able to manage information on the evolution of social, economic and scientific factors relevant to the production and adoption of intelligent technologies. Its discrete-event component [17] models the impact of legislative, research and development (R&D) and science and technology (S&T) policies on each technological area under examination. This is described in more detail in the above-cited papers as well as in the next section.

While conducting the above research, it became necessary to elaborate new methodological approaches, as the methods commonly used to model IS phenomena were found to be insufficient [2, 5, 24, 28]. The general rules and principles that govern the evolution of the IS and ICT were retrieved from historical data and expert replies, then formulated in a mathematically strict manner, analysed and applied to model the future development of advanced intelligent technologies. In [16, 22], we discuss the general applicability scope of this model beyond the above research areas.

The aim of the project SCETIST was to deliver results that should be constructively applied to developing technological policies and strategies at different levels, from corporate to multinational. A wide spectrum of modelling methods such as discrete-event-systems, multicriteria analysis, and discrete-time control were used to better understand the role of global IST development trends and to develop IS and ICT policies in an optimal control framework. A number of partial goals have been defined, which might help to achieve the ultimate objectives. These include:

- Constructing a hybrid discrete-time/discrete-event information society model that uses general technological, social, economic and behavioural trends as input information and a background for technological forecasts [17].
- Analysing technological trends and scenarios in selected areas such as expert systems, decision support systems, recommenders, neurocognitive technologies, m-health, autonomous robots, as well as development prospects of quantum and molecular computing [22].
- Devising a novel methodology for constructing evolution scenarios of ICT and conditional technological forecasts with controlled discrete-event systems.
- An ontological self-updating knowledge base which stores raw data together with IS/ICT models, trends and scenarios. The knowledge base records modification history in the form of so-called *proceedings* containing data together with details of updates. Similar proceedings can record step-by-step analyses of data by users.
- Devising multicriteria outranking methods suitable for ICT management and capable of generating constructive recommendations for decision makers concerning investment in information technologies for the development of a company, region or brand.
- Analysis of new product development cases [23] in selected technological areas submitted by industrial partners of the project.

The above partial objectives provided useful solutions to the technology management problems presented by the institutions involved in the foresight exercise as stakeholders. The knowledge gained allowed them to set strategic technological priorities as well as formulate ICT and R&D investment strategies. This is discussed further in Sect. 4 as well as in [23].

2 The Evolution of Intelligent Technologies

The development model for each of the technological areas under consideration indicates the demand for technologies and their end-products, the ability to create innovations, as well as the legal and economic conditions of technology life cycle.

These factors are common for all the technological areas mentioned in Sect. 1, although each factor has a different degree of influence. Thus, the general foresight methodology is to elaborate a common information society model that will serve all technological areas, then to merge it with specific R&D and technological models for each area under consideration. Both types of model components differ in a few fundamental features: the information society model describes a given country, or group of countries, while the research and technological development trends are globally relevant due to the rapid diffusion of ICT innovations. Furthermore, the data input to the information society model will originate mainly from publicly available statistical sources, while the S&T trends will be derived from bibliographic and patent data [2, 28] as well as from an expert Delphi survey [6, 20, 22].

This section describes the main research problem related to an advanced information technology foresight [13, 22], namely how the development of a selected high-end technology or research area depends on the global processes of ICT development, on social issues such as ICT education, and on global economic trends. Global trends considered in the model built for the project include, among others:

- falling telecommunication prices,
- the amount of data available on the web and its annual exchange through the internet,
- the growing relevance of mobile technologies,
- the technical and physical parameters of basic technologies and components,
- the diffusion of information innovations and technologies.

Social trends, such as evolving ICT consumption patterns, preference dynamics, and the growing availability of e-services and social web content have also been taken into consideration.

It is well known [13] that the sole analysis of economic indicators based on statistical data does not sufficiently explain future social and economic behaviour that is relevant to understanding the information society phenomena. Therefore, we have introduced a new class of input-output models with state variables that fit well into the IS and its advanced artificial intelligence (AI) technologies specificity. The technological evolution model consists of eight major subsystems of an information society, such as population and demographics, the legal system and IS policies, ICT for personal and industrial use, etc. Their interaction properly characterises the evolution of the overall system, which is described in more detail in [22].

During analysis, each of the factors appears as a bundle of discrete events, continuous trends and continuous or discretised state variables. The latter may be assigned numerical or symbolic values, or none of them, being just empty frames. The evolution of the information society is then modelled as a discrete-continuous-discrete-event system, where the mutual impacts of each of the elements are represented either in symbolic form, as generalised influence diagrams, or within state-space models. External controls, such as legal regulations and policies, are modelled as discrete-event controls. Technological trends form inputs, while the feedback loops allow us to model the influence of technological demand on ICT, R&D, production and supply. A causal graph of the underlying dynamical model was derived from an analysis of a time series describing macroeconomic and social indicators, from participatory group model building [17], and from an expert Delphi [22].

After scaling the dynamics based on past observations, key technological, economic or social characteristic trends can be described quantitatively as solutions to discrete-time dynamical systems of the form

$$x_{t+1} = f(x_t, \ldots, x_{t-k}, u_1, \ldots, u_m, \eta_1, \ldots, \eta_n) \tag{1}$$

where $x_t, \ldots, x_{t-k}$, are state variables, $x_j = (x_{j1}, \ldots, x_{jN}) \in IR^N$, $u_1, \ldots, u_m$ are controls, and $\eta_1, \ldots, \eta_n$ are external non-controllable or random variables. In the models analysed in [17], f was linear non-stationary with respect to x, and stationary with respect to u and η.

The evolution of complex systems that occur in a creativity and AI future study may be governed by discrete-time control systems of type (1), where controls may be identified with state or regional policy parameters. The function f may vary according to the evolution of a discrete-event system that describes the influence of political decisions, scientific discoveries or disruptive technologies [17]. A sequence of events and continuous transitions along elicited and simulated trends form a so-called *elementary scenario* [16], whose quantitative part is a trajectory to (1).

Recommendations to policy makers were derived with multicriteria prioritisation tools based on defining the reference values of objectives [12] combined with the dynamic version of the SWOTC analysis. SWOTC, or "SWOT with Challenges as an additional element", is a new method for technology assessment used first in [13]. It allowed us to properly characterise the intelligent technologies under consideration and their development trends. The IS evolution model (1) and SWOTC yield a dynamic technology benchmarking algorithm [22] which may also be used to generate optimal technological strategies or investment policies at the enterprise level.

2.1 Technological Areas Covered by SCETIST

The model presented above was coupled with a knowledge base that includes ontology management functionality, specifically ontology merging and splitting, time evolution, operations on metadata as well as data updating protocols. The usual data warehousing functionality was implemented as well, including automatic updating. An instance of a knowledge base serves to gather and process the information related to one of the technological focus areas of the project. These are listed below:

- basic internet and software technologies,
- key information society application areas (e-government, e-health, e-learning, e-commerce),
- expert systems, decision support systems and recommenders,
- machine vision and neurocognitive systems,
- molecular and quantum computing.

In addition to the above, the industrial partners of the project could propose additional subject areas for technological development forecasts. The thematic databases store the above area-specific information, while a common data block contains interdisciplinary information to be used during thematic analyses, such as macroeconomic data, social characteristics (employment, education and demographics), geographic information and other data potentially useful in providing decision-making support.

A number of specialised foresight data management and data analytics algorithms were implemented to process the data gathered. The collection of analytic methods implemented in the foresight support system [16] includes:

- enhanced trend-impact and cross-impact analysis,
- ICT consumption and consumer preference models,
- specific sector and market models concerning information society technologies and applications such as e-commerce, e-education, e-health and care services, social media, internet advertising, quantitative information market, etc.
- time series forecasting with autoregression, state-space models, and adaptive trend algorithms.

Recommendations for the project stakeholders have been generated making use of technological forecasts and multicriteria outranking methods [12, 15] tailored to ICT prioritisation problems.

The knowledge gathered in the system is continuously updated, represented and processed using Bayes networks, other types of causal models, automatic rule generation techniques and anticipatory feedback models [21]. The latter were defined and investigated within SCETIST as a technique for building conditional, future decision-dependent scenarios and for filtering them according to a defined decision rule.

2.2 *Multi-round Online Expert Delphi as a Participatory Cooperative Process*

The success of foresight research depends on the knowledge and diversity of the expert opinions. Among the ICT-based foresight tools, the most common is the online Delphi analysis, in which selected experts in various fields, from industry, academia and government complete questionnaires. The quantitative results presented in this paper were achieved with the common effort of all scientists, specialist-practitioners and ICT users in other industries who participated in collaborative research online, and completed the Delphi survey. The flexible multi-round/real-time online Delphi that was used in SCETIST to elicit knowledge on creativity future is outlined below.

The Delphi questionnaire on intelligent technologies comprises over 100 questions organised into 14 topics. The other Delphi areas relate to the future development of basic IT, socio-economic trends, fundamental issues concerning neurocognitive and vision systems as well as quantum and molecular computing. A screenshot of Questions 13.1–13.4 in the survey section devoted to creativity in the context of expert systems, decision support, and learning is shown in the above Fig. 1.

The Delphi survey technique is the most frequently used method in foresight studies (cf. e.g. [6]). The method consists in conducting a questionnaire survey among a pre-selected expert group. The survey is conducted several times, called rounds. During each round the experts involved should not contact each other to discuss the subject matter. Each expert is asked to justify the presented results in terms of related rules, trends and events. The process is usually repeated until a consensus or a 'well-justified dissensus' [27] among the experts is reached. The existence of the latter may follow, e.g. from different views of future political decisions and scientific discoveries. In the case of a dissensus, several future variants of the trend or phenomenon under investigation may emerge. Progressing to the next round is possible immediately after an expert provides a complete response to the questionnaire in the current round and after collecting a specified minimum number of responses during this round. Thus, a multi-round Delphi survey can be regarded as a persistent anytime process.

The online multi-round Delphi support system applied in SCETIST comprises a unique tool allowing us to permanently update foresight results and to model the evolution of expert views. The survey was conducted online with questionnaires of diversified degrees of interaction and complexity. Before responding to questions on a given topic, the experts could become acquainted with a short introduction to the related research problems. They could enter the exercise at any time, and after completing the online forms, they could view the results of the analysis. In addition, the experts could provide information about their expertise concerning each part of the questionnaire, which made it possible to include the opinions of people who were not specialists within all the research areas but who possessed unique knowledge in a specific area. The Delphi support system enabled us to conveniently manage the credibility of experts, which is essential when addressing a large group of respondents of diverse educational backgrounds and professional experience.

Once the online forms were completed, we performed the statistical analysis and used AI-based techniques to build models of technological, economic and social development. This Delphi elicited mainly expert opinions on the probability and timing of specific future events. These data were used to construct future trends and scenarios, rank key technologies and provide recommendations for decision makers. The statistically processed and verified information was used as the initial material for panel workshops, where it was discussed by experts and stakeholders.

13. Problems of creativity in designing expert and decision support systems, and in decision making

INTRO:
The problem of creativity in the context of implementation of creativity support systems and creativity displayed by more...

No.	In the timeframe until 2025, further development of creativity support systems will occur pertaining to different areas of creativity. Please estimate the value of given quantitative parameters of creativity support systems and autonomous creative systems	2015	2020	2025	Further development until 2030	Notes, explanations, sources and forecast justifications	Degree of certainty
13.1	Adequate and universal creativity indices and creativity measuring methods based on the effects of creative activities and psychophysiological parameters (such as eye movements, pulse, blood pressure, etc.) will be created, allowing for real time feedback in creativity support systems. Please provide the cumulative probability of the creation of such indices in the given timeframe	%	%	%	%		please choose: only expressing my subjective opinion
13.2	Cognitive research will allow for the formulation and implementation of active methods supporting the effectiveness of the learning process, fitted to different disciplines. Please provide the cumulative probability (in %) of implementation of widely available system with the following features:						
	a) system uses sensors determining user perception capabilities based on his/her physiological parameters during the learning process	%	%	%	%		please choose: only expressing my subjective opinion
	b) system is able to determine the level of concentration of an online user	%	%	%	%		please choose: only expressing my subjective opinion
	c) system measures learning efficiency in a given timeframe	%	%	%	%		please choose: only expressing my subjective opinion
	d) system measures overall user creativity (performs creativity tests)	%	%	%	%		please choose: only expressing my subjective opinion
	e) system measures creativity when applying of acquired knowledge	%	%	%	%		please choose: only expressing my subjective opinion
	f) based on (c) and (e) system recommends actions increasing creativity in learning and applying acquired knowledge	%	%	%	%		please choose: only expressing my subjective opinion
	g) system stimulates the development of general creativity	%	%	%	%		please choose: only expressing my subjective opinion
13.3	Please specify the expected number of training areas for which specialised systems enhancing learning creativity will be created. In the field "Notes", please specify the areas in which these systems are needed the most (up to 10 areas), and for which areas the implementation of such systems could present a problem and why.						please choose: only expressing my subjective opinion
13.4	Cognitive systems supporting creative processes (scientific research, literature, poetry, musical composition, industrial design, painting etc.) will be created. Please provide the cumulative probability (in %) of creation of a widely available system with the following features:						
	a) system uses sensors determining online user creative capabilities based on his/her physiological parameters	%	%	%	%		please choose: only expressing my subjective opinion
	b) system measures user satisfaction with the work done	%	%	%	%		please choose: only expressing my subjective opinion
	c) system measures overall user creativity (performs creativity tests fitted to particular area)	%	%	%	%		please choose: only expressing my subjective opinion
	d) based on (b) and (c) system recommends activities increasing user's creativity	%	%	%	%		please choose: only expressing my subjective opinion
	e) system stimulates the development of general creativity	%	%	%	%		please choose: only expressing my subjective opinion

Fig. 1 An excerpt from the Delphi survey topics devoted to creativity (Questions 13.1–13.4) available at http://www.ict.foresight.pl

After collecting and analysing the Delphi results, the project manager developed more narrowly focused questionnaires, to be completed and discussed at panel workshops. The variants of technological and economic characteristics derived from the Delphi survey may serve to build future scenarios. First, the variants can be embedded in an anticipatory network [20] to identify non-rational decisions responsible for the occurrence of some variants and to filter them out. The remaining rational variants are then clustered to form 3–5 explorative and descriptive scenarios for each group of technologies [22].

The conclusions of the survey were formulated in the expert panels, consisting of the most active experts taking part in at least two rounds of the questionnaire research. The outputs of other collaborative tools used in SCETIST, such as the IS model [17] and SWOTC were merged with the Delphi results at the scenario building stage.

3 Delphi Results Concerning the Future of Artificial Creativity

A presentation of the complete results of SCETIST regarding creativity issues would exceed the scope of this paper. Beyond Questionnaire 13, they occurred as secondary aspects in various survey topics regarding decision support and expert systems, computer vision, e-health or neurocognitive systems. Instead, we present an illustrative sample of results of the Delphi survey addressing the questions concerning creativity support systems (CSS) and artificial creativity contained in the Questionnaire 13 of the survey. The "Intro" section of this questionnaire (cf. Fig. 1) presented the main creativity-related ideas quoting a few relevant references only. Experts from different disciplines could respond to this part of the survey based on the common sense notion of creativity. A more profound scientific discussion on the nature and definitions of creativity should surely refer to the fact that out of the three basic features of creativity discussed in the literature (cf. e.g. [11]), namely novelty, usefulness, and surprise, the second cannot be derived from real-time neurophysiological measurements. These are regarded as a key feedback technique to be used in future CSS.

The tables contained in this section provide standard statistical indicators as well as the values of two interquantile ranges that are used as consensus measures between expert opinions [27]. The *interquartile range* (IQR), which is the most widespread consensus measure in Delphi surveys, is defined as the difference between the 3rd and 1st quartile. In other words, it provides an interval where 50 % of all replies between the 25 and 75th centiles fit. Along with the IQR, we provide a stronger consensus measure, termed the interquintile range or Pareto range (IQVR), defined as the difference between the 4th and 1st quintile. In other words, it takes into account the replies that range between the 20 and 80th centiles. IQVR measures the scope of replies after removing the upper and the lower 20 % of responses, hence the term *Pareto range* that refers to the well-known Pareto principle. To assess whether a consensus was achieved, IQR and IQVR are often compared to the length r_q of an interval $[r_{q,\max} - r_{q,\min}]$, where all reasonable replies might belong, i.e. $r_q = \left| r_{q,\max} - r_{q,\min} \right|$. This *reasonability interval* should be specified during the survey design for each question q separately.

Any of the above consensus measures, when divided by r_q, yields a consensus indicator that should then be compared with a certain threshold value γ that may be common for a group of questions. For a fixed γ the consensus condition IQVR/$r_q \leq \gamma$ is, of course, stronger than the requirement IQR/$r_q \leq \gamma$.

However, in SCETIST, specifically in the survey on creativity, the achievement of the consensus was not the ultimate goal. On the contrary, it was assumed that the emergence of two or more clusters in a set of replies to a Delphi question might indicate that experts used different information sources on this specific topic when providing their replies. The trustworthiness of such information sources need not be known a priori, making it impossible to judge which group of replies is more realistic.

Box 1. Selected Delphi statements in the survey [22] regarding the future development of CSS

Questionnaire 13: *Problems of creativity in designing expert and decision support systems.*

Question 13.2: *In the timeframe until 2025, cognitive research will allow for the formulation and implementation of active methods supporting the effectiveness of the learning process, customised to different disciplines. Please provide the cumulative probability of implementation of a widely available system with the following features:*

(d) the system measures overall user creativity (performs creativity tests in real time).

Question 13.3: *Please specify the expected number of training areas for which specialised systems enhancing learning creativity will be created. In the appropriate field please specify the areas in which these systems are needed the most (up to 10 areas), and for which of them the implementation of such systems could present a problem and why.*

The first selection of Delphi questions on the future of creativity, whose replies are presented in this section, is provided in Box 1 above. A statistical analysis of replies to the above questions is given in Tables 1, 2 and 3.

Question 13.2d in the survey is of special importance because any efficient creativity support systems (CSS) should be based on feedback measurement of their efficiency and that measurements must be taken in background, without disturbing the subject in his/her creative activity. This problem has been studied by many authors, starting from traditional questionnaire tests up to those based on EEG signal processing, cf. e.g. [1, 9, 24]. Taking this into account, the respondents seem to be rather skeptical, forecasting merely 20 % probability that such measurements will become operational within the next 10 years, while their estimates grow slightly to 30 % up to 2030.

Table 1 The results of a Delphi survey on the cumulative probability of implementation of a widely available CSS that measures overall user creativity related to learning processes in real time (Question 13.2d)

Specification of results	Estimation for 2015	Estimation for 2020	Estimation for 2025	Estimation for 2030
No. of replies	43	43	43	43
Shapiro–Wilk test	Negative	Positive	Positive	Positive
Mean value	10.8	17.229	26.571	36.571
Standard deviation	9.447	10.018	12.161	14.775
Left semideviation	5.093	6.963	12.803	15.597
Right semideviation	18.042	14.7	14.394	16.065
Median value:	5	15	20	35
1st quintile	2	5	10	15
2nd quintile	5	10	16	25
3rd quintile	10	15	30	40
4th quintile	27	32	47	60
Interquintile range	25	27	37	45
Interquartile range	12	16.25	27.5	32.5
No. of clusters of replies	1	1	3	3

All replies are analysed with equal weights
Source Delphi survey within the project SCETIST [22] (same source for Tables 2, 3, 4 and 5)

Table 2 The results of a Delphi survey on the cumulative probability of implementation of a widely available CSS that measures overall user creativity related to learning processes in real time (Question 13.2d)

Specification of results	Estimation for 2015	Estimation for 2020	Estimation for 2025	Estimation for 2030
Mean weighted value	9.608	10.299	12.226	13.945
Weighted standard deviation	5.468	8.049	13.015	14.518
Weighted left semideviation	15.896	13.02	13.947	14.945
Weighted right semideviation	9.608	10.299	12.226	13.945
Weighted median value:	5	10	20	30
1st weighted quintile	2	5	10	15
2nd weighted quintile	5	10	15	25
3rd weighted quintile	10	15	30	40
4th weighted quintile	25	35	45	55
Interquintile range (IQVR)	23	30	35	40
Interquartile range (IQR)	13	20	35	35
Number of clusters of replies > 2	No	No	Yes	Yes

Replies are weighted with combined trust/competence scores

Table 3 The results of a Delphi survey on the expected number of training areas for which specialised systems enhancing learning creativity will be created (Question 13.3)

Specification of results	Estimation for 2015	Estimation for 2020	Estimation for 2025	Estimation for 2030
Number of replies	41	42	41	41
Shapiro–Wilk test	Negative	Negative	Negative	Positive
Mean weighted value	2.545	4.618	7.424	12.576
Weighted standard deviation	1.644	3.14	5.636	10.722
Weighted left semideviation	1.303	2.497	3.522	7.422
Weighted right semideviation	2.027	3.779	7.022	14.541
Weighted median value:	2	4	7	10
1st weighted quintile	1	1	2	3
2nd weighted quintile	1	2	3	4
3rd weighted quintile	3	5	9	14
4th weighted quintile	4	8	11	20
Interquintile range IQVR	3	7	9	17
Interquartile range IQR	2	6	8	17
Number of clusters of replies	1	1	2	3

Replies are weighted with combined trust/competence scores; quintiles are rounded to the nearest integer

During the survey, the replies could be weighted according to a self-assessment of the certainty of the reply by the respondents combined with a competence coefficient assigned automatically by the Foresight Support System [16] based on the record of publications, projects and other achievements of the respondent. The same statistical characteristics as presented in Table 1, now calculated for responses weighted with combined credibility/competence coefficients, are shown in Table 2 below.

It can be observed that the values of both interquantile range indicators are usually smaller for the weighted average reply calculations presented in Table 2 than those in Table 1. This rule holds for all values except the IQR estimate for 2020. Moreover, the estimates for 2015, which were based on direct knowledge rather than on intuition and guesstimates are particularly tighter for weighted average replies. If, for another question in the survey, a converse inequality is satisfied then—as a principle—the difference between the weighted and non-weighted range indicators is relatively small. Therefore, from now on, we will present an analysis of the weighted values only.

Another question touched upon the expected number of areas for which specialised systems enhancing learning creativity will be created. The replies could be regarded as an estimation of needs, thus providing clues as regards the development of the corresponding software market sector. The replies were generally not very optimistic, with the median number of software development areas at only seven for 2025. This can be explained by the fact that research funding for such systems is still at a relatively low level in the EU, where all the respondents were from. However, a stronger surge is expected by 2030, with the median rising to 10 and the mean value approaching 13. The growth may be related to the effects of an increase of creativity-related research spending which is expected within the EU Horizon 2020 Programme.

A characteristic feature of the above results is that the mean value and median differ only slightly for all foresight horizons. Similarly, the difference between IQR and IQVR does not exceed 1. The variance of replies is relatively large, the right (optimistic) semideviation is first (2015 and 2020) almost 50 % larger than the left (pessimistic) one, but this ratio grows to roughly 2 for 2025 and 2030. This coincides with the simultaneous appearance of two clusters of replies after the year 2025. In 2030, the distribution of replies passes the Shapiro–Wilk normality test, which indicates that one of two clusters of diverging opinions is of a relatively small size compared to the dominating one spread approximately normally around the mean value.

The results presented in Tables 1 and 3 are visualised in Fig. 2.

Further results touch upon the future development of artificial creativity that was investigated in the Delphi section 13.10 and consisted of four sub-questions. The study of creativity issues from the point of view of artificial intelligence [7] is gaining in popularity compared to more traditional psychological research. The

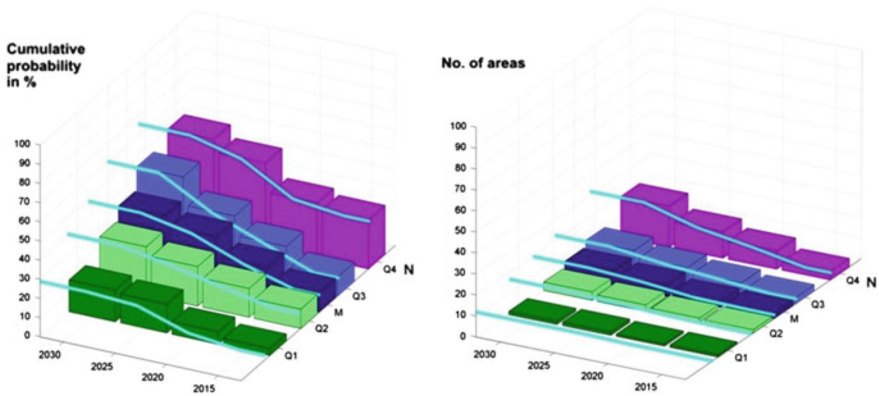


Fig. 2 The results of the Delphi survey on creativity: *left* Question 13.2d (Tab. 3), *right* Question 13.3 (Tab. 4). Q1, Q2, Q3 and Q4 denote the corresponding quintiles (at 20, 40, 60 and 80 % of responses). The results on the *left* are given as the cumulative probability of occurrence of a particular event (*left*), in % on the *vertical* axis. The average numbers of areas in the *right* figure are rounded to the nearest integer

emergence of artificial creativity, bringing new insight into the development of artificial intelligence, has been noted in scientific literature in the recent years only [10].

Box 2. A Delphi statement concerning the future development of artificial creativity [22].

Question 13.10: *Autonomous systems capable of creative activity will be developed, such that 95 % of recipients will be unable to tell the difference between an artificial system output and human artist/author output in the following areas: (a) Construction and architecture, (b) Industrial design (c) Literature (novels), (d) Other area (please specify).*

Taking into account that the replies to Questions 13.10a and 13.10b were found to be correlated with the Pearson correlation coefficient $\rho_{a,b} = 0{,}786$ for 2025, and the replies to Question 13.10d are correlated with all the remaining sub-questions ($\rho_{ad} = 0{,}526$, $\rho_{b,d} = 0{,}552$, $\rho = 0{,}687$) in Tables 4 and 5 below we will provide a

Table 4 The results of a Delphi survey on the cumulative probability of the creation of autonomous creative systems such that 95 % of recipients will be unable to tell the difference between an artificially generated, and a human creator output of construction and architectural designs (Question 13.10a)

Specification of results	Estimation for 2015	Estimation for 2020	Estimation for 2025	Estimation for 2030
Mean weighted value	6.941	13.863	22.498	31.151
Weighted standard deviation	5.764	9.432	12.191	14.438
Weighted left semideviation	3.891	7.647	11.461	16.227
Weighted right semideviation	8.072	11.397	13.134	15.711
Weighted median value:	5	10	20	30
1st weighted quintile	0	4	5	10
2nd weighted quintile	4	5	15	20
3rd weighted quintile	5	15	25	35
4th weighted quintile	15	25	40	45
Intequintile range (IQVR)	15	21	35	35
Intequartile range (IQR)	13	15	30	30
Number of clusters of replies	1	1	3	3

N = 44 replies are weighted with combined trust/competence coefficients

Table 5 The results of a Delphi survey on the cumulative probability of the creation of autonomous systems capable of creative action such that 95 % of recipients will be unable to tell the difference between artificially generated, and a human creator literary output (novels) (Question 13.10c)

Specification of results	Estimation for 2015	Estimation for 2020	Estimation for 2025	Estimation for 2030
Mean weighted value	2.094	4.763	7.95	12.887
Weighted standard deviation	2.207	4.217	7.114	9.725
Weighted left semideviation	1.594	3.168	4.844	7.393
Weighted right semideviation	3.199	5.363	10.001	13.555
Weighted median value:	1	2	5	10
1st weighted quintile	0	0	0	1
2nd weighted quintile	0	1	5	10
3rd weighted quintile	2	5	5	10
4th weighted quintile	5	15	15	35
Intequintile range (IQVR)	5	15	15	34
Intequartile range (IQR)	5	10	14	14
Number of clusters of replies	1	1	3	3

N = 44 replies are weighted with combined trust/competence coefficients

statistical analysis of the weighted replies for Questions 13.10a and 13.10c as a representative sample.

The analysis of the replies to all four questions cited in Box 2 is visualised in Fig. 3. It can be observed that around one third of experts (median value) anticipate automated architecture by 2030, while only about 10 % expect similar progress in automated literature production. The consensus among experts, as expressed by the interquintile range (Q4–Q1), is also more than twice as high as in the case of artificial architectural creativity forecasts. A salient 30 % + ‘pessimists’ group can be distinguished in all four cases. This explains why the consensus among the experts is considerably lower for all artificial creativity-related forecasts compared to the Questions 13.2d and 13.3.

Surprisingly, a better consensus was reached for ‘artificial literature’ (13.10c) than for the ‘artificial design’ (13.10a and 13.10b) for the year 2025. Observe that a group of ‘strong optimists’ may be noticed for the ‘literature’ case as well. Furthermore, it can be observed that the consensus coefficients for ‘artificial design’ rise sharply after 2020 while the corresponding values for ‘artificial literature’ for 2020 and 2025 remain similar. Among the other prospective areas for the development of ‘artificial creativity’ (Question 13.10d), the respondents mentioned

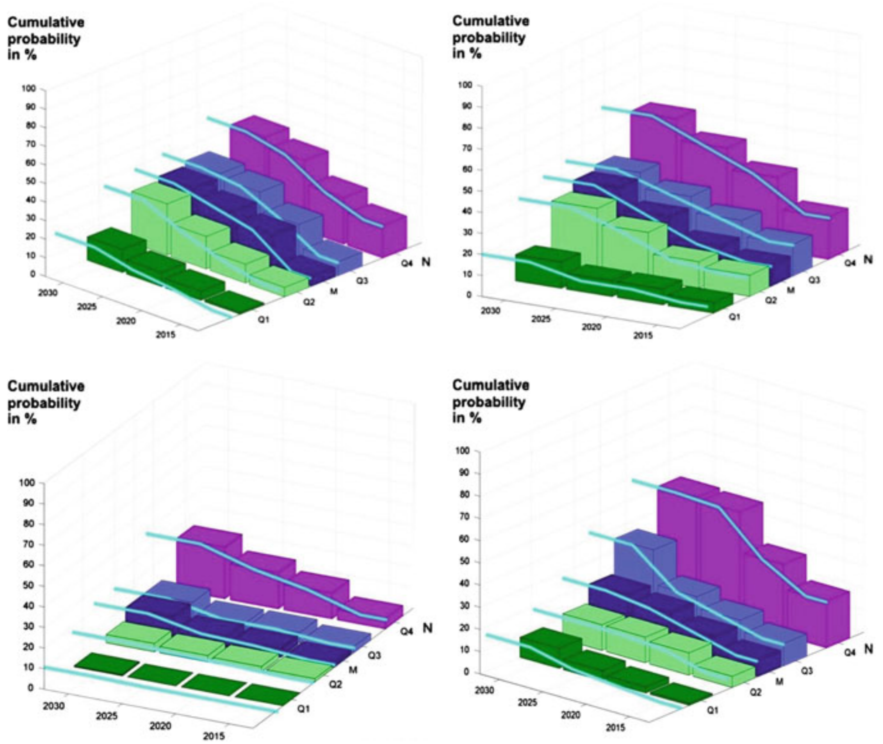


Fig. 3 The results of a Delphi survey on creativity: upper *left* Question 13.10a, *upper right* Question 13.10b, *lower left* Question 13.10c, *lower right* Question 13.10d, multi-round Delphi available at www.ict.foresight.pl. The results are given as the probability of occurrence of a particular event, in % on the *vertical* axis. Q1, Q2, Q3 and Q4 denote the corresponding quintiles

automatic music composition (7), creative decision support (five replies), including creative strategic decisions (3), automatically generated telenovels and reality shows (4), computer-created paintings and other decorative elements (3). Roughly, a quarter mentioned that the application areas of 'artificial creativity' will be diversified and hard to anticipate. For 2025, the best consensus out of the four cases analysed could be reached for 'artificial literature', with a pessimistic average value of barely 8 %.

The second round of the survey was concluded in March 2013; then it was converted to a novel multi-round tool. This idea of the survey makes it possible to contribute to research on the ICT future in an incremental way. Each newly registered expert begins from the first round and, after responding to at least 95 % of the questions on a given topic, can proceed to the next round with access to the statistics of all previous replies. The survey can be found at the web page http://www.ict.foresight.pl.

4 Conclusions

The new methodology and innovative foresight approach outlined above and in [16, 17, 20, 22] made it possible to create more reliable estimates of future development trends and scenarios of creative and intelligent technologies, as well as visualise their dynamics. The scenarios derived from an expert Delphi can again be used to re-examine the evolution principles of complex AI areas and technologies, such as cognitive creativity support systems and intelligent decision support tools. The adaptive information society evolution model [16, 17] allowed us to fuse the results of bibliometric and patentometric analysis with information elicited from experts and to characterise technological and socio-economic development in quantitative terms. Ex-post assessments of the foresight results obtained by similar but much simpler methods in 2005 [13] confirm the efficacy of the modelling methods developed and applied, as well as a good coherence of forecasts and real-life data gathered ex-post.

It should be noted that the Delphi on creativity presented in this paper constitute one of the first, if not the first, approach to exploring the future of artificial creative systems and creativity support systems in this way. Other approaches that deal with the future of creativity opt for qualitative modelling of creativity futures, either by studying autopoietic systems [4, Sect. 9] or by focussing on cognitive neuroscience research [3]. A discussion on the future design and applications of creativity support systems has started only recently, specifically including the design of group CSS [26].

More objective and quantifiable future technological characteristics have made it possible to define and recommend to decision makers appropriate policy goals and concrete measures to be implemented. The foresight outcomes indicated a growing demand for intelligent and creativity-emulating software, which would be of benefit to developers of Content Management Systems (CMS), recommenders and decision support systems (DSS). Such a growth is expected in all AI development scenarios until 2025 built in SCETIST. The main user group of the future trends and scenarios comprises innovative ICT companies seeking technological recommendations and future R&D priorities, as well as potential users of artificial intelligent and creative systems. The results of the project can also serve to inform R&D and educational institutions on the most likely in-demand areas of development of CSS for learning purposes. A well-established approach to deriving strategic recommendations from foresight outputs is based on technological roadmapping [8]. An online roadmapping methodology, tailored to the needs of innovative software companies and to the scope of foresight research, has been elaborated by the SCETIST team as well [23]. To learn more about these findings, the reader is referred to further reports and publications that can be downloaded from the project website www.ict.foresight.pl.

Acknowledgement The research presented in this paper has been conducted within the foresight project "Scenarios and Development Trends of Selected Information Society Technologies until 2025" (SCETIST) co-financed by the ERDF within the Innovative Economy Operational Programme 2006-2013, Contract No. WND-POIG.01.01.01-00-021/09.

References

1. Brookhart, S.: Assessing creativity. Educ. Leadersh. **70**(5), 28–34 (2013)
2. Daim, T.U., Rueda, G., Martin, H., Gerdsri, P.: Forecasting emerging technologies: use of bibliometrics and patent analysis. Technol. Forecast. Soc. Chang. **73**, 981–1012 (2006)
3. Gero, J.S.: Future directions for design creativity research. In: Taura, T., Nagai, Y. (eds.) Design Creativity 2010, pp. 15–22. Springer-Verlag, London (2011)
4. Iba, T.: An autopoietic systems theory for creativity. In: Riopelle, K., Gloor, P., Miller, C., Gluesing, J. (eds.): 1st Collaborative Innovation Networks Conference–COINS'2009. Procedia Social and Behavioral Sciences, vol. 2(4), pp. 6610–6625. Elsevier Science BV, Amsterdam (2010)
5. Lane, D., Pumain, D., van der Leeuw, S.E., West, G. (eds.). Complexity Perspectives in Innovation and Social Change, Springer Science + Business Media B.V. (2009)
6. Linstone, H.A., Turoff, M. (eds.): The Delphi Method. Techniques and Applications. p. 616 (1975). [Electronic version © Harold A. Linstone, Murray Turoff, 2002]
7. Pereira, C.F.: Creativity and Artificial Intelligence. A Conceptual Blending Approach. Mouton de Gruyter, Berlin, New York, p. 253 (2007)
8. Petrick, I.J., Echols, A.E.: Technology roadmapping in review: A tool for making sustainnable new product development decisions. Technol. Forecast. Soc. Chang. **71**, 81–100 (2004)
9. Plucker, J.A., Makel, M.C.: Assessment of creativity. In: Kaufman, J.C., Sternberg, R.J. (eds.) The Cambridge Handbook of Creativity, pp. 48–73. Cambridge University Press, New York (2010)
10. Schmidhuber, J.: Developmental robotics, optimal artificial curiosity, creativity, music, and the fine arts. Connect. Sci. **18**(2), 173–187 (2006)
11. Simonton, D.K.: Taking the U.S. patent office criteria seriously: a quantitative three-criterion creativity definition and its implications. Creat. Res. J. **24**(2–3), 97–106 (2012)
12. Skulimowski, A.M.J.: Decision Support Systems Based on Reference Sets. AGH University Scientific Publishers, Monographs No. 40, p.167 (1996)
13. Skulimowski, A.M.J.: Framing new member states and candidate countries information society insights. In: Compaño, R., Pascu, C. (eds.) Prospects for a Knowledge-Based Society in the New Members States and Candidate Countries, pp. 9–51. Publishing House of the Romanian Academy, Bucharest (2006)
14. Skulimowski, A.M.J.: Methods of technological roadmapping and foresight (in Polish). Chemik: Nauka-Technika-Rynek **42**(5), 197–204 (2009)
15. Skulimowski, A.M.J.: Freedom of choice and creativity in multicriteria decision making. In: Theeramunkong, T., Kunifuji, S, Nattee, C., Sornlertlamvanich, V. (eds.) Knowledge, Information, and Creativity Support Systems: KICSS 2010 Revised Selected Papers, Lecture Notes in Artificial Intelligence, vol. 6746, pp. 190–203. Springer (2011)
16. Skulimowski, A.M.J.: A foresight support system to manage knowledge on information society evolution. In: Aberer K., et al. (eds.) Social Informatics. 4th International Conference, SocInfo 2012, Lausanne, Switzerland, 5–7 Dec 2012, Proceedings. Lecture Notes in Computer Science, vol. 7710, pp. 246–259. Springer, Berlin (2012)
17. Skulimowski, A.M.J.: Discovering complex system dynamics with intelligent data retrieval tools. In: Zhang, Y., et al. (eds.) Sino-foreign-interchange Workshop on Intelligent Science and Intelligent Data Engineering IScIDE 2011, Xi'an, China, 23–26 October 2011. Lecture Notes in Computer Science, vol. 7202, pp. 614–626. Springer, Berlin (2012)
18. Skulimowski, A.M.J.: Universal intelligence, creativity, and trust in emerging global expert systems. In: Rutkowski L., et al. (eds.) Artificial Intelligence and Soft Computing. 12th International Conference, ICAISC 2013, Zakopane, Poland, 9–13 June 2013, Proceedings, Part II. Lecture Notes in Artificial Intelligence, vol. 7895, pp. 582–592. Springer, Berlin (2013)
19. Skulimowski, A.M.J.: Web-based learning in remote areas: evaluation of learning goals, scenarios and bidirectional satellite internet implementation. In: Wang, J.-F., Lau, R. (eds.)

Advances in Web-Based Learning—ICWL 2013, 12th International Conference, Kenting, Taiwan, 6–9 October 2013. Proceedings, Lecture Notes in Computer Science, vol. 8167, pp. 50–60. Springer, Berlin (2013)
20. Skulimowski, A.M.J.: Towards a new insight into technological evolution: the foresight of information society and emerging intelligent technologies. In: Skulimowski, A.M.J. (eds.) Looking into the Future of Creativity and Decision Support Systems: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems, Kraków, Poland, 7–9 November 2013, Advances in Decision Sciences and Future Studies, vol. 2, pp. 599–610. Progress & Business Publishers, Kraków (2013)
21. Skulimowski, A.M.J.: Anticipatory network models of multicriteria decision-making processes. Int. J. Sys. Sci. **45**(1), 39–59 (2014). doi:10.1080/00207721.2012.670308
22. Skulimowski, A.M.J. (ed.): Scenarios and Development Trends of Selected Information Society Technologies until 2025 (SCETIST). Final Report, Progress & Business Publishers, Kraków (2013). www.ict.foresight.pl
23. Skulimowski, A.M.J., Pukocz, P.: Enhancing creativity of strategic decision processes by technological roadmapping and foresight. In: Lee, V.C.S., Ong, K.-L. (eds.) KICSS 2012: Seventh International Conference on Knowledge, Information and Creativity Support Systems: Melbourne, Victoria, Australia, 8–10 November 2012. IEEE Computer Society. CPS Conference Publishing Services, pp. 223–230 (2012)
24. Tadeusiewicz, R.: A need of scientific reflection on the information society development. In: Bliźniuk, G., Nowak, J.S. (eds.) Information Society 2005 (in Polish), pp. 11–38. PTI, Katowice (2005)
25. Villani, D., Antonietti, A.: Measurement of creativity. In: Carayannis, E.G. (ed.) Encyclopedia of Creativity, Invention, Innovation and Entrepreneurship, pp. 1234–1238. Springer, New York (2013)
26. Voigt, M., Bergener, K.: Enhancing creativity in groups–proposition of an integrated framework for designing group creativity support systems. In: Proceedings of the 46th Hawaii International Conference on System Sciences, pp. 225–234. IEEE (2013)
27. von der Gracht, H.A.: Consensus measurement in Delphi studies: Review and implications for future quality assurance. Technol. Forecast. Soc. Chang. **79**(8), 1525–1536 (2012)
28. Wong, Ch-Y, Goh, K.-L.: Modeling the behaviour of science and technology: self-propagating growth in the diffusion process. Scientometrics **84**(3), 669–686 (2010)

Part IV
Creative Decisions

Solving a Multicriteria Decision Tree Problem Using Interactive Approach

Maciej Nowak

Abstract Making decisions involves prediction of future outcomes. In the real world, however, predictions are not known with certainty. Moreover, in serious problems multiple criteria must be taken into account. Various techniques for multicriteria decision-making problems under risk are proposed. Most of them assume that the decision process can be reduced to a single act of choice. However, real-world decision problems are often dynamic, which means that a series of choices must be made. In the paper, a new technique for a dynamic multicriteria decision-making problem under risk is proposed. The method uses a decision tree and interactive approach. A real-world example is presented to show applicability of the procedure.

Keywords Dynamic decision-making problems · Multiple criteria · Risk · Decision tree · Interactive approach

1 Introduction

Most of the real-world decision problems involve multiple criteria. The generally accepted typology of techniques used for solving them includes three types of techniques: the ones that use a priori preference information, methods exploiting a posteriori information, and the techniques in which preference information is collected in a stepwise manner [12]. The last approach assumes that the decision maker (DM) is able to provide preference information with respect to a given solution or a given set of solutions (local preference information). In each iteration a single solution (or a small subset of feasible solutions) is presented to the DM and he/she

M. Nowak (✉)
Department of Operations Research, University of Economics in Katowice, Katowice, Poland
e-mail: maciej.nowak@ue.katowice.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_23

is asked to evaluate it. The information articulated by the DM is used for generating a new solution. Procedure continues until a satisfactory solution is obtained.

Various advantages are mentioned for applying interactive techniques. First, the limited amount of preference information is required, as compared to other approaches. Second, the DM does not have to answer hypothetical questions, but is asked to evaluate well-defined solutions, that are known to exist and be feasible. It is also pointed out that the DM is closely involved in the whole problem solving process, thus obtaining more insight into the trade-offs among different criteria. Finally, it is emphasized that the DMs put much reliance in solutions generated via interactive procedures, and as a result, such solutions have better chances of being implemented.

Usually interactive techniques are applied to static decision problems, i.e., the ones in which a single decision is made. However, in many cases, the decision-making process encompasses a series of interdependent decisions that must be made at different periods of time in order to achieve an overall goal. Moreover, as the future is unknown, the results of these decisions are mostly uncertain. A decision tree is well-known tool to model and to evaluate such processes.

In this paper, a dynamic multicriteria decision-making problem under risk is considered. It can be defined as follows:

1. The decision process consists of T periods. At each period, a decision must be made. Any decision made at period t determines the characteristics of the problem at period $t + 1$.
2. Risk is taken into account. It is assumed that states of nature are defined for each period and are modeled by probabilistic distributions.
3. Multiple conflicting criteria are considered.

The aim of this work is to propose a new procedure for this problem. It combines a decision tree and interactive approach.

2 Multicriteria Decision Tree

Decision tree is a well-known and widely used tool for modeling and solving multistage decision-making problems under risk [3, 10, 21]. Through a graphical representation, even complex decision situation can be clearly presented to the DM. Three types of nodes are used in a decision tree: decision nodes (represented by squares), chance nodes (represented by circles) and final nodes (represented by dots). The branches leaving decision nodes represent decisions that can be made at this node, while the branches leaving chance nodes represent states of nature that are not controlled by the DM, but affect the decision process. The tree illustrates a multiperiod decision-making process, which starts at the initial decision node. The aim is to identify optimal decisions for each decision node.

Decision trees are used in various areas [1, 2, 7, 9, 19, 20]. As the size of the tree increases roughly exponentially with the number of variables [11], it can be successfully used only for relatively small-size problems. In this paper only such situations are considered.

Decision trees are typically used for single criterion decision problems. A multicriteria decision tree was analyzed by Haimes et al. [8], who proposed a method for generating the set of efficient solutions. Lootsma [14] combined decision tree with two cardinal methods: multiplicative AHP and SMART in order to aggregate multidimensional consequences. In [5] and [6], conditional expected value as a measure of the risk of rare events was used. More recently, Frini et al. [4] solved the multicriteria decision tree problem without generating the set of all efficient solutions. Their approach combined advantages of decomposition with the application of multicriteria decision aiding (MCDA) methods at each decision node.

An alternate concept for modeling multiperiod decision processes that was proposed recently uses hybrid anticipatory networks [18]. It assumes that that the DM takes into account the anticipated outcomes of future decision problems linked by the causal relations with the present decision problem. The problem is represented by a multigraph, where decision problems are modeled as nodes linked causally and by one or more additional anticipation relations.

In [16], an interactive technique based on INSDECM technique [15] is used to solve a multicriteria decision tree problem. Here, a modified procedure has been proposed. In contrast to the procedure presented in [16], a single solution is proposed to the DM in each iteration.

Let us assume the following notation:

T—number of periods,
$\mathbf{D}^t$—the set of decision nodes of period t (for t = 1, …, T):

$$\mathbf{D}^t = \left\{ d_1^t, \ldots, d_k^t, \ldots . d_{n_d(t)}^t \right\} \tag{1}$$

$\mathbf{D}^{T+1}$—the set of terminal nodes,
$\mathbf{A}_k^t = \left\{ a_{1(k)}^t, \ldots, a_{i(k)}^t, \ldots . a_{n_a(k)}^t \right\}$—the set of decisions (alternatives) at node d_k^t,
$\mathbf{E}_{k,i}^t = \left\{ e_{1(k,i)}^t, \ldots, e_{j(k,i)}^t, \ldots, e_{n_e(k,i)}^t \right\}$—the set of states of nature emerging from alternative $a_{i(k)}^t$.
$p_{j(k,i)}^t$—probability that $e_{j(k,i)}^t$ occurs.

For all t = 1, …, T, k = 1, …, $n_d(t)$, i = 1, …, $n_a(t, k)$ the following condition is satisfied:

$$\sum_{j=1}^{n_e(k,i)} p_{j(k,i)}^t = 1 \tag{2}$$

Let Ω_t be transition function defining the relation between indexes of two successive decision nodes k_t and k_{t+1} of periods t and $t + 1$, respectively:

$$k_{t+1} = \Omega_t(k_t, i(k_t), j(k_t, i(k_t))) \tag{3}$$

where $i(k_t)$ states for the index of the decision and $j(k_t, i(k_t))$ represents the index of the state of nature.

The aim of the analysis is to specify the strategy that should be implemented by the DM. It is defined by a decision at the initial node and decisions at all nodes that may be achieved as a result of decisions made in previous periods. Each strategy is composed of partial strategies, which consists of decisions made at the particular node of period t and decisions made in next periods. We will denote partial strategy by $s^t_{l(k)}$. As we assume that a single decision node is defined for the first period, so $s^1_{l(1)}$ denotes a strategy for the whole decision process.

Decision trees are typically limited to the optimization of a single scalar objective (expected monetary value, expected utility, etc.). In such case, the solution procedure consists of averaging out at the chance nodes and folding back at the decision nodes. At each chance node expected value is computed, while at each decision node the decision with the highest payoff value is selected. The process starts from the last period, and is repeated until the starting decision node is reached.

When multiple criteria are considered, the concept of optimality is replaced by that of efficiency. An efficient, or non-dominated, vector in multicriteria optimization cannot be improved upon in any single dimension of performance without a corresponding degradation in other dimensions. The set of efficient strategies can be identified by pairwise comparisons, and can be accelerated by folding-back-and-averaging-out procedure.

Three types of measures for evaluating strategies are considered in this paper:

- expected value: $E[X; s^1_{l(1)}]$,
- probability of event π: $P(\pi; s^1_{l(1)})$,
- conditional expected value: $E[X|\pi; s^1_{l(1)}]$.

Expected value is a common measure of performance for decision-making in the face of risk. The strategy optimizing this measure can be identified using folding-back-and-averaging-out procedure, which makes possible to eliminate inferior policies at intermediate nodes. If multiple criteria are considered and all are evaluated by expected values, the similar procedure can be applied to identify efficient strategies [8]. However, as was shown by Li [13], folding-back-and-averaging-out procedure can be used if and only if the objective function is separable and monotonic. While expected value satisfies this requirement, various risk measures do not. Conditional expected value is an example of such a function. It defines expected value of the outcome given that the magnitude of the outcome attains at least a given threshold β. In [6], a decision tree procedure using conditional expected value is proposed. This approach is used in this work.

Let $f(k)$ be the value of the criterion at terminal node d_k^{T+1}. The unconditional expected value $F^t(s_{l(k)}^t)$ for a partial strategy $s_{l(k)}^t$ can be calculated using the following recurrent formula:

$$F^t\left(s_{l(k)}^t\right) = \begin{cases} \sum_{j=1}^{n_e(k,i)} p_{j(k,i)}^t f(k') & \text{if } t = T \\ \sum_{j=1}^{n_e(k,i)} p_{j(k,i)}^t F^{t+1}\left(s_{l(k')}^{t+1}\right) & \text{otherwise} \end{cases} \quad (4)$$

where k' is specified by the transition function:

$$k' = \Omega_t(k, i(k), j(k, i(k))) \quad (5)$$

Let us now assume that the DM is interested in the probability that a particular event π occurs, which means that the process ends in any of terminal nodes for which this event takes place. Let Δ_π be the set of terminal nodes for which event π occurs. The probability that π occurs, assuming that a partial strategy $s_{l(k)}^t$ is applied, can be calculated using the following formula:

$$P_\pi^t\left(s_{l(k)}^t\right) = \begin{cases} \sum_{j=1}^{n_e(k,i)} p_{j(k,i)}^t \rho_\pi\left(d_{k'}^{T+1}\right) & \text{if } t = T \\ \sum_{j=1}^{n_e(k,i)} p_{j(k,i)}^t P_\pi^{t+1}\left(s_{l(k')}^{t+1}\right) & \text{otherwise} \end{cases} \quad (6)$$

where k' is defined by (5), and binary variable $\rho_\pi\left(d_{k'}^{T+1}\right)$ is defined as follows:

$$\rho_\pi\left(d_{k'}^{T+1}\right) = \begin{cases} 1 & \text{if } d_{k'}^{T+1} \in \Delta_\pi \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

Folding-back-and-averaging-out procedure cannot be used directly for conditional expected value. However, Frohwein and Lambert [6] showed that in such case the problem can be converted into bi-criteria one. Let $\overline{F}^1\left(s_{l(1)}^1|\pi\right)$ be the conditional expected value given that event π occurs and strategy $s_{l(1)}^1$ is used. It can be expressed as follows:

$$\overline{F}^1\left(s_{l(1)}^1|\pi\right) = \frac{\tilde{F}^1\left(s_{l(1)}^1, \pi\right)}{P_\pi^1\left(s_{l(1)}^1\right)} \quad (8)$$

where $\tilde{F}^1\left(s^1_{l(1)}, \pi\right)$ is the expected value of the variable $X\rho_\pi$, where ρ_π is defined by (7). While $\overline{F}^t\left(s^t_{l(k)}|\pi\right)$ does not satisfy conditions for folding-back-and-averaging-out procedure, both $\tilde{F}^t\left(s^t_{l(k)}|\pi\right)$ and $P^1_\pi\left(s^1_{l(1)}\right)$ do. As a result, both must be preserved and used for calculating conditional expected value. The following formula can be used for calculating $\tilde{F}^t\left(s^t_{l(k)}, \pi\right)$:

$$\tilde{F}^t\left(s^t_{l(k)}, \pi\right) = \begin{cases} \sum_{j=1}^{n_e(k,i)} p^t_{j(k,i)} f(k') \rho_\pi\left(d^{T+1}_{k'}\right) & \text{if } t = T \\ \sum_{j=1}^{n_e(k,i)} p^t_{j(k,i)} \tilde{F}^{t+1}\left(s^{t+1}_{l(k')}, \pi\right) & \text{otherwise} \end{cases} \tag{9}$$

where (5) is used to identify *k'* and (7) for calculating $\rho_\pi\left(d^{T+1}_{k'}\right)$.

The set of efficient strategies can be identified by pairwise comparisons. This process can be accelerated by folding-back-and-averaging-out approach. As noted above, conditional expected value must be replaced by two criteria.

To simplify the description of the procedure, we will assume here, that all criteria (expected value, probability measures, conditional expected value) are maximized. Then, a partial strategy $s^t_{l(k)}$ dominates a partial strategy $s^t_{l'(k)}$ if for each expected value criterion the following condition is satisfied:

$$F^t\left(s^t_{l(k)}\right) \geq F^t\left(s^t_{l'(k)}\right) \tag{10}$$

In the case of probability criterion, the condition is as follows:

$$P^t_\pi\left(s^t_{l(k)}\right) \geq P^t_\pi\left(s^t_{l'(k)}\right) \tag{11}$$

Finally, for conditional expected value criterion the following inequalities must be fulfilled:

$$\tilde{F}^t\left(s^t_{l(k)}\right) \geq \tilde{F}^t\left(s^t_{l'(k)}\right) \text{ and } P^t_\pi\left(s^t_{l(k)}\right) \leq P^t_\pi\left(s^t_{l'(k)}\right) \tag{12}$$

Additionally, at least one condition should be satisfied as a strict inequality. Conditions (10)–(12) must be used for periods *T* to 2. In the first period, formula (8) can be used for calculating conditional expected value and constraints (12) can be replaced by the following condition:

$$\overline{F}^t\left(s^t_{l(k)}|\pi\right) \geq \overline{F}^t\left(s^t_{l'(k)}|\pi\right) \tag{13}$$

The procedure for identifying efficient strategies consists of the following steps:

1. Start from the last period: $t = T$; identify partial efficient strategies for all decision nodes of period T.
2. Go to the previous period: $t = t - 1$.
3. For each decision node of period t, identify strategies satisfying the necessary condition for efficiency—consider efficient partial strategies for all decision nodes of period $t + 1$.
4. For each decision node of period t identify strategies satisfying the sufficient condition for efficiency—compare strategies pairwisely using formulas (10)–(12) in order to eliminate the ones that are dominated by any other.
5. If $t > 1$—go to 2.
6. If conditional expected value is used for evaluating strategies, use (8) to calculate its value and compare strategies pairwise using formulas (10)–(11) and (13) in order to identify efficient strategies.

In steps 3 and 6 partial strategies consisted of the decisions made for node d_k^t and all combinations of efficient partial strategies identified for nodes achieved from d_k^t are considered. The efficient partial strategies identified for the decision node of period 1 are the efficient strategies for the whole decision process.

3 Interactive Procedure

As the number of efficient strategies may be large, the DM may have a problem with selection of the final solution. In this paper we propose to use interactive approach. The method is a modified version of the procedure presented in [16]. At each iteration a single proposal is proposed to the DM. It is the one that is closest to the ideal solution. Additionally, the matrix of achievements is presented. It consists of two rows grouping of the worst (pessimistic), and the best (optimistic) values of criteria attainable independently within the current set of strategies. The DM is asked whether he/she is satisfied with the proposal. If the answer is "*yes,*" the procedure ends, otherwise, the DM is asked to express his/her preferences defining the values, that the criteria should achieve, or at least indicating the criterion, which should be improved.

Let $\mathbf{S}^{(q)}$ be the set of strategies analyzed in iteration q. By $g_m(s_i)$ we denote the value of m criterion for strategy s_i. Matrix of achievements $\mathbf{P}^{(q)}$ is as follows:

$$\mathbf{P}^{(q)} = \begin{bmatrix} \underline{g}_1^{(q)} & \cdots & \underline{g}_m^{(q)} & \cdots & \underline{g}_M^{(q)} \\ \overline{g}_1^{(q)} & \cdots & \overline{g}_m^{(q)} & \cdots & \overline{g}_M^{(q)} \end{bmatrix} \tag{14}$$

where: $\underline{g}_m^{(q)}$ is the worst and $\overline{g}_m^{(q)}$ the best value of m-th criterion attainable within the set of strategies considered in iteration q:

$$\underline{g}_m^{(q)} = \min_{s_i \in \mathbf{S}^{(q)}} \{g_m(s_i)\} \tag{15}$$

$$\overline{g}_m^{(q)} = \max_{s_i \in \mathbf{S}^{(q)}} \{g_m(s_i)\} \tag{16}$$

At the first iteration $\mathbf{S}^{(1)}$ consists of all efficient strategies.

Each iteration of interactive procedure works as follows:

1. Identify matrix of achievements $\mathbf{P}^{(q)}$.
2. Identify the strategy to be proposed to the DM:
   (a) compute weights of criteria using the following formula:
   $$w_m = \frac{1}{r_m}\left[\sum_{j=1}^{M}\frac{1}{r_j}\right]^{-1} \tag{17}$$
   where:
   $$r_m = \overline{g}_m^{(q)} - \underline{g}_m^{(q)} \tag{18}$$
   (b) for each strategy $s_i \in \mathbf{S}^{(q)}$ compute the overall evaluation e_i:
   $$e_i = \sum_{m=1}^{M} w_m g_m(s_i) \tag{19}$$
   (c) identify strategy maximizing the overall evaluation.
3. Present the results (matrix of achievements and the proposed strategy) to the DM and ask whether he/she is satisfied with the proposal. If the answer is '*yes,*' end the procedure.
4. Ask the DM whether he/she would like to define aspiration levels for criteria. If the answer is no—go to (6).
5. Ask the DM to specify aspiration levels $\tilde{g}_m^{(l)}$ for $m = 1, \ldots, M$. Identify $\mathbf{S}^{(q+1)}$—the set of strategies satisfying DM's requirements. If $\mathbf{S}^{(l+1)} = \varnothing$—notify the DM and go to (3), otherwise go to the next iteration.
6. Ask the DM to indicate the index m of the criterion, which should be improved. Identify $\mathbf{S}^{(q+1)}$—the set of strategies for which the value of the m-th criterion exceeds $g_m(s_p)$.

4 Numerical Example

The application presented here describes a real problem that was analyzed by a company providing solutions for the railway industry. The problem concerns decisions made when the company considered entering a new market. It was possible to operate as a general contractor or cooperate with a local company. Four objectives were considered: (1) to maximize the probability of success, (2) to maximize profit margin generated when the offer is accepted, (3) to minimize the cost of preparing a bid if the offer is not accepted, and (4) to maximize the evaluation describing the strategic fit.

Because of the lack of place, we omit here the detailed description of the problem, which can be found in [17]. The decision tree describing the problem is presented on Fig. 1. We assume that the process is successful if the company decides to submit an offer and the proposal is accepted. In our decision tree, success is represented by terminal nodes h1 – r1 (upper branches emanating from final chance nodes h – r). The opposite situation is represented by terminal nodes h2 – r2 (lower branches emanating form final chance nodes h – r representing the defeat in tender) and by terminal nodes that are reached as a result of decisions to give up the offer submission (3A, 6C, 9B, 10B, 12A, 15B). Thus, to evaluate strategies with respect to the first criterion we must calculate probability that the process will reach any of "success" terminal nodes.

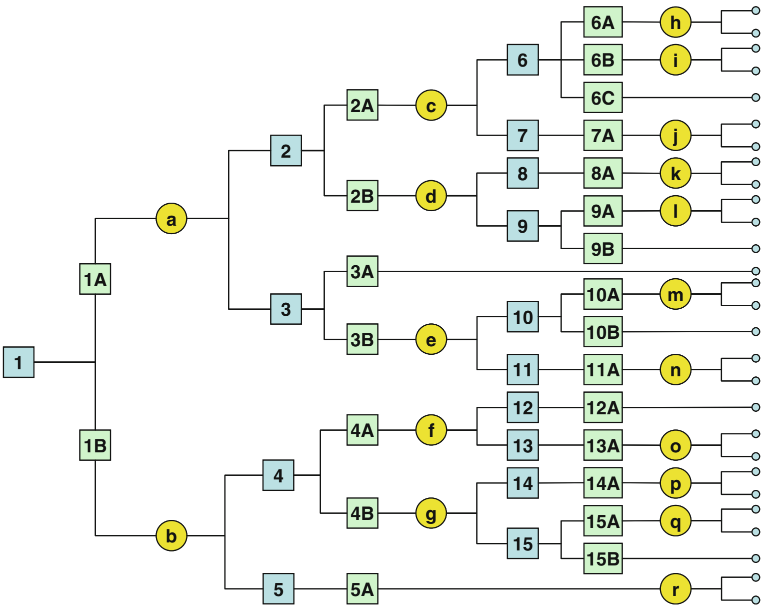


Fig. 1 Decision tree of the problem

Table 1 Probabilities of the states of nature

State of nature	Probability	State of nature	Probability	State of nature	Probability	State of nature	Probability
a1	0.7	c1	0.6	e1	0.4	g1	0.3
a2	0.3	c2	0.4	e2	0.6	g2	0.7
b1	0.6	d1	0.6	f1	0.6	h1 … r1	0.6
b2	0.4	d2	0.4	f2	0.4	h2 … r2	0.4

As the company analyzes separately financial results for the "success" and "defeat," we use conditional expected value for evaluating strategies with respect to objectives (2) and (3). Finally, each final state is evaluated with respect to the last criterion using 4 point scale, where 0 means that the company is not successful in tender, 1—the company implements the project with a local partner providing part of the equipment, 2—the company executes the project with a local partner employed for completing a part of installation work only, 3—the company implements the project as a general contractor. Table 1 describes probabilities of states of nature, while Table 2: profits, costs and qualitative criterion evaluations for each terminal node.

The total number of strategies is 18. An example of strategy is 1A/2A/6A/7A/3A. It means that decision 1A should be made at decision node 1. Then, if the first state of nature realizes at chance node *a*, the decision 2A should be made at decision node 2. After that either decision 6A or 7A should be made depending on the state of nature that will realize at chance node *c*. Finally, if the

Table 2 Values of profit margin and strategic fit criterion

Final decision/state of nature	Profit/cost	Strategic fit	Final decision/state of nature	Profit/cost	Strategic fit
6A/h1	634,733	3	10A/m2	−46,400	0
6A/h2	−46,233	0	10B	−34,333	0
6B/i1	800,867	1	11A/n1	744,667	3
6B/i2	−34,500	0	11A/n2	−46,750	0
6C	−27,867	0	12A	−39,220	0
7A/j1	870,333	1	13A/o1	694,340	3
7A/j2	−34,783	0	13A/o2	−46,700	0
8A/k1	819,467	2	14A/p1	750,467	2
8A/k2	−34,730	0	14A/p2	−46,733	0
9A/l1	760,567	2	15A/q1	710,833	2
9A/l2	−46,467	0	15A/q2	−46,033	0
9B	−39,333	0	15B	−39,167	0
3A	−27,200	0	5A/r1	756,000	3
10A/m1	694,167	3	5A/r2	−34,221	0

Table 3 Values of criteria for decisions analyzed in node 6

Decision	Objective (1)	Objective (2)		Objective (3)		Objective (4)	
	Probability of success	$\tilde{F}^3$	P_π^3	$\tilde{F}^3$	P_π^3	Evaluation equal to 3	Evaluation at least 2
6A	0.6	380,840	0.6	18,493	0.4	0.6	0.6
6B	0.6	480,520	0.6	13,800	0.4	0.6	0.6
6C	0.0	0	0.0	27,867	1.0	0.0	0.0
	max	*max*	*min*	*min*	*max*	*max*	*max*

second state of nature realizes at chance node *a*, the decision 3A should be made at the decision node 3.

Since we use conditional expected values for evaluating strategies with respect to financial objectives (maximization of profit margin and minimization of the cost of preparing the bid), each of them must be converted into two criteria in order to use folding-back-and-averaging-out approach. As to the qualitative criterion, the DM decided to analyze two characteristics: the probability, that the evaluation with respect to this criterion is exactly three, and the probability, that this evaluation is at least 2. In order to identify efficient strategies, we start from the last period. For example, three decisions are considered at the decision node 6: 6A, 6B, 6C. Table 3 presents criteria values for these decisions.

It can be noticed that the decision 6A is dominated by the decision 6B. The similar analysis is conducted for other decision nodes of the last period. Then the process goes to the second period and values of criteria are averaged according to formulas presented in the previous section. In period 1, the conditional expected values for objectives (2) and (3) are calculated. For example, conditional expected profit margin assuming that the company wins the tender for strategy 1A/2A/6B/7A/3A is calculated as follows:

$$\tilde{F}^1\left(s^1_{l(1)}, \pi\right) = 348,034 \quad P^1_\pi\left(s^1_{l(1)}\right) = 0,42$$

$$\overline{F}^1\left(s^1_{l(1)}|\pi\right) = \frac{\tilde{F}^1\left(s^1_{l(1)}, \pi\right)}{P^1_\pi\left(s^1_{l(1)}\right)} = \frac{348,034}{0,42} = 828,653$$

Finally, formulas (10), (11) and (13) are used in order to identify efficient strategies, which are as follows:

• s_1: 1A/2A/6B/7A/3A	• s_6: 1A/2B/8A/9A/3A
• s_2: 1A/2A/6B/7A/3B/10A/11A	• s_7: 1A/2B/8A/9A/3B/10A/11A
• s_3: 1A/2A/6B/7A/3B/10B/11A	• s_8: 1A/2B/8A/9A/3B/10B/11A
• s_4: 1A/2A/6C/7A/3A	• s_9: 1B/4B/14A/15A/5A
• s_5: 1A/2A/6C/7A/3B/10B/11A	

Table 4 Evaluations of efficient strategies

Strategy	Probability of success (*max*)	Profit margin in case of success (*max*)	Cost of preparing a bid in case of defeat (*min*)	Evaluation with respect to strategic fit—probability (*max*)	
				Equal to 3	At least 2
s_1	0.420	828,653	30,779	0.252	0.252
s_2	0.600	802,797	38,212	0.432	0.432
s_3	0.528	817,611	36,393	0.360	0.360
s_4	0.168	870,333	28,557	0.000	0.000
s_5	0.276	832,898	31,886	0.108	0.108
s_6	0.420	795,907	33,102	0.000	0.420
s_7	0.600	779,875	41,580	0.180	0.600
s_8	0.528	791,562	39,248	0.108	0.528
s_9	0.600	736,034	41,434	0.240	0.600

The evaluations of efficient strategies are presented in Table 4. An example of the dialog with the DM to identify the final solution is as follows:
Iteration 1:

1. The matrix of achievements and the proposal solution are identified (Table 5). The results are presented to the DM.
2. The DM is not satisfied with the proposal. He states that the cost of preparing bid in case of defeat is too high. As he is not able to specify the aspiration level for this criterion, he just asks to consider the strategies, for which the cost is lower than 38,212.
3. As five strategies satisfy the DM's requirements, the procedure goes to iteration 2.

Table 5 Data presented to the DM in iteration 1

Value	Probability of success (*max*)	Profit margin in case of success (*max*)	Cost of preparing a bid in case of defeat (*min*)	Evaluation with respect to strategic fit—probability (*max*)	
				Equal to 3	At least 2
Pessimistic	0.168	736,034	41,580	0.000	0.000
Optimistic	0.600	870,333	28,557	0.432	0.600
Proposal (s_2)	0.600	802,797	38,212	0.432	0.432

Table 6 Data presented to the DM in iteration 2

Value	Probability of success	Profit margin in case of success	Cost of preparing a bid in case of defeat	Evaluation with respect to strategic fit (probability)	
				Equal to 3	At least 2
Pessimistic	0.168	795,907	36,393	0.000	0.000
Optimistic	0.528	870,333	28,557	0.360	0.420
Proposal (s_1)	0.420	828,653	30,779	0.252	0.252

The procedure iterates in the same way. In iteration 2, the DM declares that he is satisfied with the proposal (Table 6).

5 Conclusions

Interactive approach is the leading methodology in multicriteria decision-making. Several motivations have been mentioned for implementing this approach. It is usually pointed out that limited amount of a priori preference information is required from the DM as compared to other techniques. The interactive procedure may be considered as a learning process. Observing the results of succeeding iterations of the procedure, the DM extends his/her knowledge of the decision problem. On the other hand, as the DM actively participates in all phases of problem solving procedure, he/she puts much reliance on the final solution.

In this work, a new interactive procedure for a dynamic decision-making problem under risk has been proposed. The method uses a decision tree and interactive technique.

In future work, the DM's attitude to risk will be taken into account by applying stochastic dominance rules. The author plans also to propose a modified technique for larger problems. Applications in various areas, like project portfolio management, innovation management and supply chain management will also be considered.

Acknowledgments This research was supported by the Polish Ministry of Science and Higher Education in years 2010–2013 as a research project no NN 111 267,138.

References

1. Chiu, L., Gear, T.E.: An application and case history of a dynamic R&D Portfolio selection model. IEEE Trans. Eng. Manage. **26**, 2–7 (1979)
2. Corner, J.L., Kirkwood, C.W.: Decision analysis applications in the operations research literature. 1970–1989. Oper. Res. **39**, 206–219 (1991)
3. Covaliu, Z., Oliver, R.M.: Representation and solution of decision problems using sequential decision diagrams. Manage. Sci. **41**, 1860–1881 (1995)
4. Frini, A., Guitouni, A., Martel, J.M.: A general decomposition approach for multi-criteria decision trees. Eur. J. Oper. Res. **220**, 452–460 (2012)
5. Frohwein, H.I., Haimes, Y.Y., Lambert, J.H.: Risk of extreme events in multiobjective decision trees. Part 2: Rare events. Risk Anal. **20**, 125–134 (2000)
6. Frohwein, H.I., Lambert, J.H.: Risk of extreme events in multiobjective decision trees. Part 1: Severe events. Risk Anal. **20**, 113–123 (2000)
7. Granot, D., Zuckerman, D.: Optimal sequencing and resource allocation in research and development projects. Manage. Sci. **37**, 140–156 (1991)
8. Haimes, Y., Li, D., Tulsiani, V.: Multiobjective decision tree method. Risk Anal. **10**, 111–129 (1990)
9. Hess, S.W.: Swinging on the branch of a tree: project selection applications. Interfaces **23**, 5–12 (1993)
10. Keeney, R.L., Raiffa, H.: Decisions with Multiple Objectives: Preferences and Value Tradeoffs. Wiley, New York (1976)
11. Kirkwood, C.W.: An algebraic approach to formulating and solving large models for sequential decisions under uncertainty. Manage. Sci. **39**, 900–913 (1993)
12. Korhonen, P.: Interactive methods. In: Figueira, J., Greco, S., Ehrgott, M. (eds.) Multiple Criteria Decision Analysis: State of the Art Surveys, pp. 641–665. Springer, Heidelberg (2005)
13. Li, D.: Multiple objectives and non-separability in stochastic dynamic programming. Int. J. Syst. Sci. **21**, 933–950 (1990)
14. Lootsma, F.A.: Multicriteria decision analysis in a decision tree. Eur. J. Oper. Res. **101**, 442–451 (1997)
15. Nowak, M.: INSDECM—an interactive procedure for discrete stochastic multicriteria decision making problems. Eur. J. Oper. Res. **175**, 1413–1430 (2006)
16. Nowak, M.: An interactive procedure for multiple criteria decision tree. In: Skulimowski A. (ed.) Looking into the Future of Creativity and Decision Support Systems. Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems. Progress and Business Publishers, Kraków, pp. 461–472 (2013)
17. Nowak, B., Nowak, M.: Multicriteria decision aiding in project planning using decision tree and simulation. In: Trzaskalik, T., Wachowicz, T. (eds.) Multiple Criteria Decision Making ‘10-11, pp. 163–187. University of Economics Press, Katowice (2011)
18. Skulimowski, A.: Anticipatory network models of multicriteria decision-making processes. Int. J. Sys. Sci. **45**, 39–59 (2014)
19. Stonebraker, J.S., Kirkwood, C.W.: Formulating and solving sequential decision analysis models with continuous variables. IEEE Trans. Eng. Manage. **44**, 43–53 (1997)
20. Thomas, H.: Decision analysis and strategic management of research and development: a comparison between applications in electronics and ethical pharmaceuticals. R&D Manage. **15**, 3–22 (1985)
21. Von Winterfeld, D., Edwards, W.: Decision Analysis and Behavioral Research. Cambridge University Press, Cambridge (1986)

Strategic Planning Optimization Using Tabu Search Algorithm

Wojciech Chmiel, Piotr Kadłuczka, Joanna Kwiecień, Bogusław Filipowicz and Przemysław Pukocz

Abstract In this paper, we introduce a method for the optimization of strategic tasks using an approximation algorithm. We propose a mathematical model and suboptimal algorithm for solving this problem near optimally. An overall motivation for the development of the proposed algorithm is discussed with a detailed efficiency analysis. From an end users' perspective the proposed method has practical applications because it can be used in several applied situations.

Keywords Discrete optimization · Strategic planning · Artificial autonomous decision systems · Approximation algorithms

1 Introduction

The scheduling of strategic tasks can be embedded in creative roadmapping processes of NPD (New Product Development) type [8] for a multicriteria model of creative NPD problem solving. Organizational adaptation of a unit to perform tasks is an important aspect of strategic planning. If their way of realization is dependent on

---

An earlier version of this paper was presented at the International Conference KICSS' 2013 [6].

W. Chmiel (✉) · P. Kadłuczka · J. Kwiecień · B. Filipowicz · P. Pukocz
Department of Automatics and Biomedical Engineering,
AGH University of Science and Technology, Kraków, Poland
e-mail: wch@agh.edu.pl

P. Kadłuczka
e-mail: pkad@agh.edu.pl

J. Kwiecień
e-mail: kwiecien@agh.edu.pl

B. Filipowicz
e-mail: fil@agh.edu.pl

P. Pukocz
e-mail: pukocz@agh.edu.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_24

the implemented procedures (legislation), they restrict greatly the freedom of design solutions. In the case of complex units, specialized departments are created, which are responsible for homogeneous subtasks, often referred to as operations or activities. Considering the sequence of their realization in a such defined structure, we can apply the models of scheduling theory. They provide a wide range of choice for the criterion function, constraints on resources, priorities, defining the ordering relationships between the operations, the critical deadlines and the efficiency of departments/teams. Due to constraints, the tasks may be interruptible or uninterruptible, and their processing may be serial or parallel.

In the presented case a company carries out a strategic plan—a specific set of objectives, consisting of strategic tasks. Departments, having allocated specific resources, are specialized to perform the subtasks.

The realization of the strategic plan assumes that

- the processing time of subtasks in the company departments is given,
- subtasks can not be interrupted, executed only once,
- at the same time, each department of the company can process only one subtask,
- a subtask is executed immediately when the department is free, otherwise the subtask waits for the end of the previous subtask,
- a strategic plan ends after performing the last subtask,
- the sequence in which subtasks are to be processed is determined (it is the same for each strategic plan).

The optimization problem is to find a specific sequence of strategic tasks so as to minimize the final execution time (called *makespan*). The above assumptions allow us to treat the issue as a flow-shop scheduling problem. In the case where each job has to go through all the departments in a set order, we have the flow-shop scheduling problem. When the processing order of tasks by departments is determined, but can vary, we deal with the job shop scheduling problem. In the open shop model, the operations of a task are performed in any order [3].

2 Roadmapping Model

The construction of the strategic plan in a dynamic and highly competitive environment requires an efficient, flexible and dynamic decision-making process. Roadmapping includes the development of rules for decision making that optimize criteria relevant to the implementation process and production. Depending on how the decision problem is formulated, these rules may take the form of a strategic plan, a list of priorities or a business plan related to a specific technology or product.

The roadmapping diagram is a structure of layers, corresponding to external areas of activity or influence (markets, technologies, legislation, products and public tasks). The new product development roadmapping diagram building involves selecting objects for individual layers based on significant criteria, from the layer (or layers) describing the organization. Relations between objects are defined and encoded with the selection of objects. Inter-layer relationships are usually temporal relationships. They describe the evolution of the objects of each layer using information on development trends and scenarios. Creating a strategic plan for the acquisition and implementation of technologies for the production processes is an example, when an organization should apply roadmapping. For each of these alternatives of products and technologies, and their allowable combination, or specific environmental scenarios, the optimization of decisions on expenditure for implementing various technologies is done. It leads to solving discrete optimization problems as defined hereafter in this section, where the parameters are the characteristics of the technologies and products (purchase price of license, unit costs of implementation, the lifetime of the technology, product range and the possibility of reselling their own technology described by the parameters of real options). Furthermore, roadmapping allows us to take into account the dependence of technology and products from legislative conditions (in the form of constraints), especially related to environmental protection (strategies and scenarios to adapt products to the circumstances of legislation and standards). The following optimization problem, a special case of the flow-shop scheduling problem (FSP), can be formulated during the analysis of roadmapping diagram.

Let the set $\bar{Z} = \{Z_1, Z_2, Z_3, \ldots, Z_r\}$ be the set of r strategic plans, where every plan Z_i is a subset of set $\bar{J}$ (jobs) $Zi = \{J_i\}_{J_i \in \bar{J}}$:

- $\bar{J} = \{J_1, J_2, \ldots, J_k\}$ is the set of the jobs defined by an expert. For each job the profit function is specified. The value of the profit depends on the job execution order in the strategic plan.
- Strategic plan Z_i Zi is a subset of the jobs defined by an expert on the basis of company policy. In the generalized case, all possible subsets of the jobs from $\bar{J}$ can be considered.
- Every job $J_j = \{o_{j1}, o_{j2}, \ldots, o_{jm}\}$ contains m operations corresponding to the job execution on the set of m machines (company branches). The jobs are executed in the predefined order (researches, technologies, productions, etc.).
- Each machine (company branches) can execute at most one operation (research, technology, etc.) at a time.
- The operation execution time $t_{jk} \geqslant 0$ depends on the job order execution. Some operations from different jobs may overlap or may be dependent.
- For each of the strategic plans Z_i where $i = 1, \ldots, r$, the optimal sequence of jobs which minimize C_{max} (maximal makespan) is searched. Additionally, the job sequence meets the time constraint $C_{max} \leqslant \bar{T}$.
- The goal is to find the strategic plan Z^* (and its job sequence), which maximizes the profit that is described by the equation (1) and meets the constrains (predefined order and the execution time limit $\bar{T}$).

Let us define $S(Z)$ as the set of all possible permutations of jobs in the particular strategic plan Z. Let us define the following maximization problem:

$$f(Z^*) = \max_{Z \in \bar{Z}} \max_{s \in S(Z)} \left\{ \alpha f_i(\bar{T} - C_{max}) + \gamma f_{pref}(s) - \beta f_p(s) \right\} \tag{1}$$

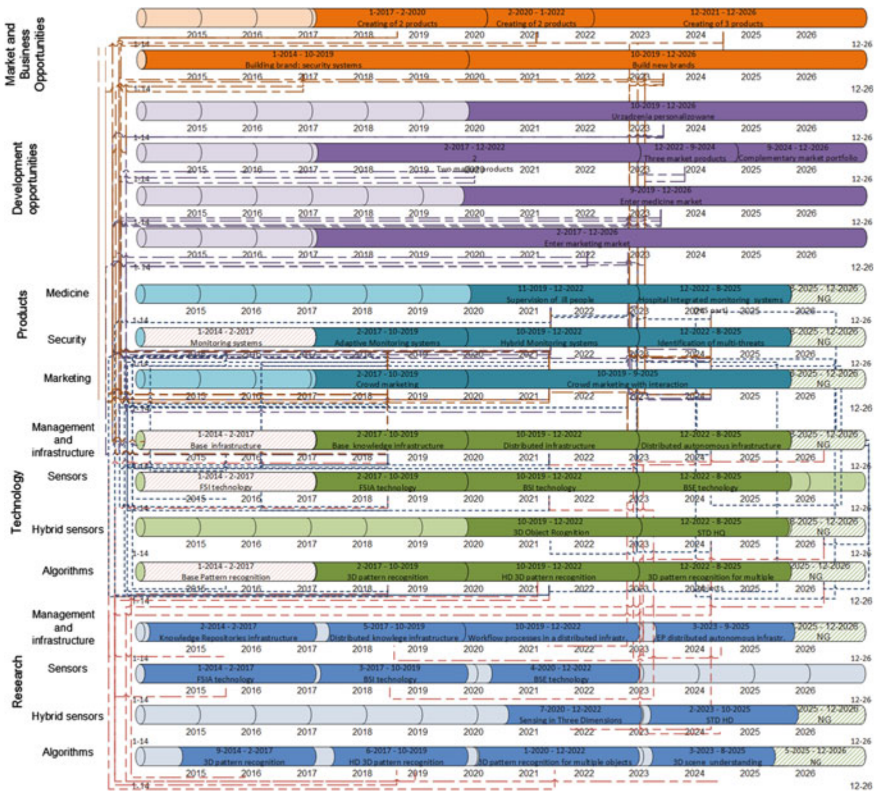


Fig. 1 An example of the roadmapping diagram

where

- α, β, γ—scalars describing preferences (set by the expert),
- $f_i(\bar{T} - C_{max})$—defines income from shortening the time of execution of the strategic plan,
- $f_p(s)$—penalty function for violation of constraints (sequence Z must satisfy a predefined partial order of the jobs and the execution time of the job sequence meets the time constraint $C_{max} \leqslant \bar{T}$),
- $f_{pref}(s)$—a certain function of technology and strategy preferences defined by experts.

Roadmapping solutions obtained using a tabu search algorithm have a job sequence $Z_i = (J_1, J_2, J_n)$, where Z_i can describe the set of decisions related to the development of any technology. It can be interpreted as a strategic plan, operational plan or implementation plan that describes a set of decisions that lead to achieving the goal (see Fig. 1). This plan can be completed with the degree of achievement of strategic objectives identified in the diagram as the states of the highlighted objects.

3 Tabu Search Algorithm

This section presents the search algorithm known as TS, which is a metaheuristics based on a local search algorithm. This method, which has been developed by Glover [5], was used to solve many *NP-hard* combinatorial problems. The essence of the method is non-monotonic, directed search of the solution space using a prohibition mechanism. The elements of the algorithm are a move that defines the neighbourhood of the current solution, short-term memory (STM), intermediate-term memory (ITM), long-term memory (LTM) and the criteria of aspiration and termination [2, 4] (see Fig. 6). The used memory can store the quasi-optimal solution, the perspective solutions with the purpose of realizing the returns, forbidden solutions, or attributes of forbidden moves and the frequency of occurrences of specific moves in order to diversify them. The main purpose of the tabu list is the elimination of the computing cycles and an effective search of the solution space, coordinated by a strategy based on the mechanisms of short-, medium- and long-term memory (STM, ITM, LTM). It stores the history of the moves that led to the current solutions in the last iterations. The main objective is to avoid the selection of the move operator which leads to the oscillations around a given solution or return to the previously explored subspace. Parameters such as the length of the tabu list, the number of iterations in which a prohibition is applicable and the choice of saved move attributes determine the effectiveness of a prohibition mechanism.

They define the intensity of the prohibition mechanism, as the cardinality of the set of solutions, which is excluded from the current neighbourhood (in Fig. 2 it is marked as a part of the neighbourhood).

The aspiration criterion is an integral component of the prohibition mechanism. It is a set of conditions determining the situations in which a prohibition is rejected [1].

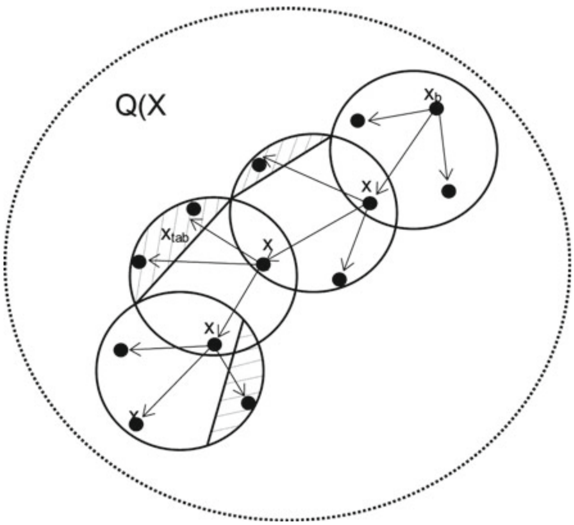


Fig. 2 Solution search using the prohibition mechanism in *TS* algorithm

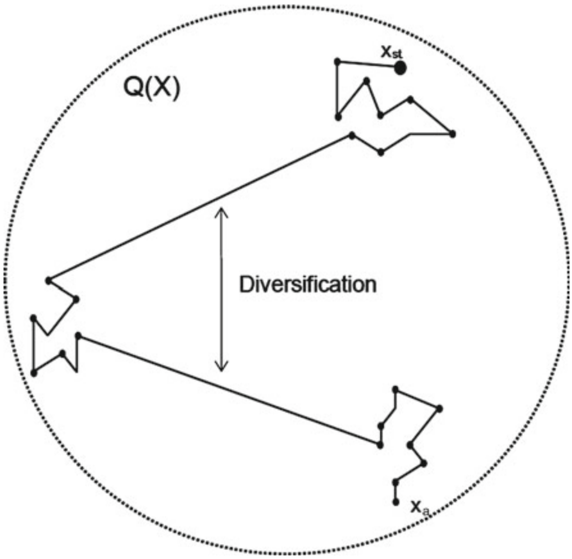


Fig. 3 Diversification mechanism in the exploration of the solution space

This allows performing forbidden moves which are attractive in further exploration. The aspiration criterion may use the conditions related to the following:

- value of the objective function of the current solution,
- value of the objective function of quasi-optimal solution,
- fulfilment of restrictions—a measure of inadmissibility solutions,
- measure of distance solutions,
- history of the algorithm execution.

Additionally, strategies for the diversification and intensification of the search process are introduced in *TS* algorithm. In a diversification approach (see Fig. 3) the solutions in the subspace, which has been sparsely explored, are preferred. Frequency-based memory (kind of long-term memory—LTM) is an instrument that records the statistical information about the previous achievement of the tabu search algorithm, such as counting the moves with certain attributes. The restart of the algorithm and the construction of the other initial solutions in the separate areas of the global solution space is another way to achieve diversification. Intensification is the exploitation of the promising areas of the solution space. It is realized by returning to the saved in the medium-term memory (in MTM) solutions obtained in phase I (see Fig. 4).

Based on the search history, a new tabu list is created. The best found solutions, i.e. those whose value of the objective function is within a specified range—derived from the current suboptimal solutions, are placed in the MTM. The distance of the successive solutions measured by the number of iterations must be greater than the

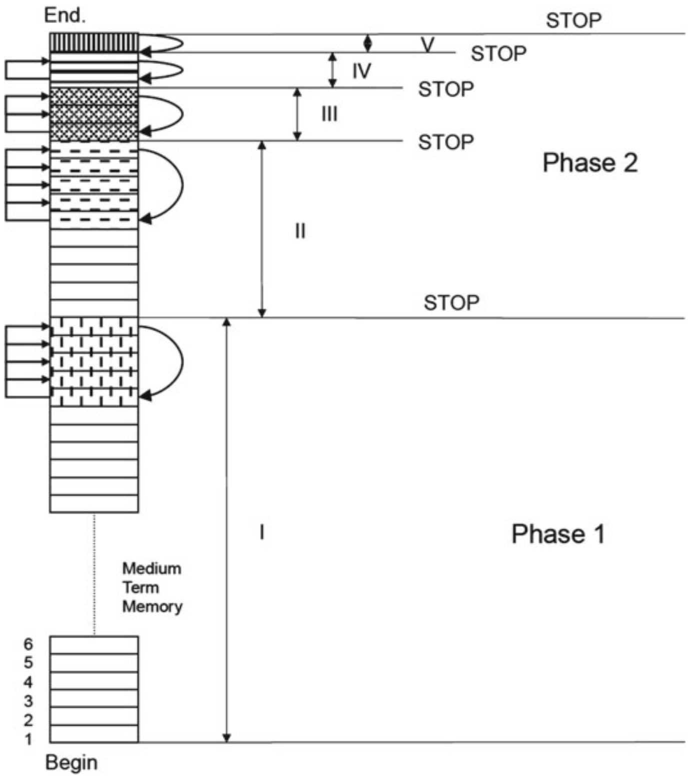


Fig. 4 Intensification mechanism in the exploration of the solution space—return to the previous solution

predetermined value, and in the case of nearby solutions, it is only one better which is saved. After saving solutions in MTM, attributes of the next moves are stored. On their basis, a new tabu list is created when the return to the solutions of the MTM is realized (see Fig. 5). It forces another sequence of the moves in the neighbourhood of promising solutions. The mechanism of the medium-term memory extracts the prospective and mutually distant areas of the solution space. During the implementation of the intensification mechanism in Phase II, the next solutions that meet the specified criteria are still added to the MTM. The algorithm will return to them after using the solutions of Phase I. The number of iterations of the algorithm in Phase II is dependent on the value of parameters. For the large number of solutions recorded in MTM, algorithm time execution significantly increases (Fig. 6).

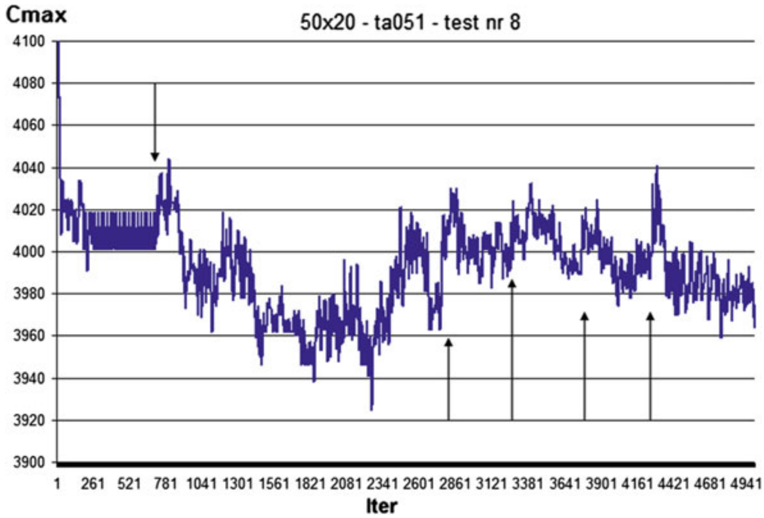


Fig. 5 Execution of *TS** algorithm for the test task `ta051` (50 jobs, 20 machines)—dependence of the objective function of the current solution on the number of iterations. *Arrows* mark the return to the solutions of MTM

4 Experiments

The tabu search algorithm was implemented for the problem described above and tested on the set of instances of size from 20 × 5 to 100 × 20 ((*number of subtasks*) × (*number of departments*)). It is worth mentioning that the problems with more than two departments (or three in some assumptions) are *NP-hard*. Due to the lack of real data (processing time of the company subtasks by departments obtained from the workflow logs) in experiments 27 the standard Taillard's test instances were used [16]. The C_{max} criterion was used, instead of the criterion function defined by the equation (1). In Table 1 the results of two versions of the algorithms were shown—the standard mechanism (*TS*) and the one using intensification (*TS**). For every test instances the best (C_{max}) and average for twelve tests ($\overline{C_{max}}$) objective function was shown [12]. For all the tests the same value of the algorithm control parameters was used.

In Table 1 below we provide the following data:

- C_{max}—the best of the makespan values (the makespan is the total length of the schedule—when all the jobs have finished processing),
- $I_{C_{max}}$—iteration in which the quasi-optimal solution was found,
- $\overline{C_{max}}$—average of C_{max} in 12 algorithm runs,
- $\delta^{TS} = (C^{TS^*}_{max} - C^{TS}_{max})/C^{TS}_{max} * 100\,\%$—percentage relative value of the improvement (values $C^{TS^*}_{max}$ and C^{TS}_{max} are average from 12 algorithms runs),

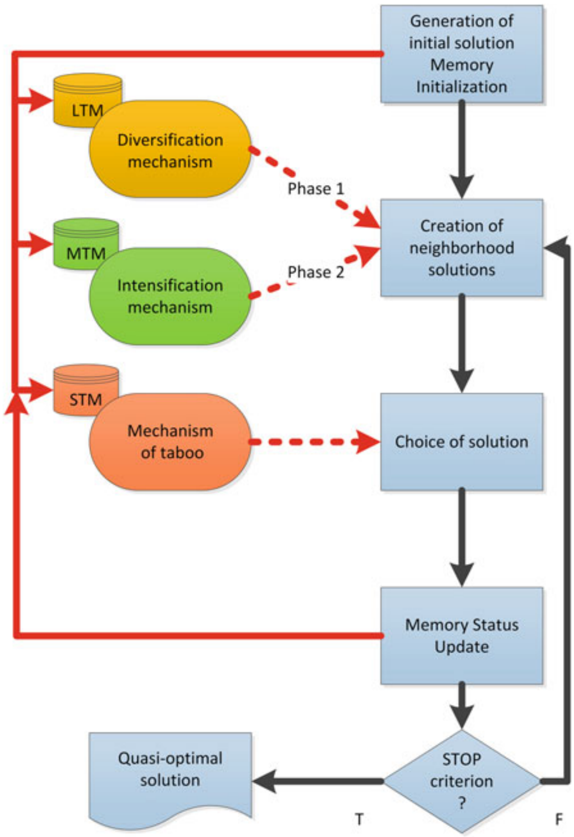


Fig. 6 Scheme of *TS* algorithm

- $\delta^{UB} = (C^{TS*}_{max_{best}} - C^{UB}_{max})/C^{UB}_{max} * 100\,\%$—percentage relative value of the improvements. The value $C^{TS*}_{max_{best}}$ is the best found objective value in 12 algorithms runs and C^{UB}_{max} is the best known value or upper bound for instance [16],
- *UB*—upper bound.

The negative values of $\overline{\delta^{TS}}$ in Table 1 mean that implementation of the diversification mechanism in algorithm *TS** leads to improving the value of the criteria function found by *TS* algorithm. In the cases of `ta021` and `ta069` the instances' algorithm enables us to obtain better solutions than the reference solutions found in the literature. In other cases *TS** finds worse solutions than the best known ones on about 1 %. The intensification mechanism improves about 3–21 times the quasi-optimal value found in the first phase (more often for the bigger test instances). However, using the medium-term memory mechanism leads to longer algorithm execution—about 1000–5000 iteration in contrast to 600 in *TS* algorithm (not using the intensification mechanism).

Table 1 Experiments' results

Instance	TS			TS*					
	C_{max}	$I_{C_{max}}$	$\overline{C_{max}}$	C_{max}	$I_{C_{max}}$	$\overline{C_{max}}$	$\overline{\delta^{TS}}$ (%)	UB	δ^{UB} (%)
ta002 − 20×5	1359	31	1365	1359	77	1361	−0.32	1359	0.00
ta006 − 20×5	1195	10	1216	1195	57	1202	−1.20	1195	0.00
ta009 − 20×5	1230	14	1251	1230	34	1238	−1.06	1230	0.00
ta012 − 20×10	1678	14	1698	1660	581	1676	−1.30	1659	0.06
ta016 − 20×10	1408	120	1420	1397	415	1405	−1.05	1397	0.00
ta017 − 20×10	1492	24	1507	1486	414	1492	−0.95	1484	0.13
ta021 − 20×20	2303	25	2323	2296	421	2311	−0.50	2297	−0.04
ta025 − 20×20	2311	9	2332	2291	208	2315	−0.74	2291	0.00
ta030 − 20×20	2186	18	2218	2183	143	2196	−0.96	2178	0.23
ta033 − 50×5	2623	11	2643	2621	11	2633	−0.37	2621	0.00
ta036 − 50×5	2829	15	2844	2829	15	2836	−0.28	2829	0.00
ta038 − 50×5	2694	18	2706	2683	79	2695	−0.40	2683	0.00
ta041 − 50×10	3046	32	3115	3046	510	3071	−1.41	3025	0.69
ta044 − 50×10	3079	11	3144	3064	595	3096	−1.50	3064	0.00
ta046 − 50×10	3054	21	3106	3021	850	3063	−1.39	3006	0.50
ta051 − 50×20	3944	26	3967	3909	2259	3925	−1.06	3886	0.59
ta055 − 50×20	3708	35	3766	3663	1290	3720	−1.20	3635	0.76
ta057 − 50×20	3797	28	3845	3722	759	3795	−1.29	3716	0.16
ta065 − 100×5	5255	21	5263	5250	34	5255	−0.15	5250	0.00
ta067 − 100×5	5246	10	5275	5246	10	5270	−0.10	5246	0.00
ta069 − 100×5	5448	37	5504	5448	37	5458	−0.18	5454	−0.11
ta071 − 100×10	5800	21	5842	5784	401	5806	−0.61	5770	0.24

(continued)

Table 1 (continued)

Instance	*TS*			*TS**				UB	δ^{UB} (%)
	C_{max}	$I_{C_{max}}$	$\overline{C_{max}}$	C_{max}	$I_{C_{max}}$	$\overline{C_{max}}$	$\overline{\delta^{TS}}$ (%)		
ta074 − 100×10	5838	39	5894	5826	1946	5852	−0.71	5791	0.60
ta078 − 100×10	5685	62	5686	5650	62	5663	−0.41	5623	0.71
ta081 − 100×20	6362	60	6427	6313	60	6377	−0.77	6286	0.43
ta085 − 100×20	6454	60	6552	6443	1541	6469	−1.28	6377	1.02
ta089 − 100×20	6396	40	6495	6374	146	6445	−0.77	6358	1.34

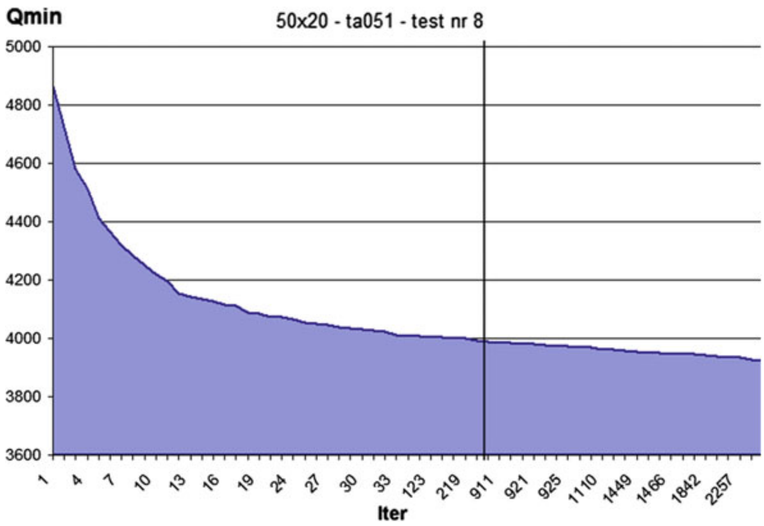


Fig. 7 Execution of the *TS** algorithm for the task `ta051`—dependence of the objective function of quasi-optimal solutions on the number of iterations. The *vertical line* marks the end of Phase I

Table 2 The average results for the second phase of *TS** algorithm

Problem size	Number of improvements	Number of returns	Return count (best C_{max})
20×5	3	8	3
20×10	5	9	5
20×20	3	8	4
50×5	3	7	2
50×10	12	12	7
50×20	13	13	8
100×5	2	7	2
100×10	9	11	5
100×20	21	13	8

A typical run of the *TS** algorithm is shown in Fig. 5. The vertical arrows show the iterations, in which return to the previous (stored in the memory MTM) solutions. Return was executed 5 times. The best solution was obtained after the first time. The graph shows that the return directs the search process in a different part of the solution space. Figure 7 presents the dependence of the objective function of the current quasi-optimal solution from the number of iterations. In Table 2, statistical data about the number of improved solutions in the second phase *TS** algorithm was collected (the average of the selected problems of the same size). Here, we also present the number of returns in the second phase and the number of returns for which a quasi-optimal solution is found. These results indicate that the effectiveness of the proposed mechanism increases with the size of the problem instance.

5 Conclusions

The results presented in this paper confirm the high efficiency of the proposed implementation of *TS* algorithm in job scheduling and particularly in strategy planning. The implemented intensification mechanism enables us to improve *TS* algorithm results above 1 %, which is comparable with the distance to the reference solutions. However, this improvement was obtained through an increase in computational effort.

The approach presented in this paper can be further extended to include tree-like scheduling variants using anticipatory networks [9, 10] which may allow the planning team to consider different future scenarios for product development. This proves particularly useful when applying ICT (Information and Computer Technology) foresight results to planning the technological development of industrial computer vision systems, (cf. Fig. 1) [15]. Foresight scenarios provide clues as regards alternative market circumstances and semiconductor technologies available and emerging technologies of the pattern recognition based on the image understanding [13, 14]. Development of the future computer vision systems requires new methods of information management. For example, these systems will use ontology for description of the knowledge collected or obtained from the external knowledge repositories. They are described as branches of an anticipatory tree [10]. Consequently, problems of type (1) are to be solved along each causal path in an anticipatory network [7, 8, 11]. The general creativity approaches in MCDM (Multicriteria Decision Planning) [8, 11] may be applied to select a compromise product development plan applying the roadmapping methodology presented in Sect. 2. During the works on the strategic planning optimization for the NPD problem a several optimization algorithm was considered. For example, among many approximation algorithms inspired by nature, the algorithm called the firefly (FA) was tested. The FA algorithm is a novel metaheuristic, which is inspired by the behaviour of fireflies [4]. However, the carried out tests have shown, in this particular problem, the advantage of the *TS* algorithm.

The next step will be the analysis of business data from company workflows to obtain test instances which describe real cases.

References

1. Chmiel, W.: Conditional expected value of objective function in approximation algorithms for pemutational problems. Automatyka **1–2**(136), 31–42 (2002)
2. Chmiel, W., Kadłuczka, P.: Hybrid algorithm using long-term memory. Automatyka **3**(1), 59–70 (1999)
3. Chmiel, W., Kadłuczka, P.: Conditional expected value of objective function in approximation algorithms for pemutational problems. Automatyka **1–2**(9), 47–56 (2005)
4. Filipowicz, B., Chmiel, W., Kadłuczka, P.: Guided search of the solution space in swarm algorithm. Automatyka **13**(2), 247–255 (2009)

5. Glover, F.: Tabu search part i. ORSA J. Comput. **1**(3), 190–206 (1989). http://pubsonline.informs.org/doi/abs/10.1287/ijoc.1.3.190
6. Skulimowski, A.M.J. (ed.): Looking into the future of creativity and decision support systems. In: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems, Kraków, Poland, Nov 7–9 2013, Advances in Decision Sciences and Future Studies, vol. 2, pp. 57–62. Progress Business & Publishers, Kraków (2013)
7. Skulimowski, A.M.J.: Methods of multicriteria decision support based on reference sets. In: Caballero, R., Ruiz F., Steuer, R.E. (eds.) Lecture Notes in Economics and Mathematical Systems, vol. 455, pp. 282–290. Springer, Berlin (1997)
8. Skulimowski, Andrzej M.J.: Freedom of choice and creativity in multicriteria decision making. In: Theeramunkong, T., Kunifuji, S., Sornlertlamvanich, V., Nattee, C. (eds.) Knowledge, Information, and Creativity Support Systems. Lecture Notes in Computer Science, vol. 6746, pp. 190–203. Springer, Heidelberg (2011)
9. Skulimowski, Andrzej M.J.: Hybrid anticipatory networks. In: Artificial Intelligence and Soft Computing. Lecture Notes in Computer Science, vol. 7268, pp. 706–715. Springer, Heidelberg (2012)
10. Skulimowski, A.M.J.: Anticipatory network models of multicriteria decision-making processes. Int. J. Syst. Sci. **45**(1), 39–59 (2014). http://www.tandfonline.com/doi/abs/10.1080/00207721.2012.670308
11. Skulimowski, A.M.J., Pukocz, P.: Enhancing creativity of strategic decision processes by technological roadmapping and foresight. In: KICSS 2012: 7th International Conference on Knowledge, Information and Creativity Support Systems, IEEE Computer Society. CPS Conference Publishing Services (2012)
12. Szymlak, B.: Tabu search algorithm for job scheduling. Master of Science thesis, supervisor: Wojciech Chmiel Ph.D., AGH Kraków (2012)
13. Tadeusiewicz, R., Szczepaniak, P.: Basic concepts of knowledge-based image understanding. In: Lecture Notes in Computer Science, vol. 4953, pp. 42–52. Springer, Heidelberg (2008)
14. Tadeusiewicz, Ryszard: Earliest computer vision systems in poland. In: Choraś, Ryszard S. (ed.) Image Processing and Communications Challenges 2. Advances in Intelligent Systems and Computing, vol. 84, pp. 3–13. Springer, Heidelberg (2010)
15. Tadeusiewicz, R.: Place and role of intelligent systems in computer science. Comput. Methods Mater. Sci. **10**(4), 193–206 (2010)
16. Taillard, E.: Benchmarks for basic scheduling problems. Eur. J. Oper. Res. **64**(2), 278–285 (1993)

Need for Collective Decision When Divergent Thinking Arises in Collaborative Tasks of a Community of Practice

D. Assimakopoulos, M. Tzagarakis and J. Garofalakis

Abstract Based on previous work (Assimakopoulos et al. Proceedings of KICSS'2013 [5]) where we introduced HelpMe tool, a tool that automatically selects a group of people according to rules and metrics, we now introduce inconsistency between divergent ranking concerning answers during collaborative tasks among experts. In HelpMe tool, users collaboratively create a knowledge base about a subject and evaluate user opinions in order to achieve quality of knowledge. The basic assumption of the tool is that "Knowledge comes from experts." This is achieved by collective evaluation, through voting and discussion in every stage (Task) of the discussion (Activity). Inconsistency appears when a set of sentences cannot be true at the same time (Adrian et al. Proceedings of KICSS'2013 [3]). During collaborative tasks in Communities of Practice, inconsistency may inspire new associations and lead to more interesting solutions. However, there are cases such as medical, legal, etc. issues in which contradicting views are not helpful since a final decision has to be made. This paper is focused on such cases and examines the possible options and the methods that have to be implemented in order for a final decision to be made. When difference in the evaluation range is observed (divergent voting), the community should be informed in order to evaluate the existing answer with an "up vote" or a "down vote." No other option is available. In case of inconsistency, we introduce a loop procedure that informs experts in order

D. Assimakopoulos (✉) · J. Garofalakis
Computer Engineering and Informatics Department, University of Patras,
N. Kazantzaki, 26500 Rion, Greece
e-mail: asimakop@ceid.upatras.gr

J. Garofalakis
e-mail: garofala@cti.gr

D. Assimakopoulos · M. Tzagarakis · J. Garofalakis
CTI, Computer Technology Institute and Press "Diophantus", University of Patras,
N. Kazantzaki, 26500 Rion, Greece
e-mail: tzagara@upatras.gr

M. Tzagarakis
Department of Economics Research Academic Computer Technology Institute,
University of Patras, N. Kazantzaki, 26500 Rion, Greece

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_25

to evaluate the task being observed as inconsistent. If inconsistency remains, or the number of evaluators is small, the process keeps going until it meets the criteria introduced in this paper. On the other hand, the system should be able to find, in its knowledge database, related conversations and examines the decisions made on related or same subject.

Keywords Collaboration · Web 2.0 · Communities of Practice (CoPs) · Argumentative collaboration · Social media · Recommender systems · Expertise finding · Inconsistency

1 Introduction

1.1 Communities of Practice and Argumentation

Communities of practice is a gathering place of people with common interests, is created. CoPs have become important places for people that seek and share experience and interests. CoPs are contribution places in which knowledge is offered to its members [19]. Basic characteristics of such a community are: (a) Interest field, (b) The community itself, and (c) Practice as mentioned in [5].

Argumentation is the study of how conclusions can be reached through logical reasoning. Argumentation focuses on a particular kind of semantic structure for organizing elements. Of central interest, therefore, is the Web 2.0 emphasis away from predefined information organizing schemes, toward self-organized, community indexing ("tagging") of elements, resulting in so-called "folksonomies" that can be rendered as tag clouds and other visualizations. Persuading "normal people" (in contrast to skilled information scientists or ontology engineers) to create structured, sometimes high-quality, metadata was previously thought impossible, and the success and limits of this approach are now the subject of a new research field that studies collaborative tagging patterns, e.g., [10].

1.2 Existing Problems in Communities of Practice

Basic problems in CoPs as described in [5] are: (a) Information overload and (b) data interconnection. All these above-mentioned problems are major to CoPs because intense effort is required in order to search and sort information and to find a user's opinion about a subject through time. In argumentative tools, some of the members evaluate questions and answers using Web 2.0 characteristics while others do not. For example, reddit's "down vote" is not the same as a "Dislike." Down vote shows that a specific post does not deserve to be in the front page. Tools with

Web 2.0 characteristics cannot get away from stack representation like forums we all do know. New ways of representation should be explored which will make it is easy for someone to: (a) visualize a conversation with the form of graph nodes or clouds or something else, (b) watch the conversation as it evolves or get involved in any stage, (c) be able to find appropriate users that would like to get involved in a specific conversation. A conversation at HelpMe tool is a collection of messages concerning the same tags. HelpMe tool chooses users who would like to participate in the conversation introducing two new metrics ULQI (user label query importance) and ULCI (user label communication importance) [5].

1.3 The Role of Tools in CoPs

Over the years, a variety of tools supporting argumentative collaboration have appeared; they usually facilitate argumentative discussions among members of a group and range from simple ones such as email, chat, and web-based forums to dialogue mapping and argumentative collaboration tools, reaching even into the realm of sophisticated conferencing and formal argumentation systems [16]. Tools in CoPs are very popular with everyone. These tools help people find users to help them with their problem. This is what is called expertise finders. Expertise finders, or expertise location engines, are systems that help find others with the appropriate expertise to answer a question. These systems have been explored in a series of studies, including Streeter and Lochbaum [15], Krulwich and Burkey [13], McDonald and Ackerman [1] as well as the studies in Ackerman et al. [2]. Newer systems, which use a social network to help find people, have also been explored, most notably in Yenta [9], ReferralWeb [12], and most recently commercial systems from Tacit and Microsoft. These systems attempt to leverage the social network within an organization or community to help find the appropriate others. In reality, relatively few people will claim themselves as an expert, but many people agree that they have some measure of expertise in some area. These systems allow everyone to contribute as they can. A person's expertise is usually described as a term vector and is used later for matching expertise queries using standard IR techniques. The result usually is a list of related people with no intrinsic ranking order or ranks derived from term frequencies. It may reflect whether a person knows about a topic, but it is difficult to distinguish that person's relative expertise levels [2].

Due to argumentative collaboration, why do we keep using tools? The answer is simple: because users' needs cannot be covered by the use of only one tool. For example, a search engine does not take into consideration the experience of the user who submits the question and as a result, the received answers do not match the user's profile. In conclusion, we can say that communities of practice use these tools in order to solve the problem collectively.

1.4 Existing Tools and Approaches in Communities of Practice

Based on previous work [5], five tools in the area of CoPs were examined: (a) CoPe_it!: an incremental formalization approach that facilitates the emergence of individual and loosely coupled resources into coherent knowledge structures and finally decisions. (b) Parmenides: is designed to support web-based policy consultation with the public and incorporates a formal model of argumentation [6]. (c) Cohere: a web tool for social bookmarking, idea linking, and argument visualization. It incorporates the Web 2.0 principles to create an environment called Cohere which aims to be semantically and technically open, provide an engaging user experience and social network, but provide enough structure to support argument analysis and visualization. Here, connections are created through ideas [8]. (d) Stackoverflow: as an alternative view of a forum. Its basic characteristic is questions and answers given under a wide variety of subjects to the community of software developers. Users can ask, answer, vote, and edit questions and answers like wiki or dig (social news web site) style. Stackoverflow users can earn reputation points and badges, depending on how other users judge them by the answers they have given. [17]. It is very well known that Stackoverflow is not just simply a revamp of a Q&A. It automatically suggests similar discussions and also, the community responses can be judged so that if a similar question was asked, the system can quickly get to the correct answer. (e) Reddit: is a social news and entertainment web site where registered users submit content in the form of either a link or a text ("self") post. Other users then vote the submission "up" or "down," which is used to rank the post and determine its position on the site's pages and front page. Content entries are organized by areas of interest called "subreddits." [18].

In [5] was introduced the HelpMe tool which is a Web 2.0 tool that supports: (a) Users who want to find answers to their problem by finding other users of the community. (b) Labels (or tags) to characterize a conversation: the Labels become the key for the HelpMe tool in order to automatically suggest similar discussions. By clicking a tag, all similar conversations are displayed. (c) Visualization through node graph, tag clouds, and statistic web pages. Node graphs can display the content tree of a conversation, user grades at any task, etc. (d) Recording of conversations for future retrieval. All conversations are stored in a SQL server database. (e) Evaluation system through metrics.

The above-mentioned tools were examined in the following Web 2.0 characteristics: (1) Rating issues: is the assessment of issues as listed by users of a community. (2) Rating of user answers: is the evaluation of the responses as rated by the users. (3) Rating of user evaluation: the answers provided by users are evaluated by others. So, as time goes by, a user rating index is created according to the issues which it tackles. (4) Karma or user reputation: some tools like Reddit encourage posts so that they do not show outside the community (i.e., self posts). Users are rated according to the self posts they make. Therefore, a karma rating is created for the Links and another karma for user responses. Similarly, users of

Table 1 Comparison between CoPs tools

	Rating issues	Rating of user answers	Rating of user evaluation	Karma or user reputation	Comments
CopeIt!	Yes	Yes	No	No	Yes
Parmenides	No	No	No	No	Yes
Cohere	No	No	No	No	Yes
Stackoverflow	Yes	Yes	Yes	Yes	Yes
Reddit	Yes	Yes	Yes	Yes	Yes
HelpMe	Yes	Yes	Yes	Yes	Yes

stackOverflow make reputation, when doing specific actions in the community. Such actions are considered the retagging of queries or close a query. (5) Comments: is the evaluation of the text answer given by users on a specific subject (Table 1).

2 HelpMe Community Tool

2.1 A Scenario of HelpMe Tool

A user seeks for a solution to his problem. He sends an email at helpme@westgate.gr and receives a link that leads him to a confirmation page. In this page he connects the initial question to an existing tag or set of tags, or creates a new tag (if no tag exists) to define the initial question. In this page, he also specifies the criteria or metrics that will be used by the system in order to find, in an automatic way, users capable of participating in a conversation whose goal is to find a solution to the initial question. These users will create an autonomous group in the community and may participate in as many conversations as the system decides. The conversation is represented as nodes to denote users and edges to denote an answer or like, dislike, best answer to an answer [5]. It should be mentioned that HelpMe tool is constantly developing and therefore it is expectable that sometimes it is not functional due to upgrading procedures. It is obvious it is used for research purposes and it is currently at a preliminary development stage. It is not a commercial product ready to use and it is subject to further research and development.

2.2 Start a New Conversation

The HelpMe tool receives a new query by the form of an email. Then, after user registration (if not registered), init user receives a link in his mailbox that sends him to the HelpMe confirmation page. In there, start user characterizes his question using existing or new tags and specifies the way he wants HelpMe tool to find users, in order to answer his init question, so that a new conversation will start. At this stage, three main issues are considered:

- User initialization: the HelpMe tool is informed that a new user exists, so he/she may use HelpMe existing tags.
- Tag initialization: the HelpMe tool is informed that a new label exists, so it may be used by the existing users of the tool. In other words, all members of the community are initialized toward the new label.

User defines metrics association: Init user creates a relationship between ULQI and ULCI, in order for HelpMe tool to find users to start a conversation with him. ULQI is the average of all queries (questions) evaluated, concerning a user label. *Query evaluation concerns how community users rank the initial question.* ULCI is the average of all answers evaluated, concerning a user label. *Answer evaluation concerns how community users rank the user label after the first stage of initial question,* i.e., the conversation that takes place between community users. Finally, init user may create his own model of choosing users to participate in his initial question, using the score formula: ULQI * perc1 + ULCI * perc2. Score formula is the resulting label score from the previous formula. It should be mentioned that, perc1 and perc2 are defined by init user and the sum should be 1 (perc1 + perc2 = 1). For example, init user may give the formula 0.7 * ULQI + 0.3ULCI, which means that he wants the system to find users whose average of questions evaluated, concerning the label stated by the initial user, is 70 % more important in comparison with the average of all answers evaluated, concerning the same label, which is 30 % less important to the init user.

Figure on the right displays how the initial question is transformed into a conversation place (Fig. 1).

2.3 Messages Received

When init user chooses the criteria in the confirmation page, a HelpMe tool function finds users who would like to participate and give answers by doing the following:

- Creating a dynamic query in such a way that users fulfill the criteria of the score formula.

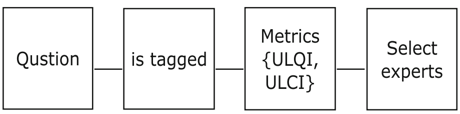


Fig. 1 HelpMe expertise finder model

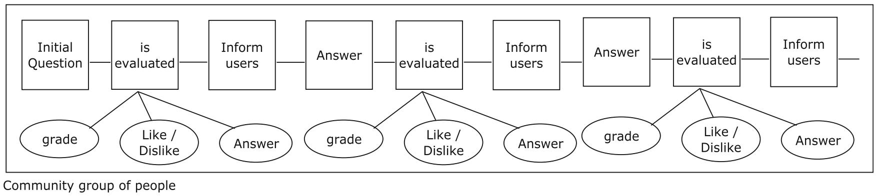


Fig. 2 Conversation model

- All selected users are recorded in database table. Every insert in that table fires a post insert trigger that sends a mail to the corresponding user.
- Then, the user node is created so that user may give an answer to a specific task of the conversation. This user node is stored in a database recursive structure that is explained in TheJit visualization section below.

Users chosen by HelpMe tool receive an email that asks for their participation in order to solve the problem (i.e., label defined by init user at the confirmation page). HelpMe tool records the users who receive the email and the corresponding link. The link creation is a very important procedure for HelpMe and gives the opportunity to any user who has received it, to participate at any time of the conversation. Important information that is carried through the link is the serial number of the conversation (i.e., Activity ID). Activity ID is created by an identity field in a database table, after HelpMe has received an email from init user. By knowing the Activity ID, HelpMe can display graphically all phases of the conversation (i.e., every phase is a TaskID) (Fig. 2).

A question is evaluated by the community users, see conversation model above, giving a grade or like/dislike or text answer, which are related to the question through the tag that characterizes the question. The first user of the community who responses to a question (box Answer of the above figure) leads the discussion to the next level (TaskID ← TaskID + 1). Any user is free to evaluate any answer at any level of the conversation. The conversation never ends, just loses interest at some time. Future work is to discover and exploit provenance of collective decisions of a conversation which took place at past time.

2.4 Visualization

In [5] HelpMe tool represents users as nodes and answers as edges. This information is stored in a database table where a recursive structure is created between father node (field name: id) and children node (filed name: children). In any step of the conversation (taskID), the text answered, the grade given, like/dislike/best answer, the average of the task, the average of the activity, and the number of conversation views are saved in an appropriate database structure. HelpMe tool uses

Fig. 3 TheJit visualization of user nodes: a user is displayed as a node with his id and email

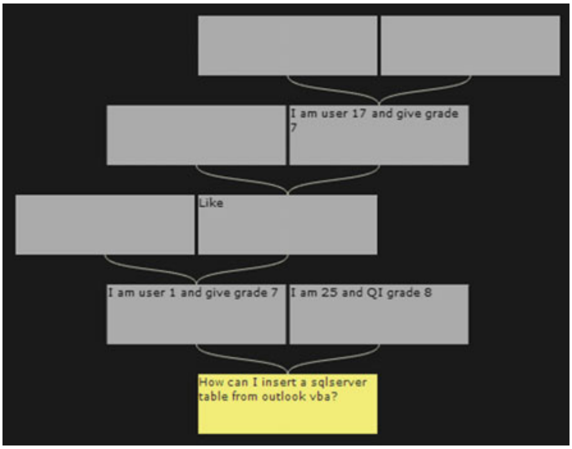


Fig. 4 TheJit Content visualization: displays the answers given at all tasks. Blank means no answer

Javascript Infovis Toolkit (TheJit) to represent users (nodes) and answers (edges). Graph visualization is more flexible because it provides many different views to the community users such as content, user info and statistics like traffic at a node, label ranking, user ranking, etc. (Figs. 3, 4 and 5).

Another type of visualization implemented in [5] was tag clouds [4, 7]. There are three types of clouds in the area of social software: (a) Tag clouds: are usually represented by html elements and may appear in alphabetical order, randomly, or according to weights. (b) Global tag clouds: where info frequencies appear with accumulative manner against elements and users (c) Sized categories: suggest the number of sub categories. HelpMe tool uses text clouds in order to represent the

Fig. 5 HelpMe tool uses text clouds in order to represent the word frequency as a weighted list

Fig. 6 Statistics can display questions by label. By clicking each button, a specific page opens (i.e., answers button, leads to a page that displays all conversation answers, label button leads to relative conversations, user button leads to community user involvement, and so on.) By clicking ActvID, for example, 290, the node graph opens. By clicking cloud, the tag cloud of the conversation opens

word frequency as a weighted list. In order to visualize the tags, appropriate structures exist. Every tag is saved separately and ranked.

The third type of visualization implemented in [5] was statistics represented in web pages (Fig. 6).

2.5 Ranking Method

A user in HelpMe tool can give a grade from 1 to 10 to a previous answer and write some text explaining the reasons of his grade as well as to justify his own opinion about the subject. Alternatively, he may like, dislike the previous answer. A like corresponds to grade 8 while a dislike corresponds to -1. Alternatively, if and only if he is the start user, he may give a best answer which counts 10 points in HelpMe statistics. HelpMe tool gives users the opportunity to display community queries: (a) In a descending/ascending order according to date and time (b) According to: views, likes, dislikes, best answer, ULQI, ULCI, label ranking, user ranking, traffic information at any node (see figure: degrees of nodes). Each question takes the place of a row where is being displayed the ActvID, votes, answers, number of views, likes, dislikes, best answers, subject, label, and init user. Statistics can be expanded on a specific user to see user questions, answers, labels used, etc., or can be expanded on a specific label to see label ranking, etc.

2.6 HelpMe Tool Workflow Algorithm

When a user in the community receives a link, by clicking it HelpMe tool makes a series of checks:

1. It checks if he is the first who gives an answer. It is easy to find if the is a TaskID greater than the current one for this conversation. The conversation, as mentioned previously, is represented by the Activity ID.
2. If there no greater TaskID for that ActvID, then he is the first who answers the question. In that case, TaskID $\longleftarrow$ TaskID + 1. The first user answers a previous question, assigns a new TaskID. On the same time he gives an answer or chooses between like, dislike, or best answer.
3. If next TaskID exists, then someone else answered first. In that case, the TaskID does not change. Then he also gives an answer or chooses between like, dislike, or best answer.

2.7 HelpMe Tool Workflow Procedure

A workflow procedure of a query stated by an initial user is presented in the figure at the bottom. The conversation with Activity ID 223 is first initiated by the start user (i.e., user1). He creates the first step of the conversation (TaskID = 001) on 07/07/2011 at 18:30. The first user who responds to that question is user3. User3 gives his own answer to the user1' label question, perhaps gives a good vote or likes the init question and then he takes the conversation to the next level

Step	ActvID	TaskID	Mail From - (out)	Mail To - (in)	Answered to	Date
1	223	001	User1	User3		07/07/2011 18:30
	223	001		User6		07/07/2011 18:30
	223	001		User7		07/07/2011 18:30
2	223	002	User3	User1	User1 (223-001-1)	07/07/2011 19:90
	223	002		User6		07/07/2011 19:90
	223	002		User7		07/07/2011 19:90
3	223	003	User7	User1	User1 (223-001-1)	08/07/2011 10:15
	223	003		User3		08/07/2011 10:15
	223	003		User6		08/07/2011 10:15
4	223	004	User6	User1	User7 (223-003-1)	08/07/2011 13:31
	223	004		User3		08/07/2011 13:31
	223	004		User7		08/07/2011 13:31
5	223	005	User3	User1	User6 (223-004-2)	08/07/2011 14:40
	223	005		User6		08/07/2011 14:40
	223	005		User7		08/07/2011 14:40

Fig. 7 Workflow tasks of a conversation

(TaskID = 002). The first user who answers at TaskID 002 is user7. User 7 has two options: to answer to the initial question of user1 or/and to answer to user2. By doing the first, the taskID will not change, while by doing the second TaskID will become 003. It should be mentioned that he is able to do both options. Every user is able to evaluate an answer at any level (taskID). A conversation never ends, it only loses its interest in time (Fig. 7).

In order for a conversation to get to the next level, someone needs to answer first to the question stated by the previous user. At any time, any other user may give his own answer but the taskID will not change. TaskID increases by one, only by the user who goes the conversation one step further (Figs. 8 and 9).

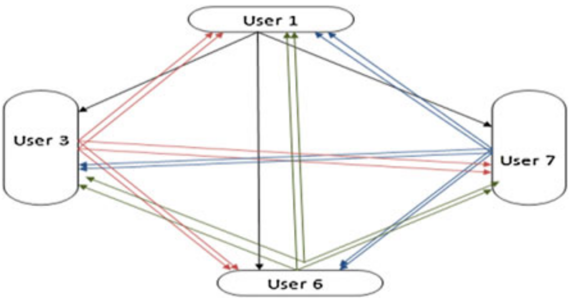


Fig. 8 Displays the traffic (answers) at any node. A node represents a user and directed edges represent traffic between users. So, at a glance, one may discover the active pairs of users

User	graph node degree in (mail TO)	graph node degree out (mail FROM)
User1	6	3
User3	5	6
User6	5	6
User7	5	6

Fig. 9 "Node in" is the number of times users replied to a user. For example, six users replied to user1. "Node out" is the number of times a user replied to a question. For example, user1 replied three times (to users: user3, user7, user6). These facts may help to build the profile of each user and are displayed in the statistics section

3 Recommender Systems

Recommender systems or recommendation systems are a subclass of information filtering system that seek to predict the "rating" or "preference" that user would give to an item or social element (e.g., people) they had not yet considered, using a model built from the characteristics of an item (content-based approaches) or the user's social environment (collaborative filtering approaches) [14]. According to Groza et al. [11], requirements of discovering and exploiting provenance are to provide a framework that consists of a model based on modularization, provenance information, identification and revision, support for domain knowledge, support for linguistic features, and complementing argumentation with orthogonal models.

Collaborative filtering methods are based on collecting and analyzing a large amount of information on users' behaviors, activities, or preferences and predicting what users will like based on their similarity to other users. Content-based filtering methods are based on information about and characteristics of the items that are going to be recommended. In other words, these algorithms try to recommend items that are similar to those that a user liked in the past. Approaches like collaborative filtering (e.g., user rating, rank labels, user opinions, keep record of views of a conversation, etc.) or content-based filtering (like, dislike buttons) have been partially implemented at HelpMe tool. HelpMe tool can recommend conversations through tags so far.

Currently, in HelpMe tool, we introduced the inconsistency that arises when difference in the evaluation range is observed (divergent voting) and propose a solution by sending invitations to experts until the inconsistency is eliminated, through score formula.

4 Methodology

HelpMe tool is an expertise finder that finds community users according to ULQI and ULCI metrics introduced here for the first time. The methodology behind the tool is expanding in five stages [5]: (a) study of forum visualization so far (b) user needs about the content of a conversation in a forum (c) design of HelpMe tool (d) implementation of the tool (e) testing and evaluation. Unfortunately, due to the page limitation it is difficult to analyze the methodology in a separate chapter. It is a simple reputation system that computes and publishes reputation scores for a set of labels and users. The opinions are typically passed as ratings to the database and the tool uses a specific algorithm, using ULQI and ULCI to dynamically compute the reputation scores based on the received ratings. Two major things that make HelpMe tool to go beyond comparing to other tools is that (a) it uses content visualization of a conversation, rather than using stack representation like forums do and (b) uses two new metrics for expertise finding and computing the reputation scores based on the received ratings.

One can find several interpretations of inconsistency that are reflected in various definitions. Intuitively, inconsistency appears when a set of sentences (formulas, theorems, beliefs) cannot be true at the same time. When inconsistency arises, the system should automatically and appropriately react [3].

5 Handling Inconsistency

What goes beyond the previous work [5] is to introduce the inconsistency that arises when difference in the evaluation range is observed (divergent voting). In HelpMe tool, an inconsistency range may be defined by the community. For example, the community dealing with a subject can define the *inconsistency range* to vary between x grades from min to max. If inconsistency takes place at any task of the activity, the community should be informed in order to evaluate the existing answer with an "up vote" or "down vote". No other option is available. This will make the user to decide if he agrees with the specific opinion or not. In HelpMe tool we do not want to suppress inconsistency. We want each task of the activity to have the approval of the community. Since each task is approved, in every stage of the activity, the result will be an approved activity (conversation). The invitation must be given to as many experts to the subject as possible. In this way incorporating inconsistency into reasoning is achieved. In HelpMe tool this process is implemented by selecting distinct users of all related conversations that exist in the tool's database. In case of equality or small number of votes, the system should seek for more experts in the community by changing the score formula. *This process can be a loop that keeps inviting experts according to the criteria specified by the community.* Such criteria can be the number of evaluators and the inconsistency range. If the number of evaluators in a specific task is very small (parameter that can be defined by the community), then the criteria will not be met. If the inconsistency range will not drop to acceptable limits (as defined by the community) the criteria will also not be met.

In other words, if the criteria do not meet, the loop will go on until the score formula reaches the lowest point and the tool will invite all members of the community. A lowest point of the score formula may be: $0.5 * \text{ULQI} + 0.5 * \text{ULCI} > 0$.

Another approach to handle inconsistency is that the system should be able to find in its knowledge database related conversations, and propose the decisions made sometime in the past. In that case resources may at a later point be obsolete or characterized as unimportant by the group. The sheer diversity of the resource types requires from individual members of the group to engage into the process of information triage, i.e., sorting the available material, interact with the resources on the space in an attempt to interpret, and recast them as well as organize them into larger structures. Some resources may even have to be filtered out or marked as unimportant [16].

References

1. Ackerman, M.S., McDonald, D.W.: Answer Garden 2: merging organizational memory with collaborative help. In: Ackerman, M.S. (ed.) Proceedings of the ACM Conference on Computer Supported Cooperative Work, Computer Supported Cooperative Work, pp. 97–105. Boston, ACM Press, New York (1996)
2. Ackerman, M.S., Wulf, V., Pipek, V.: Sharing Expertise: Beyond Knowledge Management. MIT Press, Cambridge, MA (2002)
3. Adrian, W.T., Nalepa, J.G., Ligeza, A.: On potential usefulness of inconsistency in collaborative knowledge engineering. In: Skulimowski, A.M.J. (ed.) Proceedings of KICSS'2013. Progress & Business Publishers, Kraków, pp. 14–25 (2013)
4. Aouiche, K., Lemire, D., Godin, R.: Collaborative OLAP with Tag Clouds: Web 2.0 OLAP Formalism and Experimental Evaluation, WEBIST, Madeira, Portugal (2008)
5. Assimakopoulos, D., Tzagarakis, M., Garofalakis, J.: Communities of practice for developers: HelpMe tool. In: Andrzej M.J. Skulimowski (ed.), Proceedings of KICSS'2013. Progress & Business Publishers, Kraków, pp. 26–37 (2013)
6. Atkinson, K., Bench-Capon, T., McBurney, P.: PARMENIDES: facilitating deliberation in democracies. Artif. Intell. Law **14**(4), 261–275 (2006)
7. Bielenberg, K., Zacher, M.: Groups in social software: utilizing tagging to integrate individual contexts for social navigation. Masters Thesis submitted to the Program of Digital Media, Universität Bremen (2006)
8. Buckingham Shum, S.: Cohere: Towards Web 2.0 Argumentation. Knowledge Media Institute, The Open University, UK (2008)
9. Foner, L.N.: Yenta: a multi-agent, referral based matchmaking system. In: Johnson, W.L., Hayes-Roth, B. (eds.) Proceedings of the First International Conference on Autonomous Agents, pp. 301–307. ACM Press, New York (1997)
10. Golder, S., Huberman, B.A.: Usage patterns of collaborative tagging systems. J. Inf. Sci. **32**(2), 198–208 (2006)
11. Groza, T., Handschuh, S., Breslin, J.G.: Adding provenance and evolution information to modularized argumentation models. In: WI-IAT'08. IEEE/WIC/ACM International Conference, vol. 1 (2008)
12. Kautz, H., Selman, B., Shah, M.: Referral Web: combining social networks and collaborative filtering. Commun. ACM **40**(3), 63–65 (1997)
13. Krulwich, B., Burkey, C.: The ContactFinder agent: answering bulletin board questions with referrals. In: Proceedings of the 13th National Conference on Artificial Intelligence and Eighth Innovative Applications of Artificial Intelligence Conference, AAAI 96, Portland, Oregon, vol. 1. AAAI Press, Menlo Park, CA, pp. 10–15 (1996)

14. Ricci, F., Rokach, L., Shapira, B.: Introduction to recommender systems handbook. In: Ricci, F., Rokach, F., Shapira, L., Kantor, B. (eds.) Recommender Systems Handbook, pp. 1–35. Springer, Heidelberg (2011)
15. Streeter, L., Lochbaum, K.: Who knows: a system based on automatic representation of semantic structure. In: Proceedings of the Conference on Computer-Assisted Information Retrieval, RIAO'88 Program Conference, Cambridge, MA, 21–24 March, pp. 380–388. CID, Paris (1998)
16. Tzagarakis, M., Karousos, N., Gkotsis, G., Kallistros, V., Christodoulou, S., Mettouris, C., Kyriakou, P., Nousia, D.: From 'Collecting' to 'Deciding': facilitating the emergence of decisions in argumentative collaboration. Research Academic Computer Technology Institute, N. Kazantzaki, University of Patras, 26500 Rion, Greece (2010)
17. Wikipedia Article Stack Overflow. http://en.wikipedia.org/wiki/Stack_Overflow (2014)
18. Wikipedia Article Reddit. http://en.wikipedia.org/wiki/Reddit (2014)
19. Zhang, J., Ackerman, M., Adamic, L.: Expertise networks in online communities: structure and algorithms. In: Proceedings of the 16th International Conference on World Wide Web, International World Wide Web Conference, Banff, Canada, 8–12 May, pp. 221–230. ACM Press, New York (2007)

Encoding Clinical Recommendations into Fuzzy DSSs: An Application to COPD Guidelines

Aniello Minutolo, Massimo Esposito and Giuseppe De Pietro

Abstract Clinical Decision Support Systems (DSSs) have been applied to medical scenarios by computerizing a set of clinical guidelines of interest, with the final aim of simulating the process followed by the physicians. In this context, fuzzy logic has been profitably used for modeling clinical guidelines affected by uncertainty and improving the interpretability of clinical DSSs through its expressivity close to natural language. However, the task of computerizing clinical guidelines in terms of fuzzy if-then rules can be complex and, often, requires technical capabilities not owned by physicians. In order to face this issue, this paper introduces a fuzzy knowledge editing framework expressly devised and designed to simplify the procedures necessary to codify clinical guidelines in terms of fuzzy if-then rules and linguistic variables. This framework is described with respect to a specific real case regarding the formalization of clinical recommendations extracted from the GOLD guidelines, which contain the best evidence for diagnosing and managing the Chronic Obstructive Pulmonary Disease.

Keywords Clinical DSS · Knowledge editor · Fuzzy rules

1 Introduction

Recently, decision support systems (DSSs) have been applied to medical scenarios by computerizing a set of clinical guidelines of interest, with the final aim of simulating the process followed by the physicians [4, 12]. In medical settings, the

A. Minutolo (✉) · M. Esposito · G. De Pietro
Institute for High Performance Computing and Networking,
ICAR-CNR, via P. Castellino, 111-80131 Naples, Italy
e-mail: minutolo.a@na.icar.cnr.it

M. Esposito
e-mail: esposito.m@na.icar.cnr.it

G. De Pietro
e-mail: depietro.g@na.icar.cnr.it

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_26

use of clinical guidelines has been more and more promoted since it is expected to effectively and efficiently foster medical practices and improve health outcomes when followed [13]. For this reason, clinical guidelines computerized and embedded into a clinical DSS represent an active knowledge resource that uses patient data to generate case-specific advices, supporting health professionals, the patients themselves, or others concerned about them [7].

One prerequisite for the broad usage and acceptance of clinical DSSs in medical settings is the guarantee of a high level of upgradability and maintainability in order to: (i) change clinical guidelines according to their evolution in terms of medical progresses in the treatment of individual diseases and (ii) adapt generic, site-independent guidelines to the specific patient to be treated. These issues point out two main needs: first, medical knowledge to be embedded into a clinical DSS should be represented by means of a well-defined and machine-readable language in order to enable easy formalization and management; second, doctors, health professionals, and other potential nontechnical users require intuitive and assisted solutions to directly manage the process of knowledge formalization into machine executable languages.

In the context of clinical DSSs, fuzzy logic [18] has been profitably used for modeling clinical guidelines affected by vagueness and improving the interpretability of clinical DSSs through its expressivity close to natural language [2, 6, 17]. In particular, by employing fuzzy logic, clinical guidelines can be explicitly formalized in the form of if-then rules by describing dependencies between premises and conclusions in a vague and human comprehensible way. In such a way, physicians are facilitated in understanding the advices generated by fuzzy if-then rules, even if they still remain the only ones responsible for the final decision about the actual actions to perform.

The main drawback of using a fuzzy DSS is that the task of computerizing clinical guidelines in terms of fuzzy if-then rules can be complex and, often, requires technical capabilities not owned by physicians.

In order to face this issue, this paper introduces a fuzzy knowledge editing framework expressly devised and designed to simplify the procedures necessary to computerize clinical guidelines in terms of fuzzy if-then rules and linguistic variables. This framework, already presented in [8], is here fully described with respect to a specific real case regarding the formalization of clinical recommendations extracted from the GOLD guidelines, which contain the best evidence for diagnosing and managing the chronic obstructive pulmonary disease (COPD).

In particular, graphical facilities offered by the framework are shown with reference to the encoding of two examples of GOLD guidelines by highlighting how to: (i) group together clinical recommendations to detect the positive evidence of a single abnormal situation, express them as fuzzy if-then rules, and specify a peculiar configuration for the inference; (ii) compose fuzzy ELSE rules for modeling the negative evidence when no abnormal situation has happened; (iii) simulate an actual DSS for testing the fuzzy rules codified; (iv) automatically encode these rules into a machine executable language that could be functional in the context of clinical DSSs.

The rest of the paper is organized as follows. Section 2 outlines an overview of the state-of-the-art solutions existing in the literature. In Sect. 3, the framework is introduced, by detailing its main features and the real case to which it is applied is depicted. Section 4 reports how the framework is used to encode guidelines pertaining to the real case considered. Finally, Sect. 5 concludes the work.

2 Related Work

Medical procedures can be affected by diverse forms of uncertainty, such as imprecision in monitored physiological parameters, vague thresholds for separating different health status, lack of clear functional dependencies between symptoms and clinical outcomes, and so on. This uncertainty, typically affecting clinical guidelines, can be modeled by means of fuzzy logic in the form of a fuzzy rule-based system. This latter is characterized by a set of parameters belonging to four main categories: *logical* (reasoning mechanism, fuzzy operators, membership function types, defuzzification method), *structural* (relevant variables, number of membership functions and rules), *connective* (antecedents and consequents of rules, rule weights), and *operational* (membership function values) [11].

Until now, several tools have been proposed for building understandable and accurate fuzzy rule-based systems by supporting users to define all the aforementioned parameters [3, 5, 10], even if most of them offer general-purpose facilities for modeling fuzzy knowledge, without any form of vertical arrangement for the particular domain of interest. Indeed, typically, the fuzzy knowledge base is arranged as a monolithic structure including all the rules, even if these latter are referred to different outcomes. In such a way, no form of grouping is conceived in order to better reproduce a logical partitioning of the whole set of rules depending on their outcome or on a particular inference configuration to be used. Moreover, the fuzzy knowledge base is usually built as complete, in the sense that it includes all the rules necessary to produce any possible outcome conceived in the domain of interest.

However, with specific reference to the medical domain, different peculiarities should be taken into account in order to best arrange a fuzzy DSS: (i) clinical guidelines often assume the form of sets of care recommendations associated to the same final outcome [14]; (ii) care recommendations can be linked in the sense that the outcome of a recommendation can be used as the input of another one (e.g., a recommendation about therapy necessarily depends on another one assessing the health status of the patient in care); (iii) more recommendations encoded in the form of linked rules should reasonably share the same inference configuration for producing their outcomes; (iv) recommendations are typically distilled in terms of incomplete rules, i.e., they do not cover all possible outcomes but just express only pieces of positive evidence (e.g., for determining the presence of a disease).

The solution proposed here has been conceived to face these issues in order to easily and directly support the construction of a clinical DSS by exploiting fuzzy logic, as described in the following section.

3 The Proposed Editing Framework for Encoding Guidelines into Fuzzy DSSs

The framework proposed here is aimed at providing simple and intuitive interfaces to computerize clinical guidelines in the form of fuzzy if-then rules, by enabling the encoding of all the aforementioned parameters necessary to build a fuzzy inference system underpinning a clinical DSS.

In more detail, several facilities have been designed in order to assist nontechnical users during the encoding of clinical guidelines as fuzzy rules: (i) each guideline, often made of a group of recommendations, can be encoded by means of an independent set of fuzzy rules which contribute, as a whole, to detect the positive evidence of a single abnormal situation; (ii) no ad hoc recommendation should be added to encode the negative medical evidence, i.e., when no abnormal situation has happened, since, to each group of recommendations belonging to a same guideline, an ELSE rule can be easily defined and associated.

From a more technical perspective, the framework offers functionalities to: (i) grant different levels of consistency, from the syntactical composition of a rule encoding a single clinical recommendation, by supporting the correct encoding in terms of admissible rule structure and well-formed statements, to its verification in terms of logic integrity by ensuring it fires when it should; (ii) automatically produce XML-based descriptions, according to the FdsL language [9], in order to represent the fuzzy knowledge base in format which can be machine-readable for building fuzzy DSSs and, at the same time, human comprehensible.

The choice of using an XML formalism for both configuring a fuzzy inference system and encoding its fuzzy medical knowledge is due to the possibility of supporting the seamless collaboration between users, and granting portability, extensibility and reuse, as highlighted in several studies [1, 15, 16]. For more details about the framework please refer to [8].

3.1 The Real Case: The GOLD Guidelines

The real case described here in order to highlight how the framework can be applied regards the encoding of clinical recommendations extracted from the GOLD guidelines. GOLD guidelines report the best evidence for diagnosing and managing the COPD formalized by the US National Heart, Lung, and Blood Institute and the World Health Organization in 2006.

Table 1 The GOLD guideline's recommendations for COPD severity

FEV_1/FVC ratio	FEV_1 predicted	Presence of CRF	COPD severity
<0.70	≥80 %	–	Mild
<0.70	Within [50 %, 80 %[	–	Moderate
<0.70	Within [30 %, 50 %[	–	Severe
<0.70	<30 %	–	Very severe
<0.70	<50 %	Yes	Very severe

COPD is characterized by the presence of significant airflow obstruction and some extrapulmonary effects that can lead to frequent hospitalization and eventually death from suffocation. COPD diagnosis is performed through the spirometry test which establishes the presence of airflow obstruction, by measuring the forced expiratory volume in one second (FEV_1) and the forced vital capacity ratio (FEV_1/FVC).

The case here considered regards two examples of GOLD guidelines described in the following. Table 1 reports a first guideline consisting of a set of five clinical recommendations, arranged in a tabular form, for diagnosing COPD and classifying its severity. In detail, each row represents a recommendation, the first three columns indicate the premises, i.e., the results of the spirometry test and the presence of chronic respiratory failure (CRF), whereas the last column specifies the outcome, i.e., corresponding level of severity identified.

On the other hand, Table 2 reports the second guideline consisting of a set of six clinical recommendations, arranged in a tabular form as well, for suggesting a specific treatment regarding the administration of influenza vaccine depending on the COPD severity. Also in this case, each row represents a recommendation, the first four columns indicate the premises, i.e., the results of the COPD severity assessment, the frequency of exacerbations, the presence of CRF, the patient age, whereas the last column specifies the outcome, i.e., the corresponding treatment suggested.

Table 2 The GOLD guideline's recommendations for the COPD treatment

COPD severity	COPD exacerbations	Presence of CRF	Patient age	COPD treatment
Mild, moderate, severe, very severe	–	–	≥65	Influenza vaccines
Very severe	–	–	<65	Influenza vaccines
Mild	–	–	–	Short-acting bronchodilators
Moderate, severe, very severe	–	–	–	Long-acting bronchodilators
Severe or very severe	Repeated	–	–	Inhaled glucocorticosteroids
Very severe	–	Yes	–	Long-term oxygen

As a result, depending on the COPD severity, this second guideline suggests the proper medical treatment to follow. For instance, the use of a vaccine to protect against influenza virus is recommended for COPD patients over 65 years and older, independently of their COPD severity, whereas the use of inhaled glucocorticosteroids is suggested only for COPD patients who have manifested repeated exacerbations, i.e., further amplification of the inflammatory response in their airways, and their COPD severity is severe or very severe.

All the recommendations included into the two guidelines considered are expressed by using precise thresholds. As a result, completely different outcomes could be suggested for patients with values, which are close but placed around the thresholds, so as to lead to possible wrong interpretations with respect to a direct evaluation of a physician. As a result, to encode such recommendations by means of a fuzzy rule-based DSS, the most common way is to construct fuzzy rules starting from them and, successively, codify their premises and conclusions as linguistic variables made of different fuzzy sets.

All the procedures applied by using the proposed framework are reported in the following section.

4 Encoding Clinical Recommendations into a Fuzzy DSS

The proposed framework enables to arrange the knowledge necessary to encode the aforementioned recommendations extracted from GOLD guidelines as *Declarative Knowledge* (DK) and *Procedural Knowledge* (PK).

DK includes fuzzy linguistic variables' modeling premises and outcomes occurring in the clinical recommendations specified in Tables 1 and 2, in terms of their structural parameters, such as name, universe of discourse, and the composing fuzzy sets. On the other hand, PK includes all the fuzzy rules, arranged to model the aforementioned recommendations, built on top of the linguistic variables defined.

It is important to note that the whole PK is partitioned into two groups of rules, each of them pertaining to one single clinical guideline. In such a way, it is possible to improve the level of maintainability of the whole knowledge base and, contextually, specify two different inference configurations for them. Differently, DK is shared between the two groups of rules.

The knowledge editing interface of the proposed framework has been used to encode both DK and PK (Fig. 1), starting from the recommendations reported in Tables 1 and 2.

In particular, in the upper zone of the DK area, all the linguistic variables, which have been used to codify the premises and conclusions of the two groups of clinical recommendations considered, have been specified, graphically arranged as a nested tree, where each fuzzy term is represented as a node under its owner linguistic variable. The fuzzification of the sharp boundaries identified in the clinical recommendations for each continuous premise has been done by means of trapezoidal

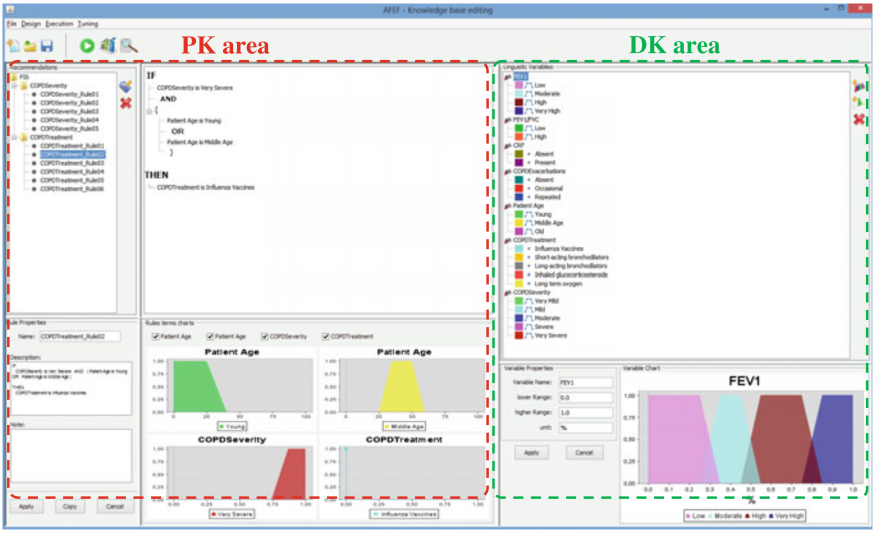


Fig. 1 The knowledge editing interface

fuzzy sets in the inner parts of the universe of discourse and semi-trapezoidal ones at its edges.

The parameters characterizing each trapezoidal fuzzy set have been chosen so that a uniform grade of vagueness around the initial crisp thresholds is achieved, i.e., these latter are considered as crossover points of two contiguous trapezoidal fuzzy sets. For instance, in the lower zone of the DK area shown in Fig. 1, the trapezoidal fuzzy sets associated to the linguistic variable FEV_1 are depicted.

Moreover, for each discrete premise, made of one or more nominal values (i.e., CRF or COPD exacerbation), fuzzy singletons are used, which are unity at some single particular points on the universe of discourse and zero everywhere else.

On the other hand, in the left-upper zone of the PK area shown in Fig. 1, two groups of fuzzy rules, graphically arranged as trees, have been depicted, where each tree is referred to a single aforementioned GOLD guideline. Moreover, in the right-upper zone of the PK area, the encoding of an example rule is graphically shown, where both the premises and conclusions are arranged as nested trees.

In more detail, the premises are graphically arranged as a root node "*IF*" where nested nodes represent, respectively, a rule statement, a logical conjunctive/disjunctive connector, or a couple of circle brackets to set a higher evaluation priority. The conclusion is graphically arranged as a root node "*THEN*", with only one rule statement as nested node, since each clinical recommendation is characterized by having a single outcome at most.

As a result, in order to encode the aforementioned clinical guidelines, the underpinning DK and PK have been graphically specified by means of the framework. The steps adopted to obtain the overall fuzzy knowledge base shown in Fig. 1 are detailed below.

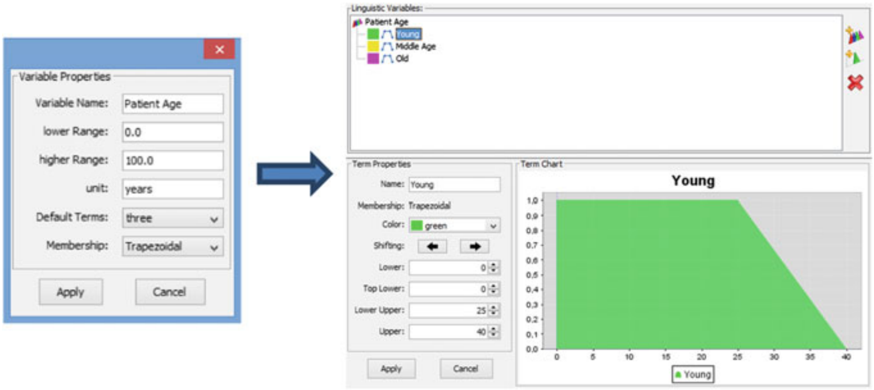


Fig. 2 Interfaces to create Declarative Knowledge

4.1 *Definition of Linguistic Fuzzy Variables and Fuzzy Sets*

The first step is the definition of DK, i.e., the linguistic variables occurring in the clinical recommendations to be modeled. As shown in Fig. 2, users are supported through a set of intuitive interfaces to define linguistic fuzzy variables (Fig. 2, left side) and fuzzy sets associated to them (Fig. 2, right side).

Each linguistic fuzzy variable has been defined by means of the *name*, the *lower* and *higher* values defining its universe of discourse, and the *unit* parameter expressing the meaning of its measured values. Regarding the number of fuzzy sets to be associated to each variable, they have been chosen depending on the typology (continuous or discrete) of premises/conclusions appearing in the clinical recommendations considered and their partitioning into intervals or discrete set of nominal values. Finally, shapes (i.e., trapezoidal and singleton) and their parameters have been chosen according to the considerations about fuzzification reported in the previous section.

In the example shown in Fig. 2, the definition of the linguistic fuzzy variable named *Patient Age* by means of the facilities offered by the framework is reported. This variable is also associated to three trapezoidal fuzzy sets, named *Young*, *Middle Age* and *Old*, respectively.

4.2 *Definition of Fuzzy if-then Rules Encoding Clinical Recommendations*

The second step is the definition of PK, i.e., the two groups of fuzzy if-then rules, which computerize the clinical recommendations considered. As shown in Fig. 3,

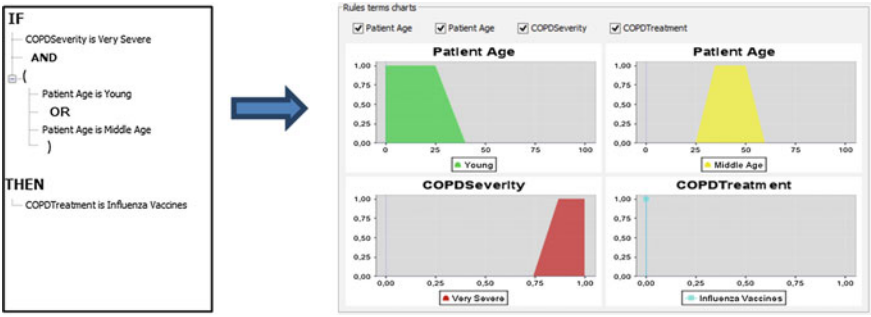


Fig. 3 Interfaces to create Procedural Knowledge

the framework provides functionalities for simply composing a fuzzy rule starting from the vocabulary of linguistic fuzzy variables and fuzzy sets defined.

In more detail, the statements composing the premises and conclusions of the fuzzy rules have been edited by using a simplified and intuitive drag-and-drop technique, which forces the user to always select a fuzzy set in the DK area so as to grant the final correctness.

For example, the statement "*COPDSeverity is Very Severe*" has been added by selecting the fuzzy set "*Very Severe*" in the nested tree under the linguistic fuzzy variable "*COPDSeverity*", and then, by dragging and dropping it in the area under the root "IF". Moreover, the composite expression "*(Patient Age is Young OR Patient Age is Middle Age)*" has been inserted by simply adding circle brackets in the rule premises, selecting the "*OR*" logical connector to use in them and dragging and dropping the fuzzy sets "*Young*" and "*Middle Age*" to create the rule statements "*Patient Age is Young*" and "*Patient Age is Middle Age*", respectively.

Similarly, also all the other fuzzy rules computerizing the clinical recommendations outlined in Tables 1 and 2 have been inserted.

Coherently with the philosophy of the proposed framework, these fuzzy rules have been grouped depending on the clinical guideline they belong to. In particular, all the rules belonging to a group share the same linguistic variables in their conclusions. For example, the rules outlined in Fig. 4 are grouped and associated to the linguistic variable *COPDSeverity,* since all the recommendations they model have the variable *COPDSeverity* in their conclusions.

As a result, the PK made by all the recommendations has been graphically arranged as a nested tree (see the left area of Fig. 4), where each rule is placed under the corresponding group through a father–child relation.

Every group of rules has been designed as a self-contained inference node, i.e., the fuzzy inference has been appropriately configured in order to best fit the peculiarities of the corresponding guideline considered.

This configuration has been performed for each group of rules by setting the operational parameters reported in the right area of Fig. 4, i.e., the identification *Name*, the *Output* variable, specified in all the conclusions of the rules belonging to

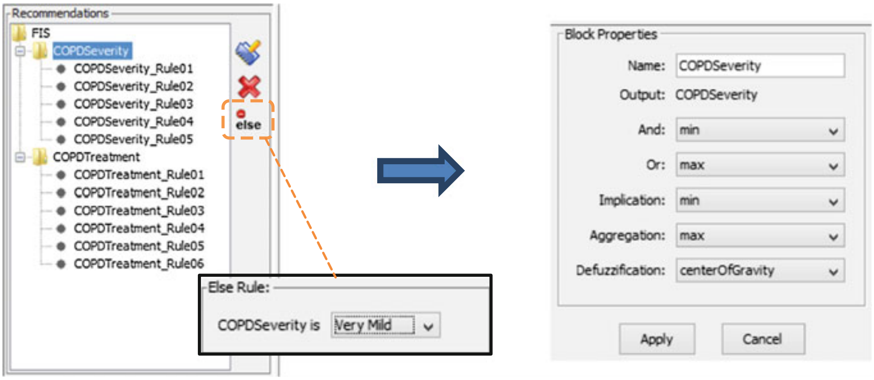


Fig. 4 The group of clinical recommendations linked to the linguistic variable *COPDSeverity*

that group, the methods used for the *Implication, Aggregation* and *Defuzzification*, the operators to be used for expressing the logical connective *And* and *Or*.

An ELSE rule has been finally added to the first group of rules, since the recommendations reported in Table 1 do not consider the case when the forced vital capacity ratio (FEV_1/FVC) is greater than 0.70, which corresponds to the absence of COPD or a very mild level of severity. As reported in the left area of Fig. 4, this ELSE rule has been associated to the first group of rules for modeling its negative evidence by simply pressing the "*else*" button.

All the fuzzy knowledge base encoded as just described has been finally translated automatically into a set of statements according to syntax of the FdsL language, so that users are not required to know and use its constructs.

4.3 Evaluation of the Fuzzy Knowledge Base Generated

The fuzzy knowledge base graphically inserted by means of the proposed framework and translated into the FdsL language has been submitted to a fuzzy reasoning engine, embedded into the framework itself, which simulates an actual fuzzy DSS for testing its behavior.

In particular, to test the integrity of the logic and ensure the fuzzy rules fire when they should, a collection of input data has been supplied to the framework in order to be evaluated. Input data can be both inserted manually through the specific interfaces or imported from FdsL descriptions.

Figure 5 depicts an example of input data inserted for the evaluation. In particular, for each fuzzy linguistic variable defined in the fuzzy knowledge base and involved in the premises of a rule, a crisp value has been set.

Once the input dataset has been created, it has been submitted to the inference engine through the *Start Reasoning* button of the framework, after being automatically codified into the FdsL language. At the end of the inference process, the

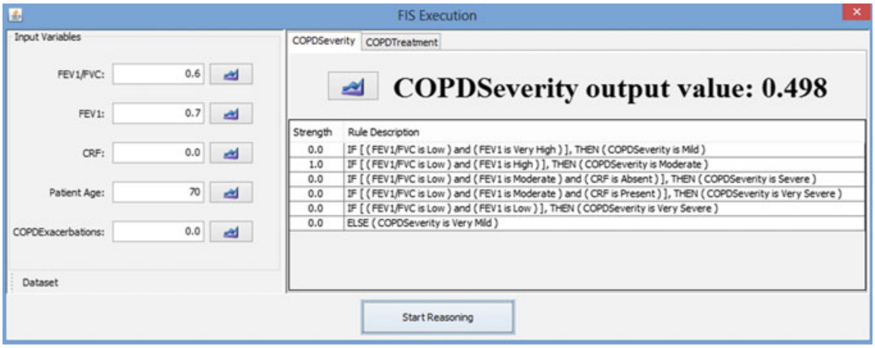


Fig. 5 The interfaces for specifying the input dataset to be evaluated and the results generated

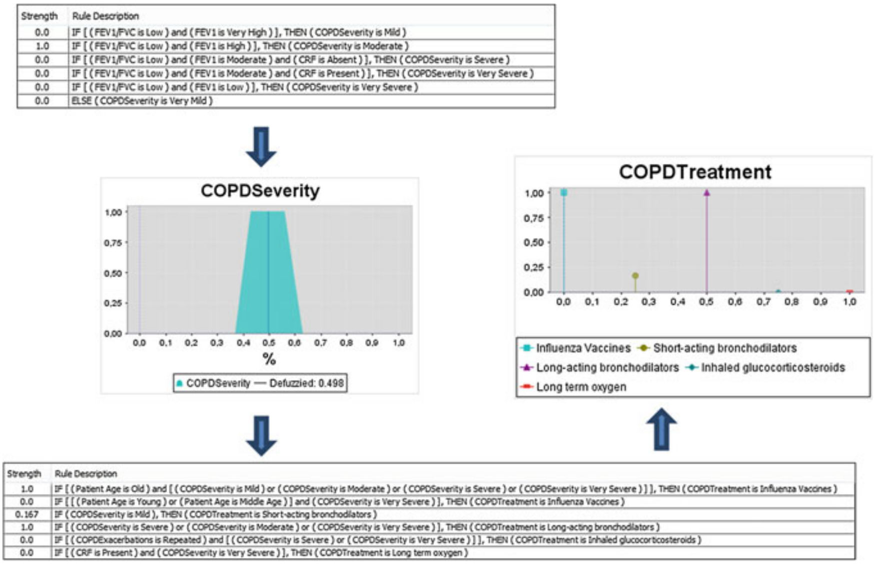


Fig. 6 The inferred fuzzy evidence for *COPD severity* and, afterwards, for *COPD treatment*

framework has reported the strength for each rule belonging to a group. The strength of a rule represents its activation degree, i.e., the degree of match of its premises by properly composing the degrees of match of the involved conditions in accordance with the logical operators connecting them. This strength indicates the level of confidence in the decision suggested by each rule, and the accuracy of reasoning outcomes.

Figure 6 reports the activation degrees computed for the two groups of rules starting from the crisp values reported in Fig. 5, and the inferred fuzzy evidence for the *COPDSeverity* and, afterwards, for the *COPDTreatment*.

In particular, the crisp output value assumed by the *COPDSeverity* variable is inferred as equal to 0.498 after evaluating the first group of rules. Such a value has been used, together with the other ones inserted by means of the graphical interface offered by the framework, to evaluate the second group of rules and infer, as a result, that the COPD treatment can assume two possible values with a maximum strength, i.e., *Influenza Vaccines* and *Long-acting bronchodilators*.

The final decision about how treatment has to be administrated to the patient is left to the physician, who can decide for only one of them or also for both in the case when they are not mutually exclusive and there are no contraindications about their contextual administration.

5 Conclusions

This paper described a fuzzy knowledge editing framework expressly devised and designed to simplify the procedures necessary to computerize clinical guidelines in terms of fuzzy if-then rules and linguistic variables. This framework, already presented in [8], has been fully described here with respect to a specific real case regarding the formalization of clinical recommendations extracted from the GOLD guidelines, which contain the best evidence for diagnosing and managing COPD.

Differently from existing solutions, which do not provide any form of vertical arrangement for the particular domain of interest, the proposed framework supports users to: (i) group together clinical recommendations to detect the positive evidence of a single abnormal situation, express them as fuzzy if-then rules, and specify a peculiar configuration for the inference; (ii) compose fuzzy ELSE rules for modeling the negative evidence when no abnormal situation has happened; (iii) simulate an actual DSS for testing the fuzzy rules codified; (iv) automatically encode these rules into a machine executable language that could be functional in the context of clinical DSSs.

The next step of the research activities will also regard the improvement of the proposed framework by means of new intuitive facilities to automatically optimize the parameters of the fuzzy inference system underpinning a clinical DSS, such as data-driven techniques for tuning the knowledge base, and more advanced verification facilities to evaluate the consistency of the clinical guidelines encoded as fuzzy rules.

References

1. Acampora, G., Loia, V.: Fuzzy markup language: a new solution for transparent intelligent agents. In: IEEE Symposium on Intelligent Agent, pp. 1–6 (2011)
2. Adeli, A., Neshat, M.: A fuzzy expert system for heart disease diagnosis. In: International Multiconference of Engineering and Computer Scientists, pp. 134–139 (2010)

3. Alonso, J.M., Magdalena, L., Guillaume, S.: KBCT: a knowledge extraction and representation tool for fuzzy logic based systems. In: IEEE International Conference on Fuzzy Systems, vol. 2, pp. 989–994. Budapest, Hungary (2004)
4. De Maio, C., Fenza, G., Gaeta, M., Loia, V., Orciuoli, F.: A knowledge-based framework for emergency DSS. Knowl.-Based Syst. **24**(8), 1372–1379 (2011)
5. Guillaume, S., Charnomordic, B.: Learning interpretable fuzzy inference systems with FisPro. Inf. Sci. **181**(20), 4409–4427 (2011)
6. Lahsasna, A., Ainon, R.N., Zainuddin, R., Bulgiba, A.: Design of a fuzzy-based decision support system for coronary heart disease diagnosis. J. Med. Syst. **36**(5), 3293–3306 (2012)
7. Liu, J., Wyatt, J.C., Altman, D.G.: Decision tools in health care: focus on the problem, not the solution. BMC Med. Inf. Decis. Mak. **6**, 1–7 (2006)
8. Minutolo, A., Esposito, M., De Pietro, G.: A fuzzy knowledge-editing framework for encoding guidelines into clinical DSSs. In: 8th International Conference on Knowledge, Information and Creativity Support Systems (KICSS), pp. 379–390 (2013)
9. Minutolo, A., Esposito, M., De Pietro, G.: A fuzzy decision support language for building mobile DSSs for healthcare applications. In: Godara, B., Nikita, K.S. (eds.) Wireless Mobile Communication and Healthcare. LNICST, vol. 61, pp. 263–270. Springer, Heidelberg (2013)
10. Moreno-Velo, F.J., Baturone, I., Sanchez-Solano, S., Barriga, A.: XFuzzy 3.0 A development environment for fuzzy systems. In: 2nd IEEE International Conference on Fuzzy Logic and Technology (EUSFLAT), pp. 93–96 (2001)
11. Peña-Reyes, C.A., Sipper, M.: A fuzzy-genetic approach to breast cancer diagnosis. Artif. Intell. Med. **17**(2), 131–155 (1999)
12. Sari, A.K., Rahayu, W., Bhatt, M.: Archetype sub-ontology: Improving constraint-based clinical knowledge model in electronic health records. Knowl. Base Syst. **26**, 75–85 (2012)
13. Scott, I.: What are the most effective strategies for improving quality and safety of health care? Intern. Med. J. **39**(6), 389–400 (2009)
14. Shiffman, R.N.: Representation of clinical practice guidelines in conventional and augmented decision tables. J. Am. Med. Inf. Assoc. **4**(5), 382–393 (1997)
15. Thomas, O., Dollmann, T.: Fuzzy-EPC markup language: XML based interchange formats for fuzzy process models. Soft Comput. XML Data Manage. **255**, 227–257 (2010)
16. Tseng, C., Khamisy, W., Vu, T.: Universal fuzzy system representation with XML. Comput. Stand. Interfaces **28**(2), 218–230 (2005)
17. Warren, J., Beliakov, G., Zwaag, B.: Fuzzy logic in clinical practice decision support system. In: 33rd Annual Hawaii International Conference on System Sciences, pp. 4–7 (2000)
18. Zadeh, L.A.: A theory of approximate reasoning. In: Hayes, J., Michie, D., Mikulich, L.I. (eds.) Machine Intelligence, vol. 9, pp. 149–194. Elsevier, New York (1979)

Approximation of Statistical Information with Fuzzy Models for Classification in Medicine

Marco Pota, Massimo Esposito and Giuseppe De Pietro

Abstract Fuzzy logic has gained increasing importance in Decision Support Systems (DSSs), in particular in medical field, since it allows to build a transparent and interpretable knowledge base. However, in order to obtain a general description of a system, probabilistic approaches undoubtedly offer the most significant information. Moreover, a good classifier to be used for medical scopes should be able to: (i) classify data items which are lacking of some input features; (ii) extract knowledge from incomplete datasets; (iii) consider categorical features; (iv) give responses in terms of a set of possible classes with respective degrees of plausibility. The approach here proposed pursues and achieve these objectives by approximating probabilistic information from incomplete datasets with an interpretable fuzzy system for classifying medical data. Resulting fuzzy sets can be interpreted as the terms of the involved linguistic variables, corresponding to numerical and/or categorical features, while weighted rules model probabilistic information. Rules are presented in two forms: the first is a set of one-dimensional models, which can be used if only one input feature is known; the second is a multidimensional combination of them, which can be used if more input features are known. As a proof of concept, the method has been applied for the detection of Multiple Sclerosis Lesions. The results show that this method is able to construct, for each one of the variables influencing the classification, an interpretable fuzzy partition, and very simple *if-then* rules. Moreover, multidimensional rule bases can be constructed, by means of which improved results are obtained, also with respect to naive Bayes classifier.

M. Pota (✉) · M. Esposito · G. De Pietro
Institute for High Performance Computing and Networking (ICAR),
National Research Council of Italy, Via P. Castellino 111, 80131 Naples, Italy
e-mail: marco.pota@na.icar.cnr.it

M. Esposito
e-mail: massimo.esposito@na.icar.cnr.it

G. De Pietro
e-mail: giuseppe.depietro@na.icar.cnr.it

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_27

Keywords Probability · Statistical learning · Fuzzy partition · Rule extraction · Classification · Linguistic variable · Clinical DSS

1 Introduction

Decision Support Systems (DSSs) are gaining increasing importance in recent research. In particular, in some fields of application like medicine, DSSs are often required to manage data affected by uncertainty and vagueness, and to offer a transparent and comprehensible knowledge base. These issues can be accomplished by using fuzzy logic [15]: on the one hand, fuzzy sets can model terms of linguistic variables; on the other hand, fuzzy rules show a clear and logic justification for each conclusion. Therefore, classification problems can be tackled by modeling medical knowledge acquired from actual data, whose classification is known (training set), in order to classify new incoming data items accordingly, by using a fuzzy knowledge base.

A medical datasets is basically constituted by a set of patterns, each made of a set of values of features, which can be extracted from various sources and therefore can be of various types: numerical, fuzzy/linguistic, categorical. Moreover, especially in the medical field, where some data could be difficult to acquire, both the training dataset and the data items to be classified could be missing of some feature values.

The knowledge modeling consists in the definition of a collection of fuzzy sets, and the construction of a rule base correlating them to classes, by means of rules defined according to the *if-then* structure. Different approaches can be used to extract the useful knowledge [1, 3, 6–9, 13, 14]. Some approaches are useful to construct fuzzy sets which are interpretable, since fixing some constraints let them able to model the terms of a linguistic variable; however, they may suffer from the drawback that membership functions do not approximate the real trend of data. Other approaches tend to maximize the "goodness" of the final DSS; however, good fitting requires no constraint, therefore resulting fuzzy sets may not be able to model interpretable terms of a linguistic variable. These two objectives (interpretability and good performances) can be jointly reached by assigning weights to rules. However, even if an optimization of interpretable fuzzy sets and weighted rules is performed, the "goodness" of the system is usually defined in terms of some measures (e.g., the classification rate) which only take into account the single class chosen by the system for a given data item, while the uncertainty of the response can also be very informative and important, especially in fields like medicine, where all possible classes should be taken into account. In general, probabilistic approaches could offer the most useful information (i.e., given an input data item, what are the probabilities of different classes). However, some approaches based on a Bayesian perspective [2] also present results in terms of only the most probable class; moreover, they do not offer transparent and interpretable systems, like fuzzy rule-based systems. Another open problem regards the possibility of classifying data items made of some unknown features and both numerical and categorical features.

This work describes an assessed version of the approach proposed by the authors in [12], which is aimed at approximating probabilistic information with an interpretable fuzzy system for classifying medical data. It essentially consists of three steps. First, a statistical approach is used to obtain one-dimensional models of the probabilities of classes, given a generic value of each input feature. Then, these mappings are correlated to constrained fuzzy sets, in order to obtain interpretable partitions as well as a set of weighted rules, by means of which it is possible to calculate probabilities of different classes, for data to be classified which are characterized by only one known value of features. Finally, one-dimensional models are combined in order to obtain multidimensional models, which are more accurate to classify data items characterized by more than one known value of features. The approximation step is here deeply refined with respect to [12], and some aspects are underlined, with particular regard to incomplete datasets, made of mixed numerical/categorical features. As a result, four objectives are reached at the same time: (*i*) the extraction of knowledge from incomplete datasets, made of both numerical and categorical features; (*ii*) the construction of a transparent and interpretable fuzzy rule-based system; (*iii*) the possibility of managing lack of knowledge about data items to be classified; (*iv*) the kind of results produced (probabilities of different classes), which is significantly attractive in the medical field.

The rest of the paper is organized as follows. Section 2 summarizes some basic concepts regarding knowledge-based systems. In Sect. 3, the proposed approach is explained, while its application to Multiple Sclerosis Lesions (MSL) classification is shown in Sect. 4. Finally, Sect. 5 concludes the work.

2 Background

This section summarizes some basic concepts about fuzzy inference and knowledge-based systems, which are useful bases for developing the proposed method.

2.1 Fuzzy Inference

A simple fuzzy classifier comprises a set of fuzzy rules. In each rule, the antecedent defines a region of the *H*-dimensional feature space, while the consequent is a class:

$$\mathrm{r}_i\text{: } \textit{If } X^{(1)} \textit{ is } F_i^{(1)} \textit{ and} \ldots \textit{and } X^{(H)} \textit{ is } F_i^{(H)} \textit{ then } Y \textit{ is } c_k, \tag{1}$$

where $X^{(1)}$, …, $X^{(H)}$ are the input features, $F_i^{(1)}$, …, $F_i^{(H)}$ are the antecedent fuzzy sets, and Y is the output variable which assumes values belonging to the set $\{c_1, \ldots, c_C\}$. A weight w_i can be associated to each rule in order to model a different impact, which can be defined by an expert or calculated using an optimization process. The *and* connective is modeled by a T-norm, applied to the set of membership grades $\mu^{(h)}$, of values $x^{(h)}$ assumed by features, to fuzzy sets $F^{(h)}$, $h = 1,\ldots, H$. Therefore, the degree of activation of the ith rule is:

$$\alpha_i(\mathbf{x}) = w_i \cdot \mathrm{T} - \mathrm{norm}\left[\mu_i^{(1)}\left(x^{(1)}\right), \ldots, \mu_i^{(H)}\left(x^{(H)}\right)\right], \tag{2}$$

where $\mathbf{x}$ is the vector of values $\{x^{(1)},\ldots, x^{(H)}\}$.

If different rules have the same class as consequent (this can also be modeled by *or* connectives in the same rule), an S-norm is applied on activations of these rules to obtain the activation A_k of the kth class c_k:

$$A_k(\mathbf{x}) = \mathrm{S} - \mathrm{norm}[\alpha_1(\mathbf{x}), \ldots, \alpha_{R_k}(\mathbf{x})], \tag{3}$$

where R_k is the number of rules with the same consequent c_k.

The output of the fuzzy classifier is usually determined by the "winner takes all" strategy, i.e., the output is the class that gets the highest degree of activation.

2.2 Knowledge Extraction

The fuzzy knowledge extraction is the construction of a model which allows to perform the decision-making process based on fuzzy logic. When the knowledge extraction is configured as data-driven, it is made by using a training dataset, and consists of different aspects, namely feature selection, fuzzy partitioning, rule base extraction, and, eventually, optimization. Feature selection aims at individuate the features which are significant for classification [5]. Fuzzy partitioning consists of the construction of membership functions of fuzzy sets to be used in the rules, and can be accomplished by using different methods [1, 7–9], which are usually unsupervised and aim at the interpretability of fuzzy sets as terms of linguistic variables. The rule base extraction consists of combining fuzzy sets in antecedents of *if-then* rules, and assigning each rule a consequent, and can be performed by dedicated methods [6, 13]. Some methods also exist which are used to obtain fuzzy partitions and a rule base at the same time [3, 14], and this supervised learning allows the calculation of fuzzy sets according to different classes of training data items, so as to result more useful for classification. However, these procedures are usually devoted to maximize only one of the following objectives: (*i*) some measure of interpretability of the knowledge base [4], (*ii*) some measure of the performances of the classifier [5, 7]. As a consequence, the best optimization procedure depends on the objectives of the classifier.

Statistical approaches [2] are particularly useful to optimize the performances of the system, since they are mainly based on Bayes' rule:

$$\mathbf{x}\, \textit{is assigned to}\, c_k \Leftrightarrow p(c_k|\mathbf{x}) \geq p(c_i|\mathbf{x}) \quad \forall\, i \neq k. \tag{4}$$

In particular, the naive Bayes classifier reaches a very simple form by assuming independence of input variables, thus obtaining:

$$\mathbf{x}\ \textit{is assigned to}\ c_k \Leftrightarrow P(c_k)\prod_h p\left(x^{(h)}|c_k\right) \geq P(c_i)\prod_h p\left(x^{(h)}|c_i\right) \quad \forall\, i \neq k, \qquad (5)$$

where class probability distributions $p(x^{(h)}|c_k)$ are often assumed to have fixed shape, defined by parametric probability density functions (e.g., Gaussian PDFs), while posterior probabilities $p(c_k|\mathbf{x})$ are less appropriate to be modeled by imposing such an assumption. Class probability distributions may be used to construct fuzzy partitions [10]. However, this method only allows to choose the most probable among classes, while in fields like medicine, the user should consider all plausible classes, together with respective probability (or possibility). Therefore, one should calculate an approximate mapping of $p(c_i|\mathbf{x})$ as a function of $\mathbf{x}$. This information can be extracted from data and can be used to build fuzzy partitions and rules [11]. However, the resulting fuzzy sets, constructed by using either class probability distributions [10] or posterior probabilities [11], could not be fully interpretable.

Fuzzy sets are interpretable if they model terms of a linguistic variable. Therefore, they should be convex, and satisfy, for each feature, the properties of normality:

$$\forall t \in \{1, \ldots, T\}, \begin{cases} \min\{\mu_t(x), x \in U\} = 0 \\ \max\{\mu_t(x), x \in U\} = 1 \end{cases}, \qquad (6)$$

and orthogonality:

$$\forall x \in U,\ \sum_{t=1}^{T} \mu_t(x) = 1\,, \qquad (7)$$

where x is the value of a feature, U is the universe of discourse for that feature, $\mu_t(x)$ is the value of the membership function of the tth fuzzy set, and T is the number of linguistic terms.

Membership functions $\mu_t(x)$ are often represented by parametric functions, which could have different shapes (triangular, trapezoidal, bell-shaped, and so on).

3 Combination of Interpretable Fuzzy Models and Probabilistic Inference

In this section, the assessed version of the approach presented in [12] is detailed, with a particular emphasis on the new approximation algorithm used, and on the ability of extracting knowledge from training datasets with missing data and mixed numerical/categorical features. In more detail, the approach allows to extract from a training dataset a knowledge base made of interpretable fuzzy sets and fuzzy rules, in order to calculate probabilities of classes, given incoming data items, made of patterns of feature values, eventually with some unknown values.

Each one of the N data items of the training set is made of an input pattern $\mathbf{x}_j = [x_j^{(1)},\ldots, x_j^{(H)}]$, $j = 1,\ldots, N$, where H is the number of features, and a corresponding class $y_j \in \{c_1,\ldots, c_C\}$, where C is the number of different classes. A classifier aims at modeling the relations between $\mathbf{x}_j$ and y_j, in order to predict the class y to which a new incoming pattern $\mathbf{x} = [x^{(1)},\ldots, x^{(H)}]$ should be assigned. The features $X^{(h)}$ can be numerical (continuous or discrete), or take values in a set of τ not ordered categories x_t, $t = 1,\ldots, \tau$. In the following, the symbol x will substitute $x^{(h)}$, where only one dimension is considered.

The approach described here consists of three steps. First, the one-dimensional functions describing posterior probabilities of classes, $p(c_k|x)$ are calculated starting from the training dataset. Then, these functions are approximated with a combination of interpretable fuzzy sets. Finally, one-dimensional weighted rule bases are obtained, and combined to construct a multidimensional model comprising the desired features, in order to classify incoming data with known values of these features.

3.1 Functions Describing Posterior Probabilities

In order to calculate posterior probability $p(c_k|x)$ as nonparametric function of x, the following procedure is adopted:

1. An equally spaced grid is used of a number n of points x_ν, with $\nu = 1,\ldots, n$, representing the whole universe of discourse U of the variable X. Therefore, each of them represents an interval whose measure is M/n, where M is the measure of the whole U, calculated as $\max\{x_j\}$-$\min\{x_j\}$.
2. Kernel functions are used to calculate the probability $p(x_\nu|c_k)$ of each point x_ν, given that the class is c_k:

$$p(x_\nu|c_k) = \frac{M}{n}\frac{1}{N_k h}\sum_{j=1}^{N_k} K\left(\frac{x_\nu - x_j}{h}\right), \tag{8}$$

   where N_k is the number of data items x_j belonging to class c_k, h is a smoothing parameter called bandwidth, and $K(\xi)$ is a normalized symmetric kernel function.
3. Probability of each class, in correspondence of grid points, is calculated by using Bayes' theorem:

$$p(c_k|x_\nu) = \frac{p(x_\nu|c_k)P(c_k)}{\sum_{i=1}^{C} p(x_\nu|c_i)P(c_i)}, \tag{9}$$

   where prior probabilities $P(c_i)$ are given by N_i/N.

If the feature is of categorical type, the grid corresponds to the set of categories, $n = \tau$, and the probability of a class in correspondence of each category $p(c_k|x_t)$ is calculated as the fraction of data items corresponding to class c_k with respect to the total data items of category x_t.

In this step, only and all training data items with known values of each involved feature are required, thus overcoming the possible lack of knowledge consisting in some unknown values in the training dataset.

3.2 Transformation into Interpretable Fuzzy Sets

The assumption is made here that any nonparametric function describing the posterior probability of a class can be approximated by a linear combination of parametric membership functions $\mu_t(x)$ of a number T of normal, orthogonal, and convex fuzzy sets F_t:

$$p(c_k|x) \cong \sum_{t=1}^{T} \lambda_{t-k}\mu_t(x)\,. \tag{10}$$

In particular, membership functions are proposed to be generated by using sigmoid functions:

$$\begin{cases} \mu_1(x) = 1 - \dfrac{1}{1+\exp(a_1 x + b_1)} \\ \mu_t(x) = \dfrac{1}{1+\exp(a_{t-1} x + b_{t-1})} - \dfrac{1}{1+\exp(a_t x + b_t)}\,, \\ \mu_T(x) = \dfrac{1}{1+\exp(a_{T-1} x + b_{T-1})} \end{cases} \tag{11}$$

where the second equation is assumed for $t \in \{2,\ldots, T-1\}$ and is not considered if $T = 2$. It is simple to verify that these fuzzy sets are orthogonal. If constraints are assigned to parameters, they are also convex and approximately normal.

The problem of approximating the function describing posterior probabilities by using (10) is an optimization problem which can be solved by any known algorithm. The parameters to optimize are a_t and b_t, with $t = 1,\ldots, T-1$, and λ_{t-k}, with $t = 1,\ldots, T$ and $k = 1,\ldots, C$. Noticing that coefficients of (10) must satisfy the following condition:

$$\forall t,\ \sum_{k=1}^{C} \lambda_{t-k} = 1\,, \tag{12}$$

then the number of parameters to optimize is $2T + TC$-2, therefore N should be sufficiently higher than it. The computational effort increases with C, also because one has to approximate C-1 functions at the same time, to find all parameters. The number T of linguistic terms is chosen by considering the weighted root mean square error of the approximation:

$$WRMSE = \sqrt{\sum_{\nu=1}^{n} \omega_\nu \left(p(c_k|x_\nu) - \sum_{t=1}^{T} \lambda_{t-k} \mu_t(x) \right)^2}, \tag{13}$$

where each squared error of the approximation is weighted to take into account the importance of well approximating each grid point, depending on its probability:

$$\omega_\nu = p(x_\nu). \tag{14}$$

The algorithm starts from $T = 2$ and increases T if *WRMSE* exceeds a threshold *ErrMax*. The last added term is removed if *WRMSE* decreases of a fraction lower than *ErrDecMin*. No more term is added if $T = Tmax$. The choice of $Tmax \leq 5$ is recommended for interpretability and for the computational effort of the optimization.

As a result, an interpretable fuzzy partition is obtained, and probability can be calculated as a linear combination of membership grades.

If the feature is of categorical type, $T = \tau$, fuzzy sets can be considered as singletons corresponding to the categories, and the coefficients are equal to the posterior probabilities calculated above: $\lambda_{t\text{-}k} = p(c_k|x_t)$.

3.3 Rule Base Construction and Inference

The proposed approach allows to construct a rule base made of the following set of very simple rules, $\forall t \in \{1,\ldots, T\}$, $\forall k \in \{1,\ldots, C\}$,

$$\mathrm{r}_{t-k}: \textit{If X is } F_t \textit{ then Y is } c_k, \tag{15}$$

where each rule is associated with the respective weight $\lambda_{t\text{-}k}$. In order to present the rule base in a more compact form, (15) can be written, $\forall t \in \{1,\ldots, T\}$, as

$$\mathrm{r}_t: \textit{If } X \text{ is } F_t \text{ then } Y \text{ is} \begin{cases} \lambda_{t-1} c_1 \\ \ldots \\ \lambda_{t-C} c_C \end{cases}, \tag{16}$$

where each $\lambda_{t\text{-}i}$ can be seen as the probability of having the class c_i, given that the fuzzy set is F_t.

As a result, if a bounded sum is used as S-norm (strong disjunction in Łukasiewicz fuzzy logic) for aggregating activations of rules which have the same consequence, then the activation of each class is equal to the probability of the same class, as can be seen by considering (3), (2), and (10):

$$A_k(x) = \min\left[1,\ \sum_{t=1}^{T} \alpha_{t-k}\right] = \min\left[1,\ \sum_{t=1}^{T} \lambda_{t-k}\mu_t(x)\right] \cong p(c_k|x)\,. \tag{17}$$

Once probabilities $p(c_k|x^{(h)})$ have been calculated for a new data item, using one-dimensional models for each one of the H known input variables, they can be combined in a “naive” way, by assuming independence of input variables:

$$p(c_k|\mathbf{x}) = \frac{P(c_k)p(\mathbf{x}|c_k)}{p(\mathbf{x})} = \frac{P(c_k)\prod_h p\left(x^{(h)}|c_k\right)}{p(\mathbf{x})} = \frac{\prod_h p\left(c_k|x^{(h)}\right)}{P(c_k)^{H-1}}. \tag{18}$$

Alternatively, different models can be aggregated to build a unique model comprising all known variables and giving the same results, made by the set $\forall\{t_1, \ldots, t_H\} \in \{1, \ldots, T_1\} \times \cdots \times \{1, \ldots, T_H\}$, of rules

$$\mathrm{r}_{\{t_1,\ldots,t_H\}}: \textit{If } X^{(1)} \textit{ is } F_{t_1} \textit{ and } \ldots \textit{ and } X^{(H)} \textit{ is } F_{t_H} \textit{ then } Y \textit{ is} \begin{cases} w_{\{t_1,\ldots,t_H\}-1} & c_1 \\ \ldots & \\ w_{\{t_1,\ldots,t_H\}-C} & c_C \end{cases}, \tag{19}$$

where $T_h \in \{T_1,\ldots,T_H\}$ is the number of terms of the hth linguistic variable, the activation of each rule is calculated by (2) with product T-norm (strong conjunction in product fuzzy logic), and the weights are given by:

$$w_{\{t_1,\ldots,t_H\}-k} = \frac{\lambda_{t_1-k}\cdot\ \ldots\ \cdot\lambda_{t_H-k}}{P(c_k)^{H-1}}\,. \tag{20}$$

In order to present this rule base in an intelligible form, weights of (19) can be split into a weight to assign to the rule and different probabilities of classes:

$$W_{\{t_1,\ldots,t_H\}} = \sum_{k=1}^{C} w_{\{t_1,\ldots,t_H\}-k} \text{ and } p_{\{t_1,\ldots,t_H\}-k} = \frac{w_{\{t_1,\ldots,t_H\}-k}}{W_{\{t_1,\ldots,t_H\}}}\,, \tag{21}$$

thus obtaining the set $\forall\{t_1, \ldots, t_H\} \in \{1, \ldots, T_1\} \times \cdots \times \{1, \ldots, T_H\}$, of rules

$$\mathrm{r}_{\{t_1,\ldots,t_H\}}: \textit{If } X^{(1)} \textit{ is } F_{t_1} \textit{ and } \ldots \textit{ and } X^{(H)} \textit{ is } F_{t_H} \textit{ then } Y \textit{ is} \begin{cases} p_{\{t_1,\ldots,t_H\}-1} & c_1 \\ \ldots & \\ p_{\{t_1,\ldots,t_H\}-C} & c_C \end{cases}, \tag{22}$$

with respective associated weights $W_{\{t_1,\ldots,t_H\}}$.

4 Application to Multiple Sclerosis Lesions and Indian Liver Patient Dataset

The proposed method has been applied for the construction of a real DSS with the aim of classifying MSLs, using an experimental training dataset, made of patterns of values of four features, referred to tissues identified in Magnetic Resonance images of brain: WM is the fraction of white matter surrounding a tissue, SF is a shape factor, DF is a distance factor measuring differences of colors, and VN is the volumetric dimension of the lesion expressed in number of voxels. Each pattern is labeled, since the classification of the related tissue as normal brain tissue (NBT) or white matter potential lesion (WMPL) is known.

First, each feature significance was evaluated by the Wilcoxon–Mann–Whitney test, with significance threshold of 0.05, and all features resulted significant. Then, by means of the procedure explained in Sect. 3.1, with a grid number $n = 100$, a bandwidth $h = M/25$, and standard Gaussian kernel functions, the posterior probabilities $p(\text{NBT}|x)$ and $p(\text{WMPL}|x)$ of classes NBT and WMPL, were found as nonparametric functions of each significant feature, as shown in Fig. 1. These functions were approximated by using the procedure explained in Sect. 3.2, with $ErrMax = 0.02$, $ErrDecMin = 0.1$ and $Tmax = 5$, as shown in Fig. 1 as well. As a consequence, a fuzzy partition was obtained for each feature, as shown in Fig. 2.

For each significant feature, a fuzzy rule base was extracted as explained in Sect. 3.3. In Table 1, the weights of the rules of different one-dimensional rule bases are reported, and in Table 2, the rule base obtained from (16), if the variable SF is considered, is presented as an example. Each rule base can be used to make

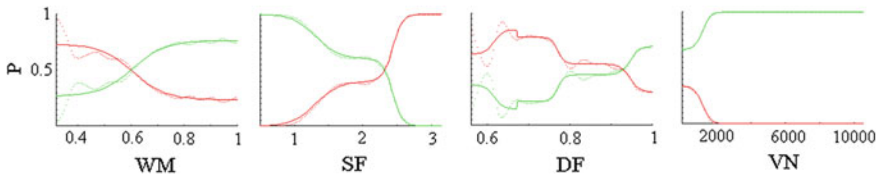


Fig. 1 Posterior probabilities of classes, $p(\text{NBT}|x)$ (*red dots*) and $p(\text{WMPL}|x)$ (*green dots*) as nonparametric functions of input features, and their approximations (*continuous lines*). (color figure online)

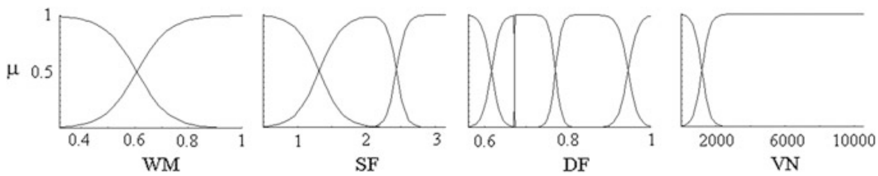


Fig. 2 Sigmoid membership functions of fuzzy sets representing partitions of input features

Table 1 Weights of different one-dimensional rule bases

Feature	$\lambda_{verylow}$-NBT	λ_{low}-NBT	λ_{medium}-NBT	λ_{high}-NBT	$\lambda_{veryhigh}$-NBT
WM		0.73		0.24	
SF		0	0.40	1	
DF	0.64	0.85	0.79	0.55	0.29
VN		0.34		0	

$\lambda_{t\text{-WMPL}} = 1 - \lambda_{t\text{-NBT}}$

Table 2 One-dimensional rule base referred to SF

Rule name	Rule						
r_{low}:	*If*	SF	*is*	Low	*then*	tissue *is*	WMPL
r_{medium}:	*If*	SF	*is*	Medium	*then*	tissue *is*	$\begin{cases} 0.40 \text{ NBT} \\ 0.60 \text{ WMPL} \end{cases}$
r_{high}:	*If*	SF	*is*	High	*then*	tissue *is*	NBT

inference by using bounded sum S-norm for aggregation of rules, and obtaining as a result good approximations of probabilities of classes, given a value of one of the features.

More rule bases can be used at the same time by combining their results as in (18). The same results can be obtained by constructing a unique model, as in (22). The rule base able to classify a data item with known features SF and VN is reported as an example in Table 3, where a rule with null weight is omitted. Weights are calculated by (20) and (21), taking into account one-dimensional weights of Table 1 and that prior probabilities of classes NBT and WMPL are, respectively, 0.33 and 0.67.

The classification power of the multidimensional model results higher, with respect to one-dimensional models. In particular, in Table 4 the classification rate (CR) obtained by the proposed method is reported, and compared to that of a naive Bayes classifier, both using all the significant features. The squared classification error (SCE) was also considered to take into account that high (low) confidence is

Table 3 Weights and rules of a multidimensional rule base

Weight	SF LT	VN LT	Conclusion
0.99	Low	low	WMPL
1.49	Low	high	WMPL
1.00	Medium	low	$\begin{cases} 0.41 \text{ NBT} \\ 0.59 \text{ WMPL} \end{cases}$
0.90	Medium	high	WMPL
1.03	High	low	NBT

Rules are of the type (LT stands for Linguistic Term): "*If* SF *is* SF LT *and* VN *is* VN LT *then* tissue *is* Conclusion"

Table 4 Classification power

Method	CR (%)	SCE
Proposed method	80.0	0.138
Naive Bayes classifier	81.0	0.190

desired to be associated to right (wrong) solutions. While CRs are similar, the SCE shows that the proposed method is better than naive Bayes classifier also in terms of results.

5 Conclusions

In this paper, a hybrid approach based on fuzzy and probabilistic views was described in order to obtain a fuzzy classifier. The good type of information extracted by the statistical approach was combined with the interpretability of a fuzzy system based on the partition of variables into linguistic terms, thus reaching the jointed objectives of a well-performing and interpretable classifier. This was achieved by approximating nonparametric functions describing posterior probabilities with linear combination of parametric membership functions, describing terms of linguistic variables. As a result, the partitions of variables and one-dimensional rule bases are constructed, thus obtaining an interpretable fuzzy system, which well approximates probabilities of different classes, given a pattern of feature values. The approach then combines results of different rule bases for improving results. On the other hand, this procedure allows to classify data with missing values of features, by combining only rule bases relative to features whose values are known. Moreover, the approximation algorithm was refined with respect to [12], and the ability of extracting knowledge from training datasets with missing data and mixed numerical/categorical features was underlined.

The reliability of the method was shown by applying it to an experimental dataset regarding Multiple Sclerosis Lesions classification.

Even if this approach offers the interpretable form of a fuzzy system, and the possibility of managing lack of knowledge and different types of features, its results differ from naive Bayes method only for making a different approximation of real probabilities, therefore the classification power of the two methods resulted very similar, if measured in terms of classification rate. On the other hand, in this approach, not only the most probable, but all classes are taken into account, together with respective probabilities. This improvement was measured by a lower squared classification error.

References

1. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Kluwer Academic Publishers Norwell, MA (1981)
2. Box, G.E.P., Tiao, G.C.: Bayesian Inference in Statistical Analysis. Wiley (1973)
3. Eick, C.F., Zeidat, N., Zhao, Z.: Supervised clustering – Algorithms and benefits. Technical Report Number UH-CS-05-10 (2005)
4. Gacto, M.J., Alcalà, R., Herrera, F.: Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures. Inf. Sci. **121,** 4340–4360 (2011)
5. Ghazavi, S.N., Liao, T.W.: Medical data mining by fuzzy modeling with selected features. Artif. Intell. Med. **43**, 195–206 (2008)
6. Glorennec, P.Y.: Algorithmes d'apprentissage pour systèmes d'inférence floue. Editions Hermès, Paris (1999)
7. Guillaume, S., Charnomordic, B.: Learning interpretable fuzzy inference systems with FisPro. Inf. Sci. **181**, 4409–4427 (2011)
8. MacQueen, J.B.: Some methods for classification and analysis of multivariate observations. In: Proceedings 5th Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, vol. 1, pp. 281–297 (1967)
9. Pedrycz, W.: Fuzzy equalization in the construction of fuzzy sets. Fuzzy Sets Syst. **64**, 21–30 (1994)
10. Pota, M., Esposito, M., De Pietro, G.: Transforming probability distributions into membership functions of fuzzy classes: a hypothesis test approach. Fuzzy Sets Syst. **233**, 52–73 (2013)
11. Pota, M., Esposito, M., De Pietro, G.: From likelihood uncertainty to fuzzyness: a possibility-based approach for building clinical DSSs. In: The 7th International Conference on Hybrid Artificial Intelligence Systems (HAIS'12), Salamanca, Spain, part II, pp. 369–380. IEEE Press, New York (2012)
12. Pota, M., Esposito, M., De Pietro, G.: Combination of interpretable fuzzy models and probabilistic inference in medical DSSs. In: 8th International Conference on Knowledge, Information and Creativity Support Systems (KICSS 2013), Krakow, Poland, pp. 541–552. Progress & Business Publishers (2013)
13. Wang, L.X., Mendel, J.M.: Generating fuzzy rules by learning from examples. IEEE Trans. Syst. Man Cybern. **22**, 1414–1427 (1992)
14. Yuan, Y., Shaw, M.J.: Induction of fuzzy decision trees. Fuzzy Sets Syst. **65**, 125–139 (1995)
15. Zadeh, L.: Fuzzy sets. Inform. Control **8**, 338–353 (1965)

Formal Specification of Temporal Constraints in Clinical Practice Guidelines

Marco Iannaccone and Massimo Esposito

Abstract In the last years, both clinical evidence and expert consensus have been codified in the form of clinical practice guidelines in order to promote an actual empowerment in the overall quality of care. Even if different solutions have been realized to specify temporal constraints in computerized guidelines, none of them proposes a formal language as the basis of guideline formalism in order to easily and directly support the temporal perspective. In such a direction, this paper proposes a formal approach, which has been seamlessly embedded into a standards-based verifiable guideline model, named GLM-CDS (GuideLine Model for Clinical Decision Support). Such an approach hybridizes the theoretic semantics of ontology and rule languages to specify a variety of temporal constraints according to some time patterns, i.e., task duration, periodicity, deadline, scheduling and time lags. These constraints are then encoded in the form of rules verifiable at run-time during the guideline enactment, in order to support the detection of violations or errors occurred with respect to the temporal perspective. As an example of application of the proposed approach, some temporal constraints have been integrated in GLM-CDS and verified by using a reasoning engine, according to the time patterns identified.

Keywords Clinical practice guidelines · Decision support systems · Time patterns · Temporal constraints · Ontology · Rules

M. Iannaccone · M. Esposito (✉)
National Research Council of Italy - Institute for High Performance Computing and Networking (ICAR), Naples, Italy
e-mail: esposito.m@na.icar.cnr.it

M. Iannaccone
e-mail: iannaccone.m@na.icar.cnr.it

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_28

1 Introduction

In the last years, both clinical evidence and expert consensus have been codified in the form of Clinical Practice Guidelines (hereafter, CPGs) in order to promote an empowerment in the overall quality of care and focus efforts to improve health outcomes when followed [14]. Recently, CPGs have been computerized into advanced Decision Support Systems (hereafter, DSSs), able to promote a better integration into the clinical workflow as well as to facilitate the automatic provision of patient-specific advice at the time and place where decisions are made [4, 15].

For this reason, many parties have been engaged in developing formalisms based on Task-Network Models (hereafter, TNMs) for encoding CPGs in the form of computer-interpretable guidelines (hereafter, CIGs) suitable for being processed by DSSs [8]. Although temporal capabilities and constraints play a fundamental role in CPGs, a majority of these CIG formalisms integrates time-related aspects only to a reduced extent [16], not enabling the possibility of seamlessly expressing a variety of temporal concepts necessary to properly encode a CPG (e.g., deadlines, minimum time lags, maximum durations, and periodicity).

This issue is a fundamental challenge to be addressed in order to model CPGs, since in most part of guidelines; tasks have to be performed respecting a set of temporal constraints concerning their relative order, duration, and delays between them. Plus, in many situations, tasks must be also repeated periodically at regular times [1]. Moreover, temporal constraints are especially important when CPGs are used in critical contexts, for instance, when physicians have to schedule different drugs regimens in specific temporal windows [3].

Recently, we proposed a standards-based process-flow-like model, named GLM-CDS (hereafter, GuideLine Model for Clinical Decision Support) [6]. It synthesizes prior work done in the guideline modeling community and directly integrates the model standardized as Health Level 7 Virtual Medical Record for Clinical Decision Support (hereafter, HL7 vMR), to explicitly define concepts and data that are used in a CIG, so that they can be seamlessly mapped to electronic health record entries [6].

In order to face the aforementioned drawbacks of the existing solutions, this paper proposes a refined and assessed version of the formal approach presented in [7]. It hybridizes the theoretic semantics of ontology and rule languages to specify a variety of temporal constraints (hereafter, TCs), within GLM-CDS, according to some time patterns, i.e., task duration, periodicity, deadline, scheduling and time lags. Such TCs are then encoded in the form of rules verifiable during the CIG enactment. A reasoning engine can be used at run-time to process and verify these TCs expressed as rules and support the detection of violations/errors with respect to the temporal perspective.

The rest of the paper is organised as follows. Section 2 outlines an overview of the state-of-the-art solutions existing in literature. In Sect. 3, the guideline model used, i.e., GLM-CDS, is described, by detailing its basic temporal aspects. Section 4 depicts the time patterns included in GLM-CDS and formulates the proposed approach to specify a variety of temporal constraints. Section 5 reports the implementation, by means of the formal approach proposed, of some temporal constraints formulated according to the time patterns identified. Finally, Sect. 6 concludes the work.

2 Related Work

In the last two decades, many formalisms, such as SAGE [17], GLIF [2], Asbru [14], EON [18] and PROforma [5], have been designed and developed as process-flow-like models for representing CPGs with a different coverage and particularities, also with reference to the temporal perspective [8].

In more detail, SAGE [17] allows describing temporal and/or iterative conditions under which an action should be repeated by means of duration and periodicity constraints. GLIF [2] deals with temporal constraints on patient data elements to be considered within the CIG to evaluate their actuality. It also allows the specification of temporal expressions to specify duration constraints on actions and decisions. Asbru [14] allows defining the duration and the starting and ending time points of a CIG activity with a variable degree of uncertainty. Moreover, its temporal scheduling constraints can be expressed with reference to multiple time lines. EON [18] enables the scheduling of guideline activities by means of temporal expressions and, in addition, the specification of duration constraints and time delays between a triggering event and the activation of specific module. In PROforma [5], CPGs are modeled as plans, which can be characterized by temporal constraints to specify task duration, delays between tasks, or preconditions of actions.

As a result, support of the time perspective in existing CIG formalisms is still rather limited with respect to the possibility of specifying a variety of time constraints, especially if compared with the wide collection of possible time patterns, which has been systematically identified in [9] by analyzing a large collection of processes in healthcare, automotive engineering, aviation industry, and other domains.

A parallel research has assumed a prominent role in the Artificial Intelligence Community in recent years, focusing on efforts to develop solutions for temporal reasoning over specific classes of constraints, i.e., qualitative constraints (e.g., "A before B"), quantitative constraints (e.g., dates, delays, and durations), or a combination of them [3, 10, 11]. However, according to such solutions, a CIG, encoded according to a predefined guideline model, has to be first translated into a formal specification language, then enriched with some temporal constraints it should hold according to specific time patterns, and, finally, delegated to a temporal network/engine for enabling automatic reasoning on it. The solution here proposed has been conceived to face this complexity in order to easily and directly support the specification of temporal constraints, as described in the following sections.

3 The GuideLine Model for Clinical Decision Support

The GLM-CDS synthesizes prior work done in the guideline modeling community and integrates the Domain Analysis Model, Release 1 of the HL7 vMR (hereafter, HL7 vMR-DAM) issued by HL7 Clinical Decision Support-Working Group, by focusing on issues pertaining the clinical decision support. It consists of a control-flow part, which is based on a formal Task-Network Model (hereafter, TNM) for codifying CPGs in terms of structured tasks connected with transition dependencies between them from an initial state of the patient. Its information model is coded in terms of guideline knowledge coherently with the HL7 vMR-DAM and populated by using existing standard terminological resources, such as Logical Observation Identifiers Names and Codes (LOINC)[1] and Systematized NOmenclature of MEDicine (SNOMED).[2] Data types used in GLM-CDS resemble the ones defined in the HL7 Vmr-DAM, which gives a simplified/ constrained version of ISO 21090 data types, based on the abstract HL7 version 3 data types specification, rel. 2.[3]

Each guideline in GLM-CDS is described in terms of four main elements: *Guideline*, *EntryPoint, ExitPoint,* and *Task*. In more detail, each (sub) guideline is represented by the TNM element named *Guideline*, whereas *EntryPoint* and *ExitPoint* represent start and end points of the TNM. The element *Task* is generic and specialized into the following sub-elements: *Action*, *Decision*, *Condition*, *Split* and *Merge*.

Action models a high-level action to be performed, which is further specialized into the sub-elements *Observation*, *Supply*, *Encounter*, *Procedure* and *SubstanceAdministration. Observation* is used to determine a measurement, a laboratory test, or a user input value. *Supply* is aimed at providing some clinical material or equipment to a patient. *Encounter* is applied to request an appointment between a patient and healthcare participants for assessing his health status. *Procedure* models an action whose outcome is the alteration of the patient's physical condition. Finally, *SubstanceAdministration* allows giving a substance to a patient for enabling a clinical effect. Each action encapsulates a list of one or more elementary and repeatable action items, namely *ObservationItem*, *SupplyItem*, *EncounterItem*, *ProcedureItem* and *SubstanceAdministrationItem,* expressed in terms of the HL7 vMR-DAM.

Decision models decision criteria for directing the control-flow from a point into the TNM to various alternatives. *Condition* is defined as an observable state of the patient that persists over time and tends to require intervention or management. It allows synchronizing the management of a patient with the corresponding guideline or parts of it. Finally, *Split* and *Merge* enable to direct the guideline flow to multiple

[1] http://loinc.org/.

[2] http://www.ihtsdo.org/snomed-ct/.

[3] http://www.hl7.org/implement/standards/rim.cfm.

parallel tasks. In particular, *Split* allows branching to multiple tasks, whereas *Merge* allows synchronizing parallel tasks by making them converging into a single point.

3.1 Temporal Concepts in GLM-CDS

Before introducing time patterns included in GLM-CDS, some general time concepts are here described, first in a general perspective and, next, with respect to the data types adopted in GLM-CDS for their instantiation.

In more detail, *instants* are defined as specific points on the time axis. The data type used for encoding an instant in GLM-CDS is *TS* (Time Stamp). It can assume a specific value represented as a calendar expression according to the format "YYYY [MM[DD[HH[MM[SS[.U[U[U[U]]]]]]]]][+ |-ZZzz]", in conformance with the constrained ISO 8601 defined in ISO 8824 (generalized time).

Intervals are derived types from instants and represent slots on the time axis bounded by two time instants. In GLM-CDS, intervals are coded by the data type *IVL_TS* (Interval Time Stamp), characterized by a low limit and a high limit. Such limits can be included or excluded within the interval, which consequently assumes a closed or open form. Moreover, low and high limits can be also valued by negative or positive infinity to indicate a left or right unbounded interval, respectively.

Durations indicates the time length of a bounded interval. In GLM-CDS, it is represented by the data type *PQ* (Physical Quantity), which is a dimensioned quantity characterized by both the unit of measure, according to the Unified Code for Units of Measure, and its numerical value (e.g., unit of measure = second and value = 3). Moreover, also the data type *IVL_PQ* (Interval Physical Quantity) can be used to represent a bounded interval instead of a single numerical value.

Events are occurrences of facts happening during a guideline execution at particular time instants. In GLM-CDS, *events* are specifically correlated to the different tasks foreseen (i.e., *Supply*, *Encounter*, *Procedure* and so on), and are characterized by a name, for instance activation or completion of a task, and a time of occurrence.

Finally, the *lifecycle* of a task within a CIG is defined as a progression through a set of states, from its creation until fail or completion, guided by the occurrence of specific events. The lifecycle in GLM-CDS is shown in Fig. 1, where the possible states a task can assume are *new, active, canceled, aborted, suspended* and *completed*. They resemble the states of an *Act* as defined in the HL7 v3 RIM.

In detail, every task is initially in a *new* state. It can be abandoned before its *activation* by changing its state from *new* to *canceled*. When it is being performed, its state changes from *new* to *active*. Whenever a task is finished normally after all of its constituents have been performed, its state becomes *completed*. Its termination can be forced prior to the originally intended completion, by changing its state into *aborted*. Finally, an active task can be temporarily disabled, by setting its state as *suspended*.

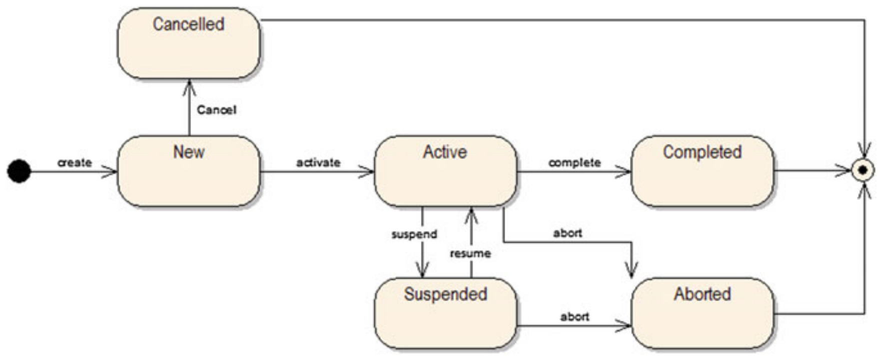


Fig. 1 The lifecycle of a task

The list of possible events defined in GLM-CDS for enabling such progressions of states is detailed in Fig. 1, as well. These events are *create, cancel, activate, complete, abort, suspend, and resume.*

4 The Formal Approach for a Time-Aware Guideline Model

4.1 Time Patterns Included in GLM-CDS

The range of time patterns included in GLM-CDS resembles the ones that have been identified in the reviewed literature [9].They have been here described, based on the different states that a task can assume during its lifecycle.

The first pattern is named *task duration* and models the time length in which a task needs to be completed. It can be expressed by indicating a minimum value, a maximum value, or both. This pattern refers to the time interval passing since a task is created and until its completion, including the intermediary states through which the task passes during its lifecycle as active or suspended, for instance.

The second time pattern is named *deadline*, and indicates that a certain event needs to be executed at a specified time point. It is expressed by indicating a specific instant on the time axis and used to constrain two events concerning the lifecycle of a task, i.e., *activate* and *complete*, which model the activation and completion of a task, respectively. In other words, this pattern enables to specify that the *active* and *completed* states of a task are strongly dependent on certain fixed instants for being assumed.

Another pattern is called *periodicity* and indicates that a task must be executed according to a certain repetitive rule. Such a way, CIGs which contain tasks that need to be executed periodically and not necessarily at an equal distance in time can be properly handled. This pattern can be expressed by specifying the time interval within which a task should be executed and the number of executions to do, by assuming that these executions will be evenly distributed within the timeframe (e.g., if proposed time interval is between 1/1/2013 to 12/31/2013, and the number of executions to do is three, ideal execution times would be 1/1/2013, 12/31/2013, and in the middle of the year).

Moreover, the pattern named *scheduling* indicates that a certain task might be executed only in a bounded time interval fixed by a particular schedule.

Finally, the last pattern defined is *time lag*, which is devised to specify that a specific time distance between two sequential or parallel tasks needs to be respected, where the order of execution is determined by the CIG control flow. It can be expressed by indicating an ordering relation between the states *active* and *completed* of the two corresponding tasks involved and, in addition, a duration, which can be defined by means of a minimum value, a maximum value, or both.

The ordering relations are *before* and *cross-overlapping* in case of two sequential tasks, whereas they are *left-overlapping*, *middle-overlapping*, and *right-overlapping* in case of two parallel tasks. In Fig. 2, an exemplification of these ordering relations for two tasks A and B has been detailed, where the events considered are *activate* A (act-A), *complete* A (comp-A), *activate* B (act-B), and *complete* B (comp-B).

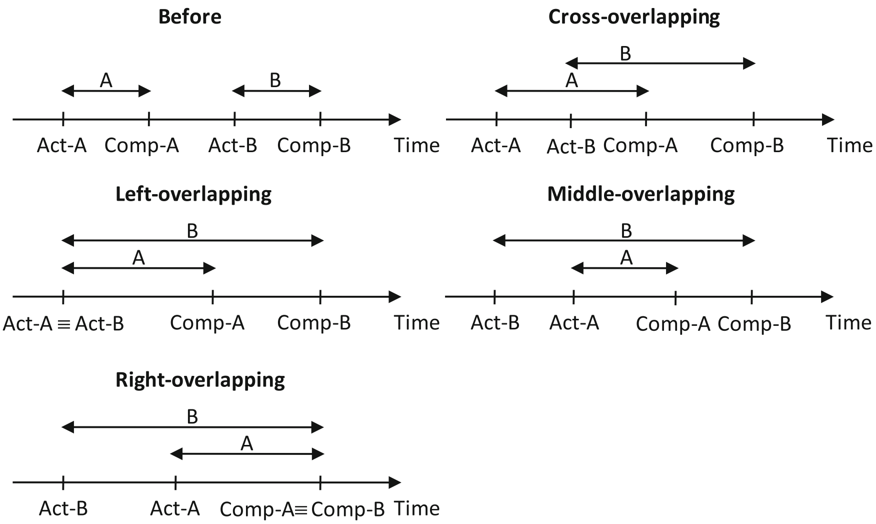


Fig. 2 Ordering relations between two sequential or parallel tasks

4.2 The Hybridization of Ontologies and Rules for Encoding Temporal Constraints

The computerization of temporal aspects in CPGs according to the time patterns above-described has been realized via a formal approach which combines ontologies and rules to implement GLM-CDS as a hybrid knowledge base and, thus, to specify temporal constraints as special rules to be checked against such a knowledge base.

Ontology languages are based on the notion of concepts (unary predicates and classes), individuals (instances of concepts), abstract roles (binary predicates between concepts) and concrete roles (binary predicates between concepts and data values). On the other hand, rule languages, typically used as an extension to ontology languages, are based on the notion of constants (numbers, symbolic non-numeric constants, and string constants), variables (symbols which can assume different values), and rule predicates (unary or binary, involving constants and/or variables).

The approach here proposed can be formally described as follows. Let us consider a generic ontology language $\mathbf{O}$ made of non-empty and disjoint sets of predicates $\mathbf{O}_p$, containing concepts and roles, and individual constants $\mathbf{O}_i$. Let us consider also a rule language $\mathbf{R}$ made of non-empty and pair-wise disjoint sets of variables $\mathbf{R}_v$, individual constants $\mathbf{R}_i$, and rule predicates $\mathbf{R}_p$.

By exploiting such alphabets of $\mathbf{R}$, an *atom* is defined as an expression in the form $a(T)$, where $a \in \mathbf{R}_p$ and T is a tuple of variables $v \in \mathbf{R}_v$ or individual constants $i \in \mathbf{R}_i$.

With these premises, the *hybrid Knowledge Base K* is defined as:

- a finite set of deductive rules $\{R\}$ over $\mathbf{O}$ and $\mathbf{R}$, assuming the form:

$$r_1(X_1) \leftarrow a_1(Y_1) \wedge \ldots \wedge a_k(Y_k) \tag{1}$$

  where:

  - $r_1(X_1)$ is named *head,* $p_1(Y_1)\ldots p_k(Y_k)$ (with $k \geq 0$) are called *body,*
  - $r_1, a_1, \ldots, a_k \in \mathbf{R}_p$ and $X_1, Y_1, \ldots, Y_k \in \mathbf{R}_v$ or $\in \mathbf{R}_i$;
  - the conditions that $\mathbf{R}_p \supseteq \mathbf{O}_p, \mathbf{O}_i \equiv \mathbf{R}_i$ are satisfied;
  - each variable of its head appears also in the body;

- a finite set of *temporal constraints* $\{TC\}$ over $\mathbf{O}$ and $\mathbf{R}$, assuming the form:

$$TC_i \leftarrow a_1(Y_1) \wedge \ldots \wedge a_k(Y_k) \tag{2}$$

where:

- $TC_i \in \mathbf{R}_p$ represents a special predicate associated to each TC;
- $a_1, \ldots, a_k \in \mathbf{R}_p$ and $X_1, Y_1, \ldots, Y_k \in \mathbf{R}_v$ or $\in \mathbf{R}_i$;
- the conditions that $\mathbf{R}_p \supseteq \mathbf{O}_p$ and $\mathbf{O}_i \equiv \mathbf{R}_i$ are satisfied.

As a consequence of that, since TCs are coded in the same form of deductive rules on the top of ontology concepts, roles, and individuals, a reasoning engine using forward chaining can be employed to process them and entail their special predicates with respect to an existing knowledge base. In particular, this kind of engine can be used to process TCs defined over a knowledge base until one is found with its body atoms known to be true. When this condition is satisfied, the engine can conclude, or infer, the special predicate associated to that TC, enabling the addition of its atom to the knowledge base. This implies, therefore, that a TC can be considered as violated when its special predicate is inferred by the reasoning engine over a knowledge base.

5 Implementation of Temporal Constraints

The proposed approach has been applied to computerize GLM-CDS and produce a knowledge base, expressed in the Web Ontology Language (hereafter, OWL), and a set of TCs on the top of it. In detail, all the basic elements forming a TNM in GLM-CDS have been computerized as ontology concepts, whereas all their specializations as subsumed ones. All the elements composing the information model of GLM-CDS, such as action items or decision models, have been encoded as ontology concepts, as well. Moreover, a number of abstract and concrete roles have been defined for expressing relationships between concepts or specifying data values a concept can assume. In the following, concepts are indicated with the first letter capitalized, whereas abstract and concrete roles are entirely written in lower case. For instance, *status* is the concrete role, which associates a *Task* with the possible states it can assume. On the other hand, examples of abstract roles are: (i) *startingTime/completionTime,* for indicating the time instants when a specific task is activated/completed; (ii) *hasTaskDuration/hasTimeLag,* for associating a *task duration* or *time lag* pattern to a task; (iii) *hasMinDuration/hasMaxDuration/hasIntervalDuration*, for expressing a duration with a minimum value, maximum value, or an interval. For major details about all the concepts and roles defined in GLM-CDS, please refer to [6].

On the top of this knowledge base, TCs have been expressed according to the time patterns described above in the form of rules with a special predicate in their heads as indicated in Tables 1 and 2. The abstract syntax used to encode rules is formalized by means of the Extended BNF, as reported in Fig. 3.

For the sake of brevity, only TCs formulated according to two of the patterns above-mentioned, i.e., *task duration* and *time lags,* have been here reported in form

Table 1 TCs formulated as rules according to the *task duration* pattern

ID	rules
1	*startingTime*(x,ts) ← *Task*(x) ∧ *status*(x,y) ∧ *equal*(y,'new') ∧ *Event*(e) ∧ *isLinkedTo*(e,x) ∧ *eventName*(e, n) ∧ *equal*(n,'activate') ∧ *eventTime*(e, ts)
2	*completionTime*(x,ts) ← *Task*(x) ∧ *status*(x,y) ∧ *equal*(y,'active') ∧ *Event*(e) ∧ *isLinkedTo*(e,x) ∧ *eventName*(e, n) ∧ *equal*(n,'complete') ∧ *eventTime*(e, ts)
3	TC_{D1} ← *Task*(x) ∧ *hasTaskDuration*(x,td) ∧ *TaskDuration*(td) ∧ *hasMinDuration*(td,pq) ∧ *PQ*(pq) ∧ *value*(pq,vd) ∧ *startingTime*(x,t1) ∧ *TS*(t1) ∧ *value*(t1,v1) ∧ *completionTime*(x,t2) ∧ *TS*(t2) ∧ *value*(t2,v2) ∧ *difference*(v2,v1,vf) ∧ *lessThan*(vf,vd)
4	TC_{D2} ← *Task*(x) ∧ *hasTaskDuration*(x,td) ∧ *TaskDuration*(td) ∧ *hasMaxDuration*(td,pq) ∧ *PQ*(pq) ∧ *value*(pq,vd) ∧ *startingTime*(x,t1) ∧ *TS*(t1) ∧ *value*(t1,v1) ∧ *completionTime*(x,t2) ∧ *TS*(t2) ∧ *value*(t2,v2) ∧ *difference*(v2,v1,vf) ∧ *greaterThan*(vf,vd)
5	TC_{D3} ← *Task*(x) ∧ *hasTaskDuration*(x,td) ∧ *TaskDuration*(td) ∧ *hasIntervalDuration*(td,pq) ∧ *IVL_PQ*(pq) ∧ *low*(pq,pql) ∧ *value*(pql,vdl) ∧ *startingTime*(x,t1) ∧ *TS*(t1) ∧ *value*(t1,v1) ∧ *completionTime*(x,t2) ∧ *TS*(t2) ∧ *value*(t2,v2) ∧ *difference*(v2,v1,vf) ∧ *lessThan*(vf, vdl)
6	TC_{D3} ← *Task*(x)∧ *hasTaskDuration*(x,td) ∧ *TaskDuration*(td) ∧ *hasIntervalDuration*(td,pq) ∧ *IVL_PQ*(pq) ∧ *high*(pq,pqh) ∧ *value*(pqh,vdh) ∧ *startingTime*(x,t1) ∧ *TS*(t1) ∧ *value*(t1,v1) ∧ *completionTime*(x,t2) ∧ *TS*(t2) ∧ *value*(t2,v2) ∧ *difference*(v2,v1,vf) ∧ *greaterThan*(vf, vdh)

Table 2 TCs in the form of rules according to the *time lag* pattern, when *before* is used

ID	rules
1	TC_{L1} ← *Task*(x) ∧ *hasTimeLag*(x,tl) ∧ *TimeLag*(tl) ∧ *before*(tl, y) ∧ *Task*(y) ∧ *hasMinDuration*(td,pq) ∧ *PQ*(pq) ∧ *value*(pq,vd) ∧ *startingTime*(y,t1) ∧ *TS*(t1) ∧ *value*(t1,v1) ∧ *completionTime*(x,t2) ∧ *TS*(t2) ∧ *value*(t2,v2) ∧ *difference*(v1,v2,vf) ∧ *lessThan*(vf,vd)
2	TC_{L2} ← *Task*(x) ∧ *hasTimeLag*(x,tl) ∧ *TimeLag*(tl) ∧ *before*(tl, y) ∧ *Task*(y) ∧ *hasMaxDuration*(td,pq) ∧ *PQ*(pq) ∧ *value*(pq,vd) ∧ *startingTime*(y,t1) ∧ *TS*(t1) ∧ *value*(t1,v1) ∧ *completionTime*(x,t2) ∧ *TS*(t2) ∧ *value*(t2,v2)∧ *difference*(v1,v2,vf) ∧ *greaterThan*(vf,vd)
3	TC_{L3} ← *Task*(x) ∧ *hasTimeLag*(x,tl) ∧ *TimeLag*(tl) ∧ *before*(tl, y) ∧ *Task*(y) ∧ *hasIntervalDuration*(td,pq) ∧ *IVL_PQ*(pq) ∧ *low*(pq,pql) ∧ *value*(pql,vdl) ∧ *startingTime*(y,t1) ∧ *TS*(t1) ∧ *value*(t1,v1) ∧ *completionTime*(x,t2) ∧ *TS*(t2) ∧ *value*(t2,v2)∧ *difference*(v1,v2,vf) ∧ *lessThan*(vf, vdl)
4	TC_{L3} ← *Task*(x) ∧ *hasTimeLag*(x,tl) ∧ *TimeLag*(tl) ∧ *before*(tl, y) ∧ *Task*(y) ∧ *hasIntervalDuration*(td,pq) ∧ *IVL_PQ*(pq) ∧ *high*(pq,pqh) ∧ *value*(pqh,vdh) ∧ *startingTime*(y,t1) ∧ *TS*(t1) ∧ *value*(t1,v1) ∧ *completionTime*(x,t2) ∧ *TS*(t2) ∧ *value*(t2,v2) ∧ *difference*(v1,v2,vf) ∧ *greaterThan*(vf,vdh)

of rules. The notation adopted specifies concepts and roles as unary and binary predicates, where *Concept*(x) is used to test if a variable *x* is an instance of *Concept*, whereas *role*(*x,y*) to test if a variable *x* is linked to a variable *y* by means of *role.*

In particular, with reference to the *task duration* pattern, the first two rules reported in Table 1 are deductive and conceived to define more complex axioms for assessing the time interval passing since a specific task is activated and until its completion. The other rules instantiate, instead, temporal constraints and, in detail,

```
Rule::= Weight "IF (" Antecedent ") THEN (" Consequent ")" ;
Weight ::= Value;
Antecedent::= {Statement};
Consequent::= {Statement};
Statement::= UnaryPredicate | BinaryPredicate | BuiltinStatement;
UnaryPredicate::= Negated UPredicate "(" Term ")";
UPredicate::= OntologyConcept;
BinaryPredicate::= Negated Bpredicate "(" Term "," Term ")";
BPredicate::= OntologyRole;
Term::= OntologyIndividual | Variable;
Negated::= not;
Variable::= "?"a | "?"b | ...... "?"z;
Value::= Real Number;
BuiltinStatement ::= LogicalBuiltinStatement | Arithmetical BuiltinStatement | TemporalBuiltinStatement | PresenceBuiltinStatement;
LogicalBuiltinStatement::= LogicalOperator "(" Variable"," Variable|Value")";
LogicalOperator::= equal  | notEqual | lessEqual | lessThan | greaterThan | greaterEqual;
ArithmeticalBuiltinStatement::= ArithmeticalOperator "(" Variable"," Variable|Value")";
ArithmeticalOperator ::= sum | difference | product | quotient;
TemporalBuiltinStatement::= TemporalOperator "(" Variable ")";
TemporalOperator::= now;
PresencelBuiltinStatement::= PresenceOperator "(" Variable "," List  ")";
PresenceOperator ::= isIn| isOut;
List::= {OntologyConcept | OntologyRole| OntologyIndividual };
```

Fig. 3 Abstract syntax for rule model expressed in Extended BNF

compare duration calculated and fixed for a task by entailing their own special predicate TC_i when the duration calculated is: (i) higher than the imposed one (see rule 3 in Table 1), in case this latter is set as a minimum value, (ii) lower than the imposed one (see rule 4 in Table 1), in case the latter is set as a maximum value, (iii) outside the imposed boundaries in case of an interval constraint specification (see rules 5 and 6 in Table 1).

On other hand, with reference to the *time lag* pattern, the first two rules in Table 1 are used also in this case to determine the time instants when a specific task is activated or completed. Starting from these time instants calculated for each task, different rules involving the various ordering relations defined above have been formulated in order to verify whether a fixed time lag between two tasks is respected or not. In particular, the rules reported in Table 2 describes, as an example, how it is possible to verify this pattern in case of two sequential tasks where the ordering relation *before* is used and the time lag is specified as minimum value (see rule 1 in Table 2), maximum value (see rule 2 in Table 2) ,or an interval (see rules 3 and 4 in Table 2).

In order to highlight how such TCs operate on instances of CIGs encoded according to GLM-CDS, a fragment of the guideline for "Advanced Breast Cancer" of the National Institute for Health and Care Excellence has been computerized as shown in Fig. 4. Starting from this CIG, two different sets of data instances have been considered in Fig. 5, pertaining a *Condition* and two consecutive *Observations*, respectively (Fig. 5a, b).

The fragment of CIG for "Advanced Breast Cancer" together with such data instances have been supplied to the reasoning engine presented in [12] as OWL statements and used to evaluate defined TCs and draw all possible violations. Such a reasoning engine has been used since it is able to adopt the forward chaining

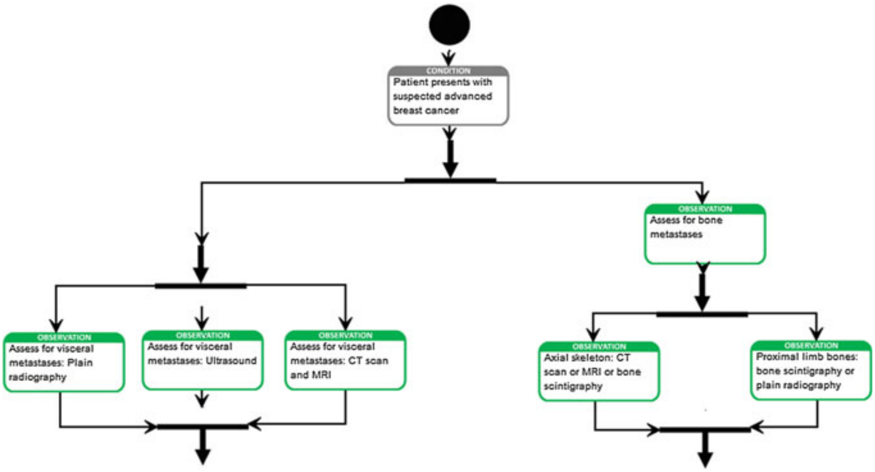


Fig. 4 The NICE guideline for advanced breast cancer encoded in GLM-CDS

(a)

Condition: *Patient with suspected advanced breast cancer*
  hasTaskDuration:
    hasIntervalDuration:
      high:
        unit: *min*
        value: *30*
  startingTime:
    value: *201309091200*
  completionTime:
    value: *201309091240*

(b)

Observation: *Assess for bone metastasis*
  hasTimeLag:
    before:
      Observation: *Axial skeleton*
    hasMinDuration:
      unit: *min*
      value: *30*
    completionTime :
      value: *201309091200*

Observation: *Axial skeleton*
  startingTime:
    value: *201309091220*

Fig. 5 Two sets of data instances producing the violation of TCs with respect to task duration and time lag patterns

scheme and process TCs expressed as rules according to the abstract syntax reported in Fig. 3.

The results produced by the engine after processing the aforesaid TCs with respect to the set of data instances reported in Fig. 5a, b indicate that TC_{D3} and TC_{L1} are violated, respectively, with reference to the task duration and time lag patterns.

6 Conclusions

To date, the attempts involved to specify and verify temporal constraints in CIGs have been not widely applied in practice, since none of these solutions is concerned with directly embedding the theoretic semantics of a formal language as basis of guideline formalism in order to easily and directly support the temporal perspective.

As a consequence of that, this paper proposed a formal approach, seamlessly embedded into the guideline model, named GLM-CDS, previously conceived by the authors and described in [6]. This approach integrates the theoretic semantics of ontology and rule languages to specify a variety of temporal constraints. In detail, such TCs are formulated according to some time patterns, i.e., task duration, periodicity, deadline, scheduling and time lags, and encoded as rules verifiable at run-time during the CIG enactment. A reasoning engine can be used at run-time to automatically process and verify these TCs in the form of rules and detect violations/errors with respect to the temporal perspective.

Preliminary tests performed on existing CPGs gave a proof of the feasibility of the approach, suggesting that it can be proficiently applied in real scenarios.

As a concluding remark, it is relevant to observe that, as far as this current work is concerned, no experimental validation has been considered, leaving the addressing of this yet critical matter for future work, when some mechanisms for verifying coverage, correctness, tractability, and applicability will be studied and possibly adopted.

References

1. Anselma, L., Terenziani, P., Montani, S., Bottrighi, A.: Towards a comprehensive treatment of repetitions, periodicity and temporal constraints in clinical guidelines. Artif. Intell. Med. **38**(2), 171–195 (2006)
2. Boxwala, A.A., Peleg, M., Tu, S., Ogunyemi, O., Zeng, Q.T., Wang, D., Patel, V.L., Greenes, R.A., Shortliffe, E.H.: GLIF3: a representation format for sharable computer-interpretable clinical practice guidelines. J. Biomed. Inform. **37**(3), 147–161 (2004)
3. Duftschmid, G., Miksch, S., Gall, W.: Verification of temporal scheduling constraints in clinical practice guidelines. Artif. Intell. Med. **25**(2), 93–121 (2002)
4. Field, M.J., Lohr, K.N.: Guidelines for Clinical Practice: From Development to Use. Institute of Medicine, National Academy Press, Washington, DC (1992)
5. Fox, J., Johns, N., Rahmanzadeh, A.: Disseminating medical knowledge: the PROforma approach. Artif. Intell. Med. **14**, 157–181 (1998)
6. Iannaccone, M., Esposito, M., De Pietro, G.: A standards-based verifiable guideline model for decision support in clinical applications. In: Process Support and Knowledge Representation in Health Care 8268, pp.143–157 (2013)
7. Iannaccone, M., Esposito, M.: Formal encoding and verification of temporal constraints in clinical practice guidelines. In: The 8th International Conference on Knowledge, Information and Creativity Support Systems (KICSS 2013), pp. 211–222. Progress & Business Publishers, Krakow (2013)
8. Isern, D., Moreno, A.: Computer-based execution of clinical guidelines: a review. Int. J. Med. Inform. **77**, 787–808 (2008)

9. Kautz H.A., Ladkin P.B.: Integrating metric and qualitative temporal reasoning. In: Proceedings 9th NationalConference on Artificial Intelligence (AAAI'91). AT&T Bell Laboratories, Murray Hill, NJ, USA. AAAI Press, MenloPark (CA, USA) (1991)
10. Lanz, A., Weber, B., Reichert, M.: Time patterns for process-aware information systems. Requir. Eng. 1–29 (2012)
11. Meiri, I.: Combining qualitative and quantitative constraints in temporal reasoning. Artif. Intell. **87**, 343–385 (1996)
12. Minutolo, A., Esposito, M., De Pietro, G.: A Mobile reasoning system for supporting the monitoring of chronic diseases. In: Wireless Mobile Communication and Healthcare, pp. 225–232 (2012)
13. Scott, I.: What are the most effective strategies for improving quality and safety of health care? Int. Med. J. **39**(6), 389–400 (2009)
14. Seyfang, A., Miksch, S., Marcos, M.: Combining diagnosis and treatment using Asbru. Int. J. Med. Inf. **68**(1–3), 49–57 (2002)
15. Shortliffe, E., Cimino, J.: Biomedical Informatics: Computer Applications in Health Care and Biomedicine (Health Informatics), Springer, New York, Inc., Secaucus, NJ (2006)
16. Terenziani P.: Reasoning about time. In: Encyclopedia of Cognitive Science, pp. 869–874. Macmillan Reference Ltd. (2003)
17. Tu, S.W., Campbell, J.R., Glasgow, J., Nyman, M.A., McClure, R., McClay, J., Parker, C., Hrabak, K.M., Berg, D., Weida, T., Mansfield, J.G., Musen, M.A., Abarbanel, R.M.: The SAGE guideline model: achievements and overview. J. Am. Med. Inform. Assoc. **14**, 589–598 (2007)
18. Tu, S.W., Musen, M.A.: Modeling data and knowledge in the EON guideline architecture. Stud. Health. Technol. Inform. **84**(Pt. 1), 280–284 (2001)

Part V
Feature Engineering and Classification

A Comparative Study on Single and Dual Space Reduction in Multi-label Classification

Eakasit Pacharawongsakda and Thanaruk Theeramunkong

Abstract Multi-label classification has been applied to several applications since it can assign multiple class labels to an object. However, its effectiveness might be sacrificed due to high-dimensionality problem in both feature space and label space. To address these issues, several dimensionality reduction methods have been proposed to transform the high-dimensional spaces to low-dimensional spaces. This paper aims to provide a comprehensive review on ten-dimensionality reduction methods that applied to multi-label classification. These methods can be categorized into two main approaches: single space reduction and dual space reduction. While the former approach aims to reduce the complexity in either feature space or label space, the latter approach transforms both feature and label spaces into two subspaces. Moreover, a comparative study on single space reduction and dual space reduction approaches with five real-world datasets are also reported. The experimental results indicated that dual space reduction approach tends to give better performance comparing to the single reduction approach. Furthermore, experiments have been conducted to investigate the effect of dataset characteristics on classification performance.

Keywords Multi-label classification · Dimensionality reduction · Dual space reduction

1 Introduction

In the past, traditional classification techniques assumed a single category for each object to be classified by means of minimum distance. However, in some tasks it is natural to assign more than one category to an object. For examples, some news articles can be categorized into both *politic* and *crime*, or some movies can be labeled

E. Pacharawongsakda (✉) · T. Theeramunkong
School of Information, Computer, and Communication Technology,
Sirindhorn International Institute of Technology, Thammasat University,
Bangkok, Thailand
e-mail: eakasit@siit.tu.ac.th; sit.ake@gmail.com

T. Theeramunkong
e-mail: thanaruk@siit.tu.ac.th

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_29

as *action* and *comedy*, simultaneously. As a special type of task, multi-label classification was initially studied by Schapire and Singer [10] in text categorization. Later many techniques in multi-label classification have been proposed for various applications such as image classification, music to emotion categorization, automated tag recommendation, Bioinformatics research, and sentiment analysis. However, these methods can be grouped into two main approaches: *Algorithm Adaptation* (AA) and *Problem Transformation* (PT) as suggested in [12]. The former approach modifies existing classification methods to handle multi-label data, e.g., Adaboost. MH [10]. On the other hand, the latter approach transforms a multi-label classification task into several single classification tasks and then applies traditional classification method on each task. This approach includes binary relevance (BR) and label powerset (LP). Later, Madjarov et al. [5] poses another approach which is an ensemble learning. This approach combines several classification models to improve predictive performance. Ensemble of classifier chain (ECC) [9] is an example of this approach.

Residing in these three main approaches, one major issue is curse of dimensionality, which means datasets have high-dimensional feature space. This situation might triggers a well-known overfitting problem since there are not enough objects for presenting all feature patterns. Therefore, classification model remembers only patterns appeared in training data but it cannot generalize to predict the unseen data. Hence, its classification performance is quite low. To solve these issues, many techniques have been proposed, e.g., sparse regularization, feature selection, and dimensionality reduction. Among these methods, the dimensionality reduction which transforms data in a high-dimensional space to those in the low-dimensional space, has been focused on multi-label classification problem. In general, these dimensionality reduction methods can be categorized into two main groups: single space reduction and dual space reduction. While methods in the former group aim to reduce the complexity in either feature space or label space, the latter group transforms both feature and label spaces into two subspaces.

Since there are many multi-label classification methods, two recently comprehensive reviews have been published by Madjarov et al. [5] and Zhang and Zhou [16]. However, these previous reviews did not examine the dimensionality reduction methods that exploited in multi-label classification. Therefore, this paper aims to provide comprehensive review on both single and dual space reduction approaches and presents performance comparison of these techniques when they are applied to the real-world datasets. In the rest of this paper, Sect. 2 gives notations for the multi-label classification task. Section 3 presents comprehensive review on single and dual space reduction methods. The multi-label benchmark datasets and experimental settings are described in Sect. 4. In Sect. 5, the performance comparisons with ten-dimensionality reduction on five datasets are given and finally Sect. 7 provides conclusion of this work.

2 Definition of Multi-label Classification Task

Let $\mathcal{X} = \mathbb{R}^M$ and $\mathcal{Y} = \{0, 1\}^Q$ be an M-dimensional feature space and Q-dimensional binary label space, where M is the number of features and Q is a number of possible labels, i.e., classes. Let $\mathcal{D} = \{(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2), \ldots, (\mathbf{x}_N, \mathbf{y}_N)\}$ is a set of N objects (e.g., documents, images, etc.) in a training dataset, where $\mathbf{x}_i \in \mathcal{X}$ is a feature vector that represents an ith object and $\mathbf{y}_i \in \mathcal{Y}$ is a label vector with the length of L, $[y_{i1}, y_{i2}, \ldots, y_{iQ}]$. Here, y_{ij} indicates whether the ith object belongs (1) or not (0) to the jth class (the jth label or not, i.e., whether it is given). For convenience, $\mathbf{X}_{N\times M} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^T$ denotes the *feature matrix* with N rows and M columns and $\mathbf{Y}_{N\times L} = [\mathbf{y}_1, \ldots, \mathbf{y}_N]^T$ represents the *label matrix* with N rows and Q columns , where $[\cdot]^T$denotes matrix transpose.

In general, two main phases are exploited in a multi-label classification problem: (1) *model training* phase and (2) *classification* phase. The goal of the *model training* phase is to build a classification model that can predict the label vector $\mathbf{y}_t$ for a new object with the feature vector $\mathbf{x}_t$. This classification model is a mapping function $\mathcal{H} : \mathbb{R}^M \rightarrow \{0, 1\}^L$ which can predict a target value closest to its actual value in total. The *classification* phase uses this classification model to assign labels. When dimensionality reduction technique is exploited to multi-label classification, it is possible to add *pre-processing* phase before the *model training* phase. Likewise, *post-processing* phase might appear after the *classification* phase. The *pre-processing* phase aims to transform high-dimensional feature space (or high-dimensional label space) to feature subspace (or label subspace). On the other hand, the purpose of *post-processing* phase is to reconstruct the label subspace to high-dimensionality label space and it is found in the label space reduction approach.

3 Dimensionality Reduction in Multi-label Classification

This section provides comprehensive review on dimensionality reduction methods that exploited in multi-label classification. As mentioned earlier, these methods can be categorized into two main approaches: single space reduction and dual space reduction. A list of dimensionality reduction methods that are reviewed in this paper are shown in Fig. 1 and next subsections describe details of these approaches.

3.1 Review of Single Space Reduction

It is possible to divide single space reduction approach into feature space reduction method and label space reduction method. The feature space reduction method was formerly studied by Yu et al. [15]. In their work, multi-label latent semantic indexing (MLSI) was proposed to project the high-dimensional feature space into a

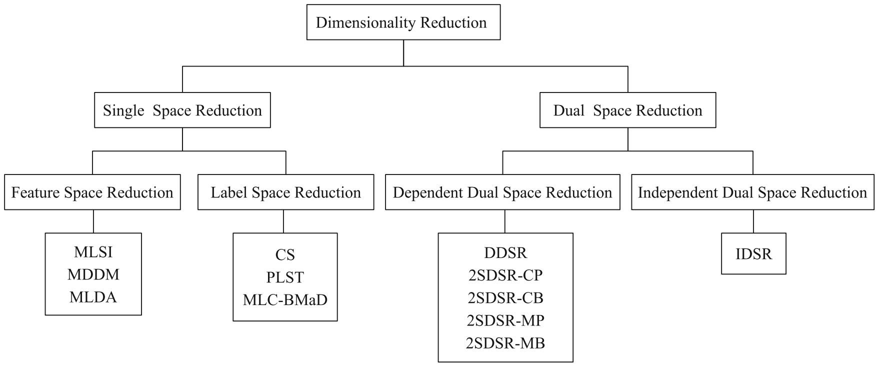


Fig. 1 The dimensionality reduction methods that reviewed in this paper. It divided into groups as discussed in the Sects. 3.1 and 3.2.

low-dimensional feature space (or feature subspace). This feature subspace is constructed by a supervised LSI since the method considers a minimum error in both feature and label spaces. By solving an optimization problem with SVM as the base classifier, the results showed that classification accuracy could be improved over the method using the original feature space and the (unsupervised) LSI. In MLSI, although labels were concerned, the method reduced the feature space but still used the high-dimensionality label space. Motivated by MLSI, multi-label dimensionality reduction via dependence maximization (MDDM) was proposed by Zhang and Zhou in [17]. In their work, Hilbert-Schmidt Independence Criterion (HSIC) was applied rather than LSI and its aim was to identify reduced feature space that maximizes dependency between the original feature space and the label space. Using MDDM with multi-label K-nearest neighbors (ML-KNN) as a base classifier, this method obtained higher accuracy than MLSI. Recently, Wang et al. [13] has proposed a method to extend linear discriminant analysis (LDA), a well-known dimensionality reduction method, to handle multi-label data.

On the other hand, for the label space reduction, to improve efficiency of multi-label classification, Hsu et al. [4] posed a sparseness problem that mostly occurred in the label space and then applied compressive sensing (CS) technique, widely used in the image processing field, to encode and decode the label space. In encoding step, the CS technique was used to create a small number of linear random projections while decoding step needs to solve an optimization problem to reconstruct the labels with respect to the sparsity error. It seems that the encoding step of this CS method seems efficient but the decoding step does not. Toward this issue, Tai and Lin [11] proposed principle label space transformation (PLST) to transform the label space into a smaller linear label space using singular value decomposition (SVD) [3]. With the properties of orthorgonal matrix derived from SVD, this matrix can be used to transform the reduced label space back to the original space. By a number of experiments, PLST was shown to improve classification performance in both accuracy and computational time, compared to BR and CS approaches. Recently, Wicker et al. [14] presented an alternative method, namely MLC-BMaD, to reduce label space using Boolean Matrix Decomposition (BMD)[6]. While SVD transforms high-dimensional binary labels to low-dimensional numeric values, BMD still preserves the label information in boolean format. With this property, this work shown higher performance, compared to the PLST and BR.

3.2 Review of Dual Space Reduction

As mentioned before, several approaches were presented to handle either the high-dimensionality problem in feature space or the sparseness problem in label space. These two aforementioned problems motivated us to study predictive performance when both spaces are simultaneously reduced. For this point, it is worth investigating performance in several situations, such as dependent dimensionality reduction of feature and label spaces, and independent dimensionality reduction of feature and label spaces. Thus, we proposed various approaches that examined the effect of dependent and independent approaches. For instances, the dependent dual space reduction (DDSR) and two-stage dependent dual space reduction (2SDSR) are classified to the dependent approach and the independent dual space reduction (IDSR) is classified to the independent approach. The concepts of these dual space reduction approaches are described as following.

The DDSR approach [7] aims to eliminate redundancy as well as noise in both feature and label spaces by using singular value decomposition (SVD) [3]. By this situation, it should gain better predictive performance and lower computational cost. This approach computes a cross-covariance matrix to construct dependency profile between feature and label spaces before applying SVD to grasp dependency among features and labels and then reduce noise. Both features and labels are mapped to the same low-dimensional space, revealed by SVD. Finally, feature subspace is used to predict each labels in the label subspace. To reconstruct the labels, the component of SVD is used to project labels in the label subspace back to those in the high-dimensional label space and an adaptive threshold is used to map a numeric value to a binary value for label determination. However, the DDSR has one limitation which is the dimensions of both feature and label spaces are required to reduce to the same number and it must be smaller than the minimum number of the number of features and that of labels. To overcome this issue, the 2SDSR [8] presented two-stage dimensionality reduction framework. In the first stage, the labels were transformed to label subspace. For the second stage, the small numbers of features which maximize dependency between features and that label subspace were uncovered. By this situation, it is possible to employ various feature space reduction methods such as CCA and MDDM, as previously described, to the framework. Likewise the PLST and MLC-BMaD, as reviewed in the previous sections, can be exploited to label space reduction in the 2SDSR framework. Therefore, that work composed of four variations of 2SDSR which are 2SDSR-CP, 2SDSR-CB, 2SDSR-MP, and

2SDSR-MB. Label reconstruction in the *post-processing* phase is quite similar to the DDSR approach that previously described.

As opposed to the dependent approach, IDSR method [7] was developed to reduce feature and label spaces by considering them separately. However, to exploit dependency among features and dependency among labels, this approach calculates the covariance matrix of the object-feature matrix and the covariance matrix of the object-label matrix. Semantically, one expresses dependency between features and the other represents dependency between labels. While IDSR employed the covariance metric, it is also possible to use other dependency metrics, for example, entropy, chi-square or symmetric uncertainty (SU), to represent the dependency among features and that among labels. After that two independent SVDs are applied to these matrices for the purpose of projecting feature space and label space to low-dimensional spaces. The prediction can be done on these two subspaces before transforming back to the high-dimensional label space.

3.3 General Framework of Single and Dual Space Reduction

As single and dual space reduction approaches are reviewed in previous subsections, we found common steps in those aforementioned algorithms. Therefore, we present the general framework of single and dual space reduction as shown in Algorithm 1.

In the *pre-processing* phase (line 1–8), single and dual space reduction methods that are reviewed in previous sections can be employed. Most of these methods exploited statistical technique to transform high-dimensional feature (or label) space to feature (or label) subspace. After that feature subspace and/or label subspace are obtained along with the projection matrix as shown in line 2, 5 and 7.

In the *model training* phase (line 9–11), a classification model is built from the feature subspace or label subspace or both feature and label subspaces. Among existing methods on multi-label classification, binary relevance (BR) is a simple approach and widely used. BR reduces the multi-label classification task to a set of binary classifications and then builds a classification model for each class. While the label matrix $\mathbf{Y}$ has K dimensions, it is possible to build a classification model for each dimension; that is, K classification models are constructed for K dimensions. Moreover, each model, denoted by $h_k(\mathbf{X})$ is a classification model built for predicting each column $\mathbf{Y}[;,k]$ using the matrix $\mathbf{X}$ as shown in line 10 in the Algorithm 1.

In the *classification* phase (line 12–15), an unseen feature vector $\mathbf{X}_t$ is transformed to the low-dimensional feature vector if the feature space reduction approach is applied. Then this vector is sent to a series of classification models $h(\mathbf{X}_t)$ to estimate the value in each dimension of the label vector as shown in line 15.

In case of label space reduction or dual space reduction, the projection matrix is multiplied to reconstruct the high-dimensional label vector $\hat{\mathbf{Y}}$ using the *reconstructLabelSpace* function as shown in the *post-processing* phase (line 17). Finally, the set of predicted multiple labels is the union of the dimensions which have the value of 1.

Algorithm 1 General Framework of Single and Dual Space Reduction

```
Input: (1) a training set composed of
           (1.1) a feature matrix X
           (1.2) a label matrix Y
       (2) a test object X_t
       (3) a number of reduced features (D) and a number of reduced labels (K)
       (4) dimensionality reduction approach
Output: the predicted label matrix Ŷ for the test object X_t

[Pre-processing phase]
if the approach is Feature Space Reduction then
    runFeatureSpaceReduction(X, Y, D)
    set K = Q
else if the approach is Label Space Reduction then
    runLabelSpaceReduction(Y, K)
else if the approach is Dual Space Reduction then
    runDualSpaceReduction(X, Y, D, K)
end if

[Model training phase]
for k = 1 to K do
    h_k(X) ← modelConstruction(X, Y[:, k])
end for

[Classification phase]
if the approach is Feature Space Reduction then
    reduceFeatureSpace(X_t, D)
end if
Ŷ ← [h_1(X_t), h_2(X_t), ..., h_K(X_t)]

[Post-processing phase]
if the approach is Label Space Reduction or Dual Space Reduction then
    reconstructLabelSpace(Ŷ)
end if
```

4 Datasets and Experimental Settings

To evaluate the performance of these single and dual space reduction approach, the benchmark datasets are downloaded from MULAN.[1] Table 1 shows the characteristics of five multi-label datasets. For each dataset, N, M and Q denote the total number of objects, the number of features and the number of labels, respectively. L_C represents the *label cardinality*, the average number of labels per example and L_D stands for *label density*, the normalized value of *label cardinality* as introduced by [12].

Since each object in the dataset can be associated with multiple labels simultaneously, the traditional evaluation metric of single-label classification could not be applied. In this work, we apply two types of well-known multi-label evaluation metrics [9]. As the label-based metric, *hamming loss* and *macro F-measure*, are used to

[1] http://mulan.sourceforge.net/datasets.html.

Table 1 Characteristics of the datasets used in our experiments

Dataset	Domain	N	M		Q	L_C	L_D
			Nominal	Numeric			
Corel5k	Images	5,000	499	–	374	3.522	0.009
Enron	Text	1,702	1,001	–	53	3.378	0.064
Emotions	Music	593	–	72	6	1.869	0.311
Scene	Image	2,407	–	294	6	1.074	0.179
Yeast	Biology	2,417	–	103	14	4.237	0.303

evaluate each label separately. As the label-set-based metric, *accuracy* and *0/1 loss* are applied to assess all labels as a set. For more information about these evaluation metrics, readers are encouraged to see our previous work [8]. Using tenfold cross validation method, the results of the four evaluation metrics are recorded and shown in Table 2. All multi-label classification methods used in this work is implemented in R environment version 2.11.1[2] and linear regression is used for the regression model in the *model training* phase.

5 Performance Comparison on Single and Dual Space Reduction

This section presents performance comparison of single and dual space reduction. In this table, BR is used as a baseline that does not employ dimensionality reduction in both feature and label spaces. For single space reduction approach, MDDM, PLST, and MLC-BMad were selected to use in this study. The DDSR, 2SDSR-MP, 2SDSR-MB, 2SDSR-CP, 2SDSR-CB, and IDSR are dual space reduction techniques. The mean value of the *Hamming loss*, *macro F-measure*, *accuracy*, and *0/1 loss* metric of each algorithm are shown in Table 2. The best value in each row is emphasized by bold. The superscript $< x >$ in MDDM approach presents the percentage of reduced features comparing with the original feature size. Likewise, the superscript $[y]$ in PLST and MLC-BMaD approach mean the percentage of reduced labels, compared to the original. The numbers in the superscript (x, y) for DDSR, 2SDSR-MP, 2SDSR-MB, 2SDSR-CP, 2SDSR-CB, and IDSR represent the reduction percentage of dimensions in the feature space and that in the label space, respectively.

In Table 2, we observed that the feature reduction method is suitable for a dataset which has a large number of features, for example, the MDDM showed better performance, compared to BR when it was applied to the *enron* dataset. As more details, the *enron* has a large number of features (M = 1,001) but a small number of objects (N = 1,702). This situation might lead to the overfitting problem since there are not

[2] http://www.R-project.org/.

Table 2 Performance comparison (mean) between BR, MDDM, PLST, MLC-BMaD, DDSR, 2SDSR-MP, 2SDSR-MB, 2SDSR-CP, and 2SDSR-CB in terms of *Hamming loss* (*HL*), *macro F-measure* (*F1*), *accuracy* and *0/1 loss* on the five datasets

Dataset	Metrics	BR	MDDM	PLST	MLC-BMaD	DDSR	2SDSR-MP	2SDSR-MB	2SDSR-CP	2SDSR-CB	IDSR
Corel5k	HL ↓	0.0094	**0.0094**$^{<20>}$	**0.0094**$^{[20]}$	**0.0094**$^{[20]}$	0.0145$^{(30,40)}$	0.0131$^{(40,100)}$	0.0133$^{(20,20)}$	0.0142$^{(74.95,100)}$	0.0771$^{(14.99,20)}$	0.0150$^{(60,100)}$
	F1 ↑	0.0284	0.0284$^{<100>}$	0.0286$^{[40]}$	0.0299$^{[20]}$	0.1215$^{(15,20)}$	0.1065$^{(40,100)}$	**0.1282**$^{(20,40)}$	0.0948$^{(74.95,100)}$	0.1277$^{(14.99,20)}$	0.1143$^{(60,100)}$
	Accuracy ↑	0.0528	0.0528$^{<100>}$	0.0533$^{[40]}$	0.0537$^{[20]}$	0.1543$^{(30,40)}$	0.1352$^{(20,100)}$	**0.1716**$^{(20,40)}$	0.1279$^{(74.95,100)}$	0.1117$^{(74.95,100)}$	0.1502$^{(60,100)}$
	0/1 Loss ↓	0.9948	0.9948$^{<100>}$	0.9944$^{[20]}$	0.9962$^{[20]}$	0.9944$^{(45,60)}$	**0.8960**$^{(40,100)}$	0.9952$^{(20,20)}$	0.8974$^{(74.95,100)}$	0.9000$^{(74.95,100)}$	0.9948$^{(60,100)}$
Enron	HL ↓	0.1019	**0.0477**$^{<20>}$	0.0721$^{[20]}$	0.0746$^{[20]}$	0.0529$^{(3.18,60)}$	0.0576$^{(20,80)}$	0.0647$^{(20,100)}$	0.1234$^{(8.33,100)}$	0.1775$^{(8.33,100)}$	0.0532$^{(20,100)}$
	F1 ↑	0.1957	0.2559$^{<40>}$	0.2170$^{[40]}$	0.2628$^{[40]}$	**0.3118**$^{(5.29,100)}$	0.2925$^{(20,100)}$	0.2855$^{(20,60)}$	0.1902$^{(8.33,100)}$	0.2836$^{(3.33,40)}$	0.2926$^{(20,100)}$
	Accuracy ↑	0.3328	0.4132$^{<40>}$	0.3475$^{[20]}$	0.4369$^{[40]}$	**0.4723**$^{(5.29,100)}$	0.4406$^{(20,100)}$	0.3672$^{(20,100)}$	0.3111$^{(8.33,100)}$	0.4377$^{(3.33,40)}$	0.4617$^{(20,100)}$
	0/1 Loss ↓	0.9089	0.8790$^{<20>}$	0.9077$^{[60]}$	0.9206$^{[60]}$	**0.8713**$^{(5.29,100)}$	0.8831$^{(20,100)}$	0.9084$^{(20,100)}$	0.9166$^{(8.33,100)}$	0.8930$^{(8.33,100)}$	0.8766$^{(60,100)}$
Emotions	HL ↓	0.2068	0.2048$^{<80>}$	**0.2037**$^{[60]}$	0.2068$^{[100]}$	0.2728$^{(8.33,100)}$	0.2130$^{(100,80)}$	0.2178$^{(60,100)}$	0.2130$^{(4.24,80)}$	0.3269$^{(2.12,40)}$	0.2152$^{(100,60)}$
	F1 ↑	0.6317	0.6317$^{<100>}$	0.6339$^{[60]}$	0.6317$^{[100]}$	0.6455$^{(8.33,100)}$	0.6757$^{(100,80)}$	**0.6813**$^{(60,100)}$	0.6757$^{(4.24,80)}$	0.5915$^{(5.29,100)}$	0.6804$^{(100,20)}$
	Accuracy ↑	0.5091	0.5091$^{<100>}$	0.5091$^{[100]}$	0.5091$^{[100]}$	0.5090$^{(8.33,100)}$	0.5516$^{(100,80)}$	0.5509$^{(60,100)}$	**0.5516**$^{(4.24,80)}$	0.4160$^{(5.29,100)}$	0.5534$^{(60,40)}$
	0/1 Loss ↓	0.7467	0.7435$^{<80>}$	**0.7300**$^{[60]}$	0.7467$^{[100]}$	0.8293$^{(8.33,100)}$	0.7316$^{(100,60)}$	0.7552$^{(60,100)}$	0.7316$^{(3.18,60)}$	0.9360$^{(2.12,40)}$	0.7417$^{(60,60)}$
Scene	HL ↓	0.1105	**0.1051**$^{<40>}$	0.1105$^{[100]}$	0.1105$^{[100]}$	0.1190$^{(1.63,80)}$	0.1140$^{(20,100)}$	0.1119$^{(20,100)}$	0.1213$^{(2.04,100)}$	0.3103$^{(0.82,40)}$	0.1113$^{(60,80)}$
	F1 ↑	0.648	0.6513$^{<40>}$	0.6480$^{[100]}$	0.6480$^{[100]}$	0.7046$^{(2,100)}$	0.7156$^{(20,100)}$	0.7162$^{(20,100)}$	0.6859$^{(2.04,100)}$	0.4574$^{(2.04,100)}$	**0.7201**$^{(20,80)}$
	Accuracy ↑	0.5302	0.5302$^{<100>}$	0.5302$^{[100]}$	0.5302$^{[100]}$	0.6109$^{(1.63,80)}$	0.6370$^{(20,100)}$	0.6371$^{(20,100)}$	0.6048$^{(2.04,100)}$	0.2962$^{(2.04,100)}$	**0.6451**$^{(20,80)}$
	0/1 Loss ↓	0.5186	0.5156$^{<80>}$	0.5186$^{[100]}$	0.5186$^{[100]}$	0.5106$^{(1.63,80)}$	0.4886$^{(20,100)}$	**0.4753**$^{(20,100)}$	0.5056$^{(2.04,100)}$	0.9273$^{(0.82,40)}$	0.4778$^{(40,80)}$
Yeast	HL ↓	0.2008	**0.2000**$^{<80>}$	0.2008$^{[100]}$	0.2010$^{[80]}$	0.2807$^{(13.59,100)}$	0.2686$^{(80,80)}$	0.2158$^{(100,40)}$	0.2711$^{(13.59,100)}$	0.2517$^{(2.72,20)}$	0.2626$^{(80,100)}$
	F1 ↑	0.4455	0.4489$^{<80>}$	0.4455$^{[100]}$	0.4451$^{[80]}$	0.4926$^{(2.72,20)}$	0.4913$^{(40,40)}$	**0.5133**$^{(40,60)}$	0.4822$^{(2.72,20)}$	0.5092$^{(8.16,60)}$	0.4927$^{(40,100)}$
	Accuracy ↑	0.5019	0.5030$^{<80>}$	0.5019$^{[100]}$	0.5009$^{[80]}$	0.4884$^{(8.16,60)}$	0.4930$^{(60,60)}$	**0.5154**$^{(40,40)}$	0.4877$^{(13.59,100)}$	0.4366$^{(5.44,40)}$	0.5021$^{(60,60)}$
	0/1 Loss ↓	0.851	**0.8465**$^{<60>}$	0.8506$^{[80]}$	0.8510$^{[60]}$	0.9259$^{(13.59,100)}$	0.9143$^{(40,100)}$	0.8701$^{(80,60)}$	0.9143$^{(10.87,80)}$	0.9520$^{(2.72,20)}$	0.9086$^{(80,100)}$

↓ indicates the smaller the better; ↑ indicates the larger the better; the superscript $< x >$ shows the percentage of dimensions reduced from the feature space, the superscript $[x]$ shows the percentage of dimensions reduced from the label space, and the superscript (x, y) shows the percentage of dimensions reduced from the original ones for the feature space and that for the label space

Table 3 Percentage of wins and average rank of each method

Method	Percentage of wins				Average rank			
	HL (%)	F1 (%)	Accuracy (%)	0/1 Loss (%)	HL	F1	Accuracy	0/1 Loss
BR	12.5	0	0	0	3.2	8	6.6	6
MDDM	**50**	0	0	**20**	**1.2**	7.2	5.6	4.2
PLST	25	0	0	**20**	2.4	7.4	6.2	4
MLC-BMaD	12.5	0	0	0	3.4	7.4	5.8	7.2
DDSR	0	20	20	**20**	7.4	3.4	4.8	5.6
2SDSR-MP	0	0	0	**20**	5.8	3.8	3.8	**3.4**
2SDSR-MB	0	**60**	**40**	**20**	6	**1.8**	3	6
2SDSR-CP	0	0	0	0	7.8	6	5.17	4.8
2SDSR-CB	0	0	0	0	9.2	5.8	8	7.6
IDSR	0	20	**40**	0	6.2	2.4	**1.67**	4

enough objects to present all attribute's patterns. In similar way, the label reduction methods could increase predictive performance when it applied to dataset whose the number of labels are high. As an example, this approach presented 6.71 % *macro F-measure* improvement when the MLC-BMaD with 40 % label subspace is applied to the *enron*. However, the dual space reduction approach tends to give better performance comparing to the feature space reduction and label space reduction. In addition, the 2SDSR-MB with 20 % of feature subspace and 40 % of label subspace can improve 10 % *macro F-measure* and 12 % *Accuracy* when it was applied to the *corel5k* ($M = 499$ and $Q = 374$). However, there is no method that perform well for all evaluation metrics. Therefore, we counted the the number of times that each algorithm provided the best performance compared to others and denoted as the percentage of wins in Table 3.

6 Meta-Learning from Experimental Results

Meta-learning approach has been widely used to investigate the reason why classification method tends to give better performance in particular dataset [1, 2]. Motivated by these previous works, this paper uses characteristics of datasets as meta-features and the multi-label classification method that gives the best performance for each evaluation metric as meta-label. As more details, the complete list of meta-features are described in Supplementary website.[3] Instead of use five datasets as training examples, the meta-features are computed from datasets used in ten-fold cross-validation scheme, therefore, the meta-dataset has 50 training examples.

[3] http://dataminingtrend.com/kicss2013/supplementary.pdf.

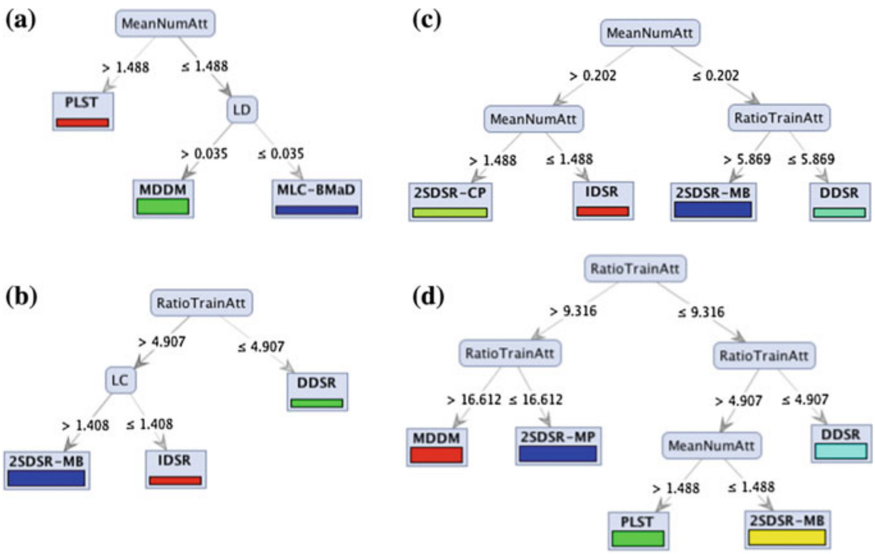


Fig. 2 Meta-model decision trees for **a** *Hamming loss*, **b** *macro F-measure*, **c** *accuracy* and **d** *0/1 loss*

In order to find relationship between dataset characteristics and multi-label classification method, the decision tree model is built. Figure 2, presents meta-model decision trees for each evaluation metric. These trees were generated by RapidMiner Studio[4] which is a widely used open source software for data mining. From these meta-models, we can guide the particular multi-label classification approach to the new dataset. For example, if the new dataset has the ratio of number of training examples to the number of features (RatioTrainAtt) less than or equal to 4.907, the DDSR approach could give the best performance in terms of *macro F-measure*.

7 Conclusions

This work aim to review dimensionality reduction techniques in multi-label classification. These techniques are grouped into two categories: single space reduction and dual space reduction. The goal of former approach is to reduce complexity in either feature space or label space by transforming high-dimensional feature (or label) space into feature (or label) subspace. The latter approach intent to reduce complexity in both feature and label spaces by transforming both spaces into the low-dimensional spaces. By comparative studies with ten-dimensionality reduction techniques on a broad range of multi-label datasets, we noticed that the dual space reduction approach tends to give better performance compared to single space reduction.

[4]http://rapidminer.com.

Acknowledgments This work has been supported by the TRF Royal Golden Jubilee Ph.D. Program [PHD/0304/2551]; and the Government Research Fund via Thammasat University, Thailand, and the National Research University Project of Thailand Office of Higher Education Commission.

References

1. An, A., Cercone, N.: Rule quality measures for rule induction systems: description and evaluation. Comput. Intell. **17**(3), 409–424 (2001)
2. Chekina, L., Rokach, L., Shapira, B.: Meta-learning for selecting a multi-label classification algorithm. In: IEEE 11th International Conference on Data Mining Workshops (ICDMW), pp. 220–227, Dec 2011
3. Golub, G., Reinsch, C.: Singular value decomposition and least squares solutions. Numerische Mathematik **14**, 403–420 (1970)
4. Hsu, D., Kakade, S., Langford, J., Zhang, T.: Multi-label prediction via compressed sensing. In: Proceedings of the Advances in Neural Information Processing Systems, vol. 22, pp. 772–780 (2009)
5. Madjarov, G., Kocev, D., Gjorgjevikj, D., Deroski, S.: An extensive experimental comparison of methods for multi-label learning. Pattern Recogn. **45**(9), 3084–3104 (2012)
6. Miettinen, P.: The boolean column and column-row matrix decompositions. Data Min. Knowl. Disc. **17**(1), 39–56 (2008)
7. Pacharawongsakda, E., Theeramunkong, T.: Towards more efficient multi-label classification using dependent and independent dual space reduction. In: Proceedings of the 16th Pacific-Asia Conference on Knowledge Discovery and Data Mining(PAKDD), PAKDD'12, pp. 383–394 (2012)
8. Pacharawongsakda, E., Theeramunkong, T.: A two-stage dimensionality reduction framework for multi-label classification. In: Proceedings of the PAKDD Workshop on quality issues, measures of interestingness and evaluation of data mining models (QIMIE), QIMIE 2013, pp. 330–341 (2013)
9. Read, J., Pfahringer, B., Holmes, G., Frank, E.: Classifier chains for multi-label classification. Mach. Learn. 1–27 (2011)
10. Schapire, R., Singer, Y.: Boostexter: a boosting-based system for text categorization. Machine Learn. **39**(2/3), 135–168 (2000)
11. Tai, F., Lin, H.T.: Multi-label classification with principle label space transformation. In: Proceedings of the 2nd International Workshop on Learning from Multi-Label Data (MLD'10), pp. 45–52 (2010)
12. Tsoumakas, G., Katakis, I., Vlahavas, I.: Mining Multi-label Data. Data Mining and Knowledge Discovery Handbook, 2nd edn. Springer, New York (2010)
13. Wang, H., Ding, C., Huang, H.: Multi-label linear discriminant analysis. In: Proceedings of the 11th European Conference on Computer Vision: Part VI, pp. 126–139. Springer, Berlin (2010)
14. Wicker, J., Pfahringer, B., Kramer, S.: Multi-label classification using boolean matrix decomposition. In: Proceedings of the 27th Annual ACM Symposium on Applied Computing, pp. 179–186. Springer, Berlin (2012)
15. Yu, K., Yu, S., Tresp, V.: Multi-label informed latent semantic indexing. In: Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 258–265 (2005)
16. Zhang, M.L., Zhou, Z.H.: A review on multi-label learning algorithms. IEEE Trans. Knowl. Data Eng. **99**(PrePrints), 1 (2013)
17. Zhang, Y., Zhou, Z.H.: Multilabel dimensionality reduction via dependence maximization. ACM Trans. Knowl. Disc. Data (TKDD) **4**(3), 1–21 (2010)

Feature Selection Using Cooperative Game Theory and Relief Algorithm

Shounak Gore and Venu Govindaraju

Abstract With the advancements in various data mining and social network-related approaches, datasets with a very high feature—dimensionality are often used. Various information theoretic approaches have been tried to select the most relevant set of features, and hence bring down the size of the data. Most of the times these approaches try to find a way to rank the features, so as to select or remove a fixed number of features. These principles usually assume some probability distribution for the data. These approaches also fail to capture the individual contribution of every feature in a given set of features. In this paper, we propose an approach which uses the Relief algorithm and cooperative game theory to solve the problems mentioned above. The approach was tested on NIPS 2003 and UCI datasets using different classifiers and the results were comparable to the state-of-the-art methods.

Keywords Feature selection · Game theory · Shapley values · Relief algorithm

1 Introduction

A lot of modern day machine learning problems, such as social network clustering, gene array analysis, and various other bioinformatic problems have to deal with a lot of data in the form of a large number of features. Feature selection is a process of selecting a subset of features from the original features such that it boosts the performance of the task at hand by either improving the accuracy or reducing the running time. Feature selection is defined by many authors by looking at it from various angles. But many of these definitions are similar in intuition and/or content. The following lists those that are conceptually different and cover a range of definitions [6].

S. Gore (✉) · V. Govindaraju
Department of Computer Science University at Buffalo, Buffalo, NY 14260, USA
e-mail: shounakg@buffalo.edu

V. Govindaraju
e-mail: govind@buffalo.edu

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_30

- *Idealized*: find the minimally sized feature subset that is necessary and sufficient to the target concept [17].
- *Classical*: select a subset of M features from a set of N features, $M < N$, such that the value of a criterion function is optimized over all subsets of size M [23].
- *Improving Prediction accuracy*: the aim of feature selection is to choose a subset of features for improving prediction accuracy or decreasing the size of the structure without significantly decreasing prediction accuracy of the classifier built using only the selected features [16].
- *Approximating original class distribution*: the goal of feature selection is to select a small subset such that the resulting class distribution, given only the values for the selected features, is as close as possible to the original class distribution given all feature values [16].

The different feature selection algorithms based on the above ideologies fall under three broad categories, namely, the filter method, the wrapper method, and a hybrid method [22]. The filter methods are classifier independent while the wrapper and hybrid methods depend on some classifier. The wrapper methods use a predetermined classification algorithm to measure the performance of the feature subset. Due to this, the wrapper methods may produce a subset of features that is well suited only for the predetermined classification algorithm. Filter methods [7] separate the classification and feature selection components, and define a heuristic scoring criterion to act as a proxy measure for the scoring accuracy. Filters evaluate statistics of the data independent of any particular classifier, thereby, extracting features that are generic, having incorporated few assumptions [2]. Hybrid or embedded method [13] tries to make use of the advantages of both the above methods by exploiting their different evaluation criteria in different search stages [21].

As can be seen from the description of three methods, each method has some advantages and disadvantages. Filter methods are independent of the learning algorithms and assess the relevance of features by looking only at the intrinsic properties of the data [27]. Filter methods are fast as compared to any of the other methods and are also computationally less expensive. A variety of evaluation criteria have been used for this purpose. Mutual information has been one such widely used and researched evaluation criteria [3, 8, 11, 14, 15, 20, 24] However, most of these selectors discard features which are highly correlated to the selected ones although relevant to the target class, which is likely to ignore some features having strong discriminatory power as a group but weak as individuals [28]. The main reason for this disadvantage is that information theory-based measurements fail to find out how the features collaborate with each other and what the contribution of each feature is toward the collaboration.

To solve the above problem, we propose a hybrid method which uses a filter method called the Relief algorithm [18] along with the game theoretic principle called Shapley values. The Shapley values help to find the contribution of each feature in a collaboration, thus allowing us to overcome the disadvantage of standard filter methods. Shapley values have been successfully used as a wrapper method for feature selection by Cohen et al. [4, 5]. Cohen et al. used multi-perturbation Shapley analysis along with the accuracy of a predefined classifier to find the contribution of each feature [4]. However, due to the dependence on the classifier, like all wrapper methods, this does not give a very generalized subset of features. Another disadvantage of this method is its high computational complexity. We therefore use a computationally inexpensive variation of the Relief algorithm framework to evaluate the contribution of the features. Relief algorithm also gives a strong sense of how well a feature or a group of features discriminates between different classes.

The rest of the paper is organized as follows: Sect. 2 talks about some basic concepts related to the Relief algorithm and cooperative game theory. We use these concepts to build our algorithm. The proposed algorithm is discussed in details in Sect. 3. Section 4 discusses about the datasets used. Section 5 discusses the parameter selection and the various experimental results. Section 6 summarizes the results and talks about the future work.

An earlier version of this paper was presented at the International Conference KICSS 2013 [12].

2 Background

Feature selection is an important preprocessing step in a lot of machine learning approaches, especially in the clustering and classification task. Both these tasks have one thing in common. A cluster or class should have minimal intra-cluster distance, or in other words, all the elements of a class should be as similar as possible. The Relief algorithm proposed by Kira et al. [18] and its many variants are based on this ideology of low intra-cluster and higher inter-cluster distances. According to the Relief algorithm, a feature is a good feature if it can easily segregate different classes from one another.

In game theory, coalitional or cooperative game is a game where a group of players come together in a cooperative way in order to form a consensus. The coalitional game is hence a game where a group of players try to come together in order to enforce that a given output is achieved and in the process maximize their own profits. In the next section, we describe both these ideologies in more details.

2.1 Relief Algorithm

The key ideas on which Relief algorithm is based on is that the nearest examples of same class should be closer than the nearest example of any other class. The Relief algorithm makes use of this fact to rank the features. The first version as proposed by Kira et al. worked only for a binary data. Since then, many variants like the Relief-F [19] for classification problems and Relief-F [25] for regression have been proposed. The general structure of the Relief algorithm is described in Algorithm 1. The function *diff* can be any function which calculates the distance between the two points. The proposed algorithm is loosely based on the general structure of the Relief algorithm.

Algorithm 1: General Structure of the feature selection process based on Relief algorithms

Data: an m X n matrix
Result: weights assigned to every feature
initialize all the weight values W_j to zero;
for *1 to m* **do**
    select a random example say k;
    for this example find the nearest observation from the same class, named h;
    for the example find the nearest observation from the some other class, named m;
    for *j = 1 to n* **do**
        Update $W_{\mathrm{j}} \leftarrow W_{\mathrm{j}} + diff(x_{\mathrm{kj}}, x_{\mathrm{mj}}) - diff(x_{\mathrm{kj}}, x_{\mathrm{hj}})$;
    end
end

2.2 Shapley Values for Feature Selection

In cooperative game theory, a coalition is defined as a subset of players, $\mathrm{N} = \{1, 2, 3, \ldots, \mathrm{n}\}$, which is able to make a binding agreement. Any subset of N, including N itself, can form a coalition [10]. Cooperative game theory introduces a concept of coalitional game which can be defined by a pair (N, v) where $\mathrm{N} = \{1, \ldots, \mathrm{n}\}$ is the set of all players and v(S), for every $S \subseteq N$, is a real number associating a worth with the coalition S. Game theory further pursues the question of representing the contribution of each player to the game by constructing a value function, which assigns a real value to each player. The values correspond to the contribution of the players in achieving a high payoff [4].

Shapley value as defined by Shapley [26] gives an elegant way to calculate the contribution of every player in a coalition. Shapley value measures the distribution of the power among players in the voting game [10]. The Shapley value considers the interaction between the players, and hence it captures the correlation between different players. The Shapley value $\phi_i(v)$ corresponding to the payoff for player i is given by the formula

$$\phi_i = \frac{1}{n!} \sum_{\pi \epsilon \prod} \Delta_i(S_i(\pi)), \tag{1}$$

where $\prod$ is the set of permutations over N and $S_i(\pi)$ is the set of players appearing before the *i*th player in the permutation π. Δ_i is called the marginal contribution of player *i* and is given by the formula

$$\Delta_i = v(S \bigcup \{i\}) - v(S). \tag{2}$$

The Shapley value is ideal to be adapted to the feature selection problem because it takes into consideration the interaction between players which a lot of other feature selection algorithms fail to do. Hence, we propose an algorithm which makes use of the best features of both the above approaches.

Algorithm 2: General Structure of the proposed feature selection algorithm based on Shapley values and Relief Algorithm

Data: The training data
Result: A subset of the features
initialize all the features in the feature set F to be selected;
intialize d = SomeNumber;
while $NewAccuracy >= \alpha * OldAccuracy$ **do**
  for the currently selected set of features, create the training and test sets;
  calculate the value $V(s)$ for each of the permutation using the Relief algorithm;
  calculate the Shapley value for the features;
  eliminate features from the feature set F when new importance of a feature (*newImp*) is less than the old importance of the feature (*oldImp*);
  $newImp_f < \beta * oldImp_f$. ;
  increment the value of d for the next iteration.;
end

3 Proposed Algorithm

The proposed approach we combines the best points of the Shapley value and the Relief algorithm to develop our algorithm. The typical feature selection algorithms are either forward selection or backward elimination. The forward selection algorithm starts with no features at all and subsequently adds one or more features in every iteration. Backward elimination starts with the entire set of features, or a considerably big subset of features and then eliminates one or more feature at a time. The proposed algorithm follows a backward elimination strategy.

In this algorithm, we first use the Relief Algorithm to decide how well a feature discriminates between different classes. However, finding the nearest neighbor for every point is computationally very expensive. Hence, instead of finding the nearest point from the same and a different class, we find out the nearest class centroid for the same and different class. We use this method to find a score for individual features and a combination of features. It is used to calculate the Shapley value of each of the remaining features. The Shapley value is used to decide if the feature should be retained in the subsequent iterations or not.

As can be seen in the original formula of calculating the Shapley values, all the different permutations in which the feature might be selected are considered to calculate the final Shapley value. This is a P-complete problem. Thus, we use the Shapley value approximation, which approximates the Shapley values to a fair extent using the following optimization of the original formula

$$\phi_i = \frac{1}{n!} \sum_{\pi \epsilon \prod_d} \Delta_i(S_i(\pi)), \tag{3}$$

where d is the number of players (features) which can interact with one another at a given point of time. $d = 1$ is as good as saying there is no interaction between the players at all while calculating the feature weight. While the interaction $d \backsimeq \sqrt{n}$ is what a lot of ensemble of classifiers like Random forests use. The general structure of the proposed feature selection algorithm using Shapley values and Relief method can be seen in the Algorithm 2. The parameter α is the control parameter used to decide when to stop the iterations of the algorithm. $\alpha = 1$ denotes that the accuracy of the classification algorithm increases at every iteration and the algorithm stops at the iteration when the accuracy stops to increase.

4 Datasets

The proposed algorithm was tested using the various datasets from the UCI machine learning repository [9]. Some of the datasets had missing values. The missing values were replaced by a random value in the range of $mean \pm covariance$. A dataset can have a lot of different variations in terms of number of samples (M), number of classes (C) and the number of features (N). We selected the datasets which gave us the maximum variation based on these parameters. The details about each dataset are mentioned in the Table 1.

As can be seen from Table 1, we selected datasets which allow us to validate the performance of our algorithm on both two-class and multi-class problems. Also the selected datasets are such that we can see the performance of the proposed algorithm in both the extreme cases, namely, when $M \gg N$ and also when $N \gg M$. Another

Table 1 Datasets used

Name	Examples	Features	Classes
Arcene	100	10,000	2
Arrhythmia	452	279	16
Dexter	300	20,000	2
Dorothea	800	100,000	2
Gisette	6000	5000	2
Glass identification	214	10	7
Isolet	7797	617	26
Lung cancer	32	56	3
Madelon	2000	500	2
Musk	6598	168	2
Optical recognition	5620	64	10

The various datasets used alongwith the number of samples, number of features and number of classes in each point to note is that the datasets used cover a wide array of real-life problems. A lot of these problems do require feature selection as a pre-processing step to make the total process either fast or computationally inexpensive.

5 Experiments—Parameter Selection and Results

We used a variety of different linear and nonlinear classification algorithms to test the performance of the proposed method. We use k-nearest neighbor classifier, support vector classifier, and the classification trees for the multi-class classification problems. We used perceptron learning in addition to the above-mentioned classifiers to test the performance of the two-class classification problems. All the programming was done in *Python*. The standard classification algorithms implemented in the machine learning module in Python, *mlpy* [1] were used.

Some of the datasets that we use have a separate training and test data. In such cases, we have used the test data to test the performance of the proposed approach. In other datasets only the training data is available. For cases like these, we use a ten-fold cross-validation approach. The cross-validation process is repeated 10 times and then the accuracy is averaged out. The accuracies thus mentioned in the tables below are either the accuracy on the test set or the averaged cross-validation accuracy. A variety of different experiments were performed with the datasets mentioned above to decide the most optimal value for the three different parameters namely α, β and d.

5.1 *Parameter d*

In the proposed approach, the parameter d controls the number of features interacting with one another at a given time. d decides the number of permutations that are considered to find out the contribution of any feature. For example, if there are m sample observations, the current value of d is k and there are n features, then we have to look at ${}^nP_k$ permutations to find out the contribution of each feature. The number of computations is hence of the order of $O(mn^k)$. For a very large value of n its good to keep the value of d as less as possible. Thus, to optimize this computational cost, we start with a small d, eliminate some features and then do a step-wise increase in the value of d. This ensures that the number of computations are optimized to the maximum limit.

Because the number of computations depends on the number of samples, m, and number of features presently selected, n, we do not increment the value of d until one of two conditions is satisfied (i) the value $m * n$ is below a certain threshold value ($v1$) or (ii) the ratio of the number of features left to the original number of features is less than a threshold value ($v2$). The thresholds ($v1$) and ($v2$) help us discard a lot many features assuming the features are independent or have a very small amount of interaction with other features. Once the number of features is sufficiently less,

Table 2 Results for d = 1 and $\alpha = 1$

Name	knn	svm	c-tree	Perc	Time
Arcene	78.8 % (451)	81.3 % (**340**)	**85.3** % (4401)	81.3 % (2391)	45.2
Arrhythmia	77.19 % (31)	**80.21** % (**23**)	74.96 % (29)	–	20.4
Dexter	93.20 % (1264)	**95.64** % (**73**)	94.95 % (201)	90.42 % (1023)	94
Dorothea	81.26 % (2413)	84.72 % (**129**)	83.96 % (1219)	**90.51** % (254)	251
Gisette	92.34 % (142)	95.22 % (134)	89.41 % (223)	**95.63** % (120)	39.6
Glass	**61.76** % (4)	55.05 % (3)	58.19 % (**2**)	–	2.4
Isolet	71.90 % (41)	**76.3** % (58)	75.11 % (**37**)	–	15.12
Lung cancer	**51.32** % (4)	38.14 % (3)	42.97 % (**2**)	–	1.61
Madelon	82.75 % (97)	**86.26** % (100)	85.38 % (54)	86.25 % (**38**)	7.34
Musk	90.11 % (35)	91.47 % (**27**)	92.13 % (29)	**93.15** % (46)	11.42
Optical	91.52 % (34)	90.13 % (**21**)	**92.43** % (26)	–	8.19

The accuracies for d = 1 and $\alpha = 1$ (number of features selected) and the average time. The data in bold shows the highest accuracy and the lowest number of features selected for each dataset

Table 3 Results for d = 4 and $\alpha = 1$

Name	knn	svm	c-tree	Perc	Time
Arcene	80.1 % (404)	82.1 % (**272**)	**85.4** % (2814)	81.3 % (894)	801.23
Arrhythmia	75.03 % (29)	**81.27** % (**18**)	80.19 % (25)	–	1027.5
Dexter	94.22 % (941)	**96.19** % (**54**)	96.1 % (95)	94.89 % (567)	3997.23
Dorothea	77.81 % (1987)	83.13 % (**94**)	83.01 % (628)	**85.71** % (208)	19745.37
Gisette	94.19 % (112)	92.9 % (**83**)	95.21 % (101)	**95.78** % (110)	1562.1
Glass	**60.42** % (3)	53.42 % (3)	56.42 % (**2**)	–	71.34
Isolet	74.93 % (37)	**77.59** % (35)	77.24 % (**32**)	–	1152.8
Lung cancer	**46.23** % (4)	39.92 % (3)	44.27 % (**2**)	–	194.37
Madelon	85.52 % (58)	85.67 % (78)	**88.84** % (51)	88.36 % (**34**)	657.89
Musk	91.83 % (30)	91.76 % (23)	93.21 % (**21**)	**94.78** % (27)	926.8
Optical	92.71 % (30)	93.12 % (22)	**98.31** % (**21**)	–	593.57

The accuracies for d = 4 and $\alpha = 1$ (number of features selected) and the average time. The data in bold shows the highest accuracy and the lowest number of features selected for each dataset

we increase the value of d, and hence allow interactions between more number of features. A number of different experiments were done to find the best possible values that ($v1$) and ($v2$) can take and it was found out that $v1 = 20{,}000$ and $v2 = 0.5$ gave the best possible results in term of accuracy and also keeping the computational time optimal. With the values of ($v1$) and ($v2$) fixed at 20,000 and 0.5, respectively, we did experiments to find out the most effective staring value of the parameter d. Because the number of features varies between 10 and 100,000, we selected the initial values of d to be between 1 and 4. The results obtained for these combinations are shows in Tables 2 and 3.

From the data seen from Tables 2 and 3 we can see that the accuracy increases atleast slightly when the initial value of d increases. However, it should also be noted that the processing time also increases with an increase in the value of d. In a lot of machine learning approaches, time is of essence and hence we need to select the initial value d such that it optimizes the running time as well. In the rest of the approaches, we select the initial value of d to be one, so as to keep the processing time to be as low as possible.

5.2 Parameter α

In a lot of machine learning and data mining tasks, there often is a need to have a trade-off between the overall data to be processed and the overall efficiency of the task at hand. A small compromise on the accuracy is acceptable in cases where the final dataset has to be as small as possible. The parameter α allows us to tune the algorithm for this trade-off. As it can be seen in the proposed algorithm, we stop the iteration of the algorithm when $NewAccuracy \geq \alpha * OldAccuracy$. The *OldAccuracy* is the highest accuracy seen till a given point of time. $NewAccuracy \geq \alpha * OldAccuracy$ thus ensures that we continue to iterate through the algorithm only if the accuracy obtained in the current iteration, *NewAccuracy*, is greater than the highest accuracy times some constant α.

$\alpha = 1$ simply means that the accuracy increases at every iteration and we stop as soon as the accuracy stops increasing. This ensures that the algorithm stops atleast at a local maxima of the possible accuracies. $\alpha = 1$ also allows the algorithm to be fast since we stop as soon as the accuracy stops increasing. This is good if we want to keep the processing time as low as possible. But because we have a strict threshold, the number of features selected might not be very low. Thus we still are left with a lot of data to handle, even after the feature selection process. Table 2 shows the results when $\alpha = 1$.

If we relax this constraint, that is reduce the value of the parameter α, we can reduce the number of features selected even more. There is a compromise on the accuracy, however we reduce the final data we need to handle. This is particularly useful when the number of samples M is huge or the number of features N is way more than the number of samples M, $N \gg M$. Reducing the value of α means that the algorithm stops its iterations when the accuracy of the algorithm goes below a fraction of the highest possible accuracy. Table 4 shows the results for $\alpha = 0.95$. We can thus state with confidence that on most occasions, a smaller α leads to a lower number of features being selected.

5.3 Parameter β

We eliminate features from the feature set F when new importance of a feature (*newImp*) is less than the old importance of the feature (*oldImp*) $newImp_f < \beta * oldImp_f$. In traditional coalitional game theoretic approach, a new coalition is formed

Table 4 Results for d = 1 and $\alpha = 0.95$

Name	knn	svm	c-tree	Perc	Time
Arcene	76.2 % (**231**)	80.1 % (312)	**83.9** % (2914)	80.1 % (2109)	48.3
Arrhythmia	77.19 % (31)	**78.54** % (**22**)	73.15 % (27)	–	23.6
Dexter	91.80 % (746)	**95.64** % (**63**)	94.95 % (201)	90.42 % (1023)	94
Dorothea	81.26 % (2413)	82.12 % (**42**)	83.96 % (1219)	**89.78** % (215)	298.4
Gisette	90.98 % (96)	**95.22** % (134)	90.78 % (123)	94.8 % (**93**)	56.2
Glass	**61.76** % (4)	55.05 % (3)	58.19 % (**2**)	–	2.4
Isolet	71.56 % (40)	**75.11** % (**37**)	**75.11** % (**37**)	–	18.3
Lung cancer	**51.32** % (4)	38.14 % (3)	42.97 % (**2**)	–	1.61
Madelon	81.75 % (74)	**85.76** % (**37**)	85.03 % (53)	**85.76** % (**37**)	13.72
Musk	89.14 % (32)	91.13 % (**25**)	89.14 % (32)	**92.73** % (42)	14.38
Optical	**90.48** % (32)	90.13 % (**21**)	**90.48** % (32)	–	12.51

The accuracies for d = 1 and $\alpha = 0.95$ (number of features selected) and the average time. The data in bold shows the highest accuracy and the lowest number of features selected for each dataset

only when the new coalition allows the players forming the coalition to have higher profits than any prior coalition. We use the parameter β to attain this particular effect. $\beta = 1$ takes care of strictly enforcing this particular condition. In the proposed algorithm, parameter d controls the level of interaction amongst the features. This may reduce the importance of the feature in subsequent iterations of the algorithm. The parameter β allows us to overcome this effect. β can take the values between the range of 0 and 1. From various experiments, we concluded that the best possible value for β is 0.85.

5.4 Results

In Table 5 we summarize the best results obtained for each of the dataset in terms of the accuracy and compare it with various state-of-the-art results for the same dataset.We can see that our proposed approach out performs the state-of-the-art methods on some datasets, while for the other datasets we have comparable accuracy. The proposed approach sometimes fails to beat the state-of-the-art performance. But it can be easily explained by the use of the parameters α, β and d. One important objective of the proposed approach was to come up with a method which can be easily tuned as per the needs. Hence, if we tune the parameters in order to achieve only the maximum accuracy possible, we might be able to beat the state-of-the-art results.

Table 5 Comparison of the best results with the state-of-the-art results

Name	Highest accuracy (%)	State-of-the-art (%)
Arcene	85.4	86 [4]
Arrhythmia	81.92	84.2 [4]
Dexter	96.19	94 [4]
Isolet	77.59	83.65 [28]
Musk	94.78	96.13 [28]
Optical recognition	98.31	98.72 [28]

The state-of-the-art results are compiled from various published results

6 Conclusions

The major contribution of the algorithm lies in the use of the three parameters d, α and β. Using these three parameters we have a complete control over the feature selection task. The parameter d allows us to control the level of interactions amongst various features. A higher d allows us to have a higher amount of accuracy while a lower d reduces the number of computations drastically. α is useful when we have a tradeoff between the final data at hand and the overall accuracy of the system. We prove it experimentally that we can easily change the value of α depending on the problem at hand. β allows us to from a coalition of features such that their overall contribution to the coalition increases, which also means that the importance of the feature towards the overall coalition increases.

As can be seen from the above results, the proposed algorithm is close to the state-of-the-art feature selection techniques in most of the cases and beats some of the others either when it comes to processing time or the accuracy achieved by the classification algorithm. The main purpose of the paper was to come up with a novel coalition-based algorithm which lets the users have the control on deciding what is more important, the accuracy or the speed. As shown in the experiments, we achieve the same successfully. This proves that the proposed approach can easily be used to do a smart feature selection for a lot of real-world problems.

References

1. Albanese, D., Visintainer, R., Merler, S., Riccadonna, S., Jurman, G., Furlanello, C.: mlpy: Machine Learning Python (2012)
2. Brown, G., Pocock, A., Zhao, M., Lujan, M.: Conditional likelihood maximization: a unifying framework for information theoretic feature selection. JMLR **13**, 27–66 (2012)
3. Cai, R., Hao, Z., Yan, X., Wen, W.: An efficient gene selection algorithm based on mututal information. Neurocomputing **72**, 991–999 (2009)
4. Cohen, S., Dror, G., Rupin, E.: Feature selection via coalitional game theory. Neural Comput. **19**, 1939–1961 (2007)
5. Cohen, S., Rupin, E., Dror, G.: Feature selection based on the Shapley Value. In: Proceedings of the Nineteenth IJCAI, pp. 665–670. Morgan Kaufmann Publishers (2005)

6. Dash, M., Liu, H.: Feature selection for classification. Intell. Data Anal. Elsevier **1**(3), 131–156 (1997)
7. Duch, W.: Feature extraction: foundations and applications. Chapter 3, Studies in Fuzziness and Soft Computing, pp. 89–117. Springer (2006). ISBN: 3-540-35487-5
8. Fleuret, F.: Fast binary feature selection with conditional mutual information. JMLR **5**, 1531–1555 (2004)
9. Frank, A., Asuncion, A.: UCI Machine Learning Repository, Irvine. University of California, School of Information and Computer Science, CA (2010)
10. Friedman, W.: Game Theory with Applications to Economics. Oxford University Press, New York (1990)
11. Gomez-Verdejo, V., Verleysen, M., Fleury, J.: Information-theoretic feature selection for functional data classification. Neurocomputing **72**, 3580–3589 (2009)
12. Gore, S., Govindaraju, V.: Feature selection using cooperative game theory and relief algorithm. In: Skulimowski, A.M.J. (ed.) Looking into the Future of Creativity and Decision Support Systems, Krakow, Poland, 79, Nov 2013, Advances in Decision Sciences and Future Studies, vol. 2, pp. 114–125. Progress & Business Publishers, Krakow (2013)
13. Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L. (eds.): Feature Extraction: Foundations and Applications. Springer (2006). ISBN: 3-540-35487-5
14. Hild, K.E., Erdogmus, D., Torkkola, K., Principe, J.C.: Feature extraction using information theoretic learning. IEEE Trans. PAMI **28**, 1385–1392 (2006)
15. Huang, D., Chow, T.W.S.: Effective feature selection scheme using mutual information. Neurocomputing **63**, 325–343 (2005)
16. Jon, G.H., Kohavi, R., Peger, K.: Irrelevant features and the subset selection problem. In: Proceedings of the Eleventh ICML, pp. 121–129 (1994)
17. Kira, K., Rendell, L.A.: The feature selection problem: traditional methods and a new algorithm. In: Proceedings of Ninth National Conference on AI, pp. 129–134 (1992)
18. Kira, K., Rendell, L.A.: A practical approach to feature selection. In: Proceedings of ICML, pp. 249–256 (1992)
19. Kononeko, I.: Estimating attributes: ananlysis and extensions of RELIEF. ECML, pp. 101–106 (1998)
20. Kwak, N., Choi, C.-H.: Input feature selection by mutual information based on Parzen windows. IEEE Trans. PAMI **24**, 1667–1671 (2002)
21. Liu, H., Yu, L.: Toward integrating feature selection algorithms for classification and clustering. IEEE Trans. KDE **17**(4), 491–501 (2005)
22. Liu, J., Lee, S.: Study on feature select based on coalition game. In: Proceedings of the International Conference on Neural Networks and Signal Processing, pp. 445–450 (2008)
23. Narendra, P. M., Fukunaga, K.: A branch and bound algorithm for feature selection. IEEE Trans. Comput. **C-26**(9), 917–922 (1977)
24. Peng, H., Long, F., Ding, C.: Feature selection based on mutual information: criteria of max-dependency, max-relevance and min-redundancy. IEEE Trans. PAMI **27**, 1226–1238 (2005)
25. Robnik Sikonja, M., and Kononenko, I.: An adaptation of relief for attribute estimation in regression. In: Proceedings of the Fourteenth ICML, pp. 296–304 (1997)
26. Shapley, L.S.: A value for n-person games. In: Kuhn, A.W.T.H.W. (ed.) In Contributions to the Theory of Games, vol. 2. Princeton University Press, New York (1990)
27. Sun, X., Liu, Y., Li, J., Zhu, J., Liu, X., Chen, H.: Using cooperative game theory to optimize the feature selection problem. Neurocomputing **97**, 86–93 (2012)
28. Sun, X., Liu, Y., Li, J., Zhu, J., Liu, X., Chen, H.: Feature evaluation and selection with cooperative game theory. Pattern Recogn. **45**, 2992–3002 (2012)

Classification with Rejection: Concepts and Evaluations

Wladyslaw Homenda, Marcin Luckner and Witold Pedrycz

Abstract Standard classification process allocates all processed elements to given classes. Such type of classification assumes that there are only native and no foreign elements, i.e., all processed elements are included in given classes. The quality of standard classification can be measured by two factors: numbers of correctly and incorrectly classified elements, called True Positives and False Positives. Admitting foreign elements in standard classification process increases False Positives and, in this way, deteriorates quality of classification. In this context, it is desired to reject foreign elements, i.e., not to assign them to any of given classes. Rejecting foreign elements will reduce the number of false positives, but can also reject native elements reducing True Positives as side effect. Therefore, it is important to build well-designed rejection, which will reject significant part of foreigners and only few natives. In this paper, evaluations of classification with rejection concepts are presented. Three main models: a classification without rejection, a classification with rejection, and a classification with reclassification are presented. The concepts are illustrated by flexible ensembles of binary classifiers with evaluations of each model. The proposed models can be used, in particular, as classifiers working with noised data, where recognized input is not limited to elements of known classes.

Keywords Rejection rule · Binary classifiers ensemble · Reclassification

W. Homenda (✉) · M. Luckner
Faculty of Mathematics and Information Science, Warsaw University of Technology,
Plac Politechniki 1, 00-660 Warsaw, Poland
e-mail: homenda@mini.pw.edu.pl

W. Pedrycz
System Research Institute, Polish Academy of Sciences,
ul. Newelska 6, 01-447 Warsaw, Poland

W. Pedrycz
Department of Electrical and Computer Engineering,
University of Alberta Edmonton, Alberta T6G 2V4, Canada

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_31

1 Introduction

Pattern recognition is one of the leading subjects in the field of computer science in its both theoretical and practical aspects. For decades, it has been a subject of intense, purely theoretical research inspired by practical needs. The results have been published in prestigious scientific journals. Example applications are: recognizing printed text, manuscripts, music notation, biometric features, voice, speaker, recorded music, medical signals, images, etc.

The aspect of rejection of foreign element in pattern recognition is shallowly researched and not considered in practical applications. By foreign elements, we understand elements not belonging to recognized classes. Dissemination of technologies using pattern recognition increases the importance of identifying foreign elements. For example, in recognition of printed texts, foreign elements (blots, grease, or damaged symbols) appear in a negligible scale due to regular placement of printed texts' elements (letters, numbers, punctuation marks) and due to their good separability. These features of printed texts allow employing effective segmentation methods. However, in recognition of such sources as geodetic maps or music notation, problem of foreign elements is more important. Unlike printed text, such sources contain elements placed irregularly and overlapping native elements. Such elements are hardly distinguishable by size and shape analysis. Thus, strict rules of segmentation will result in rejection of many native symbols.

Due to weak separability of foreign and native elements of recognized sources, segmentation criteria must be more tolerant than in the case of printed texts in order not to reject native elements at the stage of segmentation. In consequence, more foreign elements are subjected to stages of recognition following segmentation and should be eliminated then. The problem of analysis of foreign elements is highly important in such domains as analysis of medical signals and images or recognition of geodesic maps or music notations (printed and handwritten) and its importance will be increasing in future.

The problem of rejecting foreign elements in recognition tasks is not present frequently in research and is rather rarely present in papers on pattern recognition. Assuming that classified elements are always native ones, i.e., they belong to one of recognized classes, and ignoring rejection of foreign elements is rather a standard attempt. Alike, papers describing practical applications of pattern recognition methods ignore the problem of foreign elements, what may come from insufficient theoretical research of this subject and limited abilities of existing rejection methods. There are significant exceptions, which show that the rejection problem cannot be disregarded, c.f. [12].

The motivation of this study arises from discussion on classification with rejection option. As outlined above, up-to-date research and practice still need conducting further studies on new aspects in the domain of pattern recognition. It is expected that research in this area will overcome technological barriers and will increase effectiveness in the areas mentioned above.

The paper is structured as follows. Related research and introductory remarks are presented in Sect. 2. In Sect. 3, various ensembles of classifications based on binary classifiers are shown. The discussion includes evaluation criteria of rejection and reclassification methods. In Sect. 4 an example is discussed. Conclusions and directions of further research are presented in Sect. 5.

2 Preliminaries

The problem discussed above (formally defined in Eq. 2) is important in practice. In Fig. 1, we present a fragment of a geodesic map raising classification errors. Assumption that all patterns subjected to classification are digits raises classification errors for background elements unless they are rejected. This observation clearly shows a need for a rejection option in classification tasks. The rejection option can be described by a classification function with the rejection value in addition to computed classes. Such function may return 0 when an object should be rejected or a number of recognized class otherwise.

2.1 Related Research

The theoretical foundations of rejection were created by Chow [2]. He created the optimal rejection rule that optimizes a classification error. The solution was limited to binary classifiers and presented as a method for Bayes classifiers. Recognized elements are rejected when distance to a discrimination plane is lower than a declared threshold. The optimal threshold is a compromise between a number of misclassified elements and rejected correct results.

Fig. 1 Examples of correctly classified digits (the *right part*) and background elements incorrectly classified as digits (the *left part*). Assumption that all patterns subjected to classification are digits raises classification errors for background elements unless they are rejected

The definition for a multi-class issue was presented by Ha [5]. The Chow's rule is calculated for all pairs of classes separated by a discrimination plane. There are also solutions for a linear multi-classification task [3]. However, most of theoretical works are limited to the binary case [10, 16].

The Chow's rule was redefined to practical applications as a rejection rule for support vector machines classifiers [7], where the probability is estimated as distance to discrimination plane.

The distance function can be also used to define a rejection rule based on dissimilarity between a class pattern and a recognized element [8].

Another practical trend is related to neural networks. The output of the neurons in feed-forward neural networks is determined by values of neurons in the output layer, where each neuron is equivalent to a recognized class. The class with a maximal value of the neurons' outputs is taken as the final classification result. This reject decision is issued when the value of the neuron is relative small [11]. In this approach, typical for fuzzy sets [4], when a new pattern cannot be classified as a single class with high certainty by a trained neural network, such a pattern is rejected in classification process, c.f. [6].

In natural way, a pattern can be rejected when two or more output neurons fire similar values. The same method can be used in all voting classifiers in a draw case [6]. Among ensembles of binary classifiers, the rejection option can be applied for ECOC (error-correcting output coding) classification systems [14].

The rejection problem has been raised in the recognition tasks of printed documents, namely recognition of music scores and geodetic maps. Rejection methods applied in such practical tasks, which in fact were created ad hoc or were implemented as dedicated methods of considered classifiers, increased effectiveness, and recognition rates, c.f. [9].

The above works outline highly important problem, which is solved neither in theory and research nor in practice. Solutions based on the Chow's rule are limited to the cases of misclassifications between classes. The optimal solution assumes that rejected symbols increase the false negative factor of error for one of recognized classes. Unfortunately, noised and disrupted input patterns may be misclassified diminishing recognition efficiency.

On the other hand dedicated solutions force choice of classification methods. Moreover, there is a doubt, if methods based on rare instances can be applied in practice or if a low level neuron's response means false positive case.

2.2 Classification and Rejection Functions

Standard classification is described as a mapping Φ from the Cartesian product of features $\mathbb{X} = \mathrm{X}_1 \times \mathrm{X}_2 \times \cdots \times \mathrm{X}_\mathrm{m}$ into the set of classes $\mathbb{C} = \{\mathrm{C}_1, \mathrm{C}_2, \ldots, \mathrm{C}_\mathrm{n}\}$. For the sake of simplicity we assume that the function Φ will take indexes of classes $i = 1, \ldots, n$ as its values instead of classes themselves. Since classification is usually parameterized, we will denote classification parameters by α. Therefore, a classification function will be finally denoted as

$$\boldsymbol{\Phi}_\alpha : \mathbb{X} \to \{1, 2, \ldots, n\}, \quad \boldsymbol{\Phi}_\alpha(\overline{x}) = i \tag{1}$$

for given $\overline{x} = (x_1, x_2, \ldots, x_m) \in \mathbb{X}$ and $i \in \{1, 2, \ldots, n\}$.

Therefore, classification is carried out using a classification function $\phi_\alpha(\overline{x}) = i$, which calculates an index $i \in \{1, \ldots, n\}$ of recognized class C_i based on classification parameters α.

This definition of classification always raises misclassification for foreign elements, when a classified element does not belong to any of recognized classes. Let us denote the set of feature vectors of such elements by $\mathbb{S}$. Then, for any $\overline{x} \in \mathbb{S}$ the classification function $\boldsymbol{\Phi}_\alpha$ raises error:

$$(\forall \overline{x} \in \mathbb{S})(\forall i = 1, 2, \ldots, n)\ (\boldsymbol{\Phi}_\alpha(\overline{x}) = i) \Rightarrow (x \notin C_i). \tag{2}$$

Therefore, we use the following classification function to avoid misclassification of objects not belonging to any class:

$$R_{\alpha,\beta} : \mathbb{X} \to \{1, 2, \ldots, n\} \cup \{0\}$$

$$R_{\alpha,\beta}(\overline{x}) = \begin{cases} 0 & \overline{x} \in \mathbb{S},\ \text{i.e. } \overline{x} \text{ should be rejected} \\ \boldsymbol{\Phi}_\alpha(\overline{x}) & \text{otherwise} \end{cases} \tag{3}$$

where β is a possible extra parameter affecting rejection and independent on α.

In some circumstances, the learning set must be considered in classification process. In such a case, a classification function is denoted as follows:

$$\boldsymbol{\Phi}_\alpha : \mathbb{X} \times \mathbb{L} \to \{1, 2, \ldots, n\}, \quad \boldsymbol{\Phi}_\alpha(\overline{x}, L) = i \tag{4}$$

where: $\mathbb{L}$ is a space of possible learning sets, L is a given learning set, and the rest of notation is compatible with formula Eq. 1.

Alike, when learning set must be considered, we denote a classification function with rejection as follows:

$$R_{\alpha,\beta} : \mathbb{X} \times \mathbb{L} \to \{1, 2, \ldots, n\} \cup \{0\}$$

$$R_{\alpha,\beta}(\overline{x}, L) = \begin{cases} 0 & \overline{x} \in \mathbb{S},\ \text{i.e. } \overline{x} \text{ should be rejected} \\ \boldsymbol{\Phi}_\alpha(\overline{x}, L) & \text{otherwise} \end{cases} \tag{5}$$

3 Classification with Rejection

In this section, we discuss different architectures of classification with rejection based on binary classifiers, e.g., on support vector machines. In Fig. 2, standard classification architecture for four classes is presented. Notice that architecture of classification is a kind of a (unbalanced) binary tree with $n - 1$ classifiers being internal nodes and n classes of elements being leaves. This architecture can easily be extended to more than four classes. Of course, we may have different configurations of classification architecture for higher number of classes. For instance, we may have "cascade" configuration in a form of highly unbalanced three with one path of internal nodes, i.e., one path of classifiers. Other configurations may be based on different binary trees with the same numbers of internal nodes (classifiers).

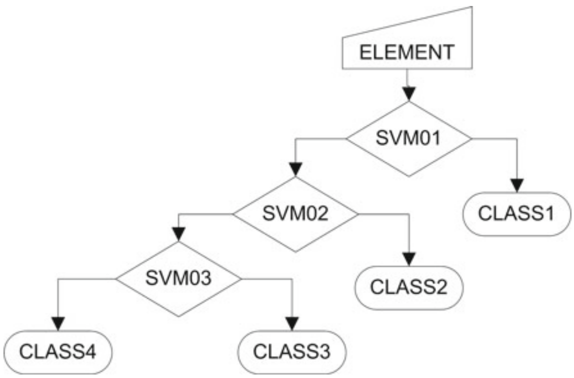
3.1 Architecture of Ensembles of Binary Classifiers with Rejection

In upper scheme of Fig. 3, a cascade configuration of classification with rejection is presented. Foreign elements are assumed to create additional class, which is classified at the beginning of the paths of binary classifiers. Then next classes are identified "step by step." Notice that this configuration may be slightly reorganized: elements of recognized classes may be identified first and then rejected elements may be identified at the end of the path of classifiers.

A balanced tree configuration of classification with rejection outlined in middle scheme of Fig. 3 allows for splitting the set of foreign elements for several subclasses. The number of subclasses of the class of foreign elements is equal to the number of recognized classes.

In general, in both configurations of upper and middle schemes of Fig. 3 rejection can be simply interpreted as standard classification with additional class(es)

Fig. 2 Standard classification, the space of objects split into recognized classes



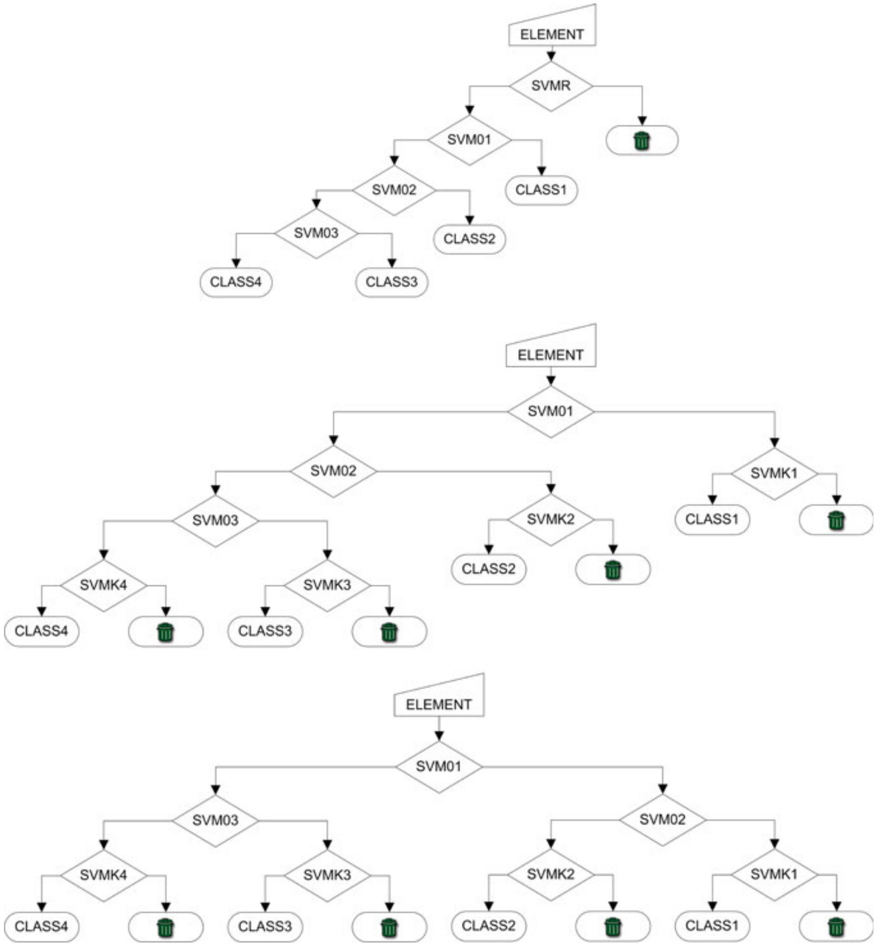


Fig. 3 Classification with rejection—balanced tree configuration with many classes of rejected elements

of foreign elements. Therefore, standard classification methods may be used in such kind of classification with rejection. Namely, supplementing the set of recognized classes with class(es) of foreign elements, we get standard classification. This method increases the number of classes by once or twice and this is the only one obstacle of processing foreign elements.

The method of distinguishing foreign elements may be used for optimization of classification without significant increment of complexity. Such a configuration, based on classifiers used in basic stage of classification, is presented in upper scheme of Fig. 4.

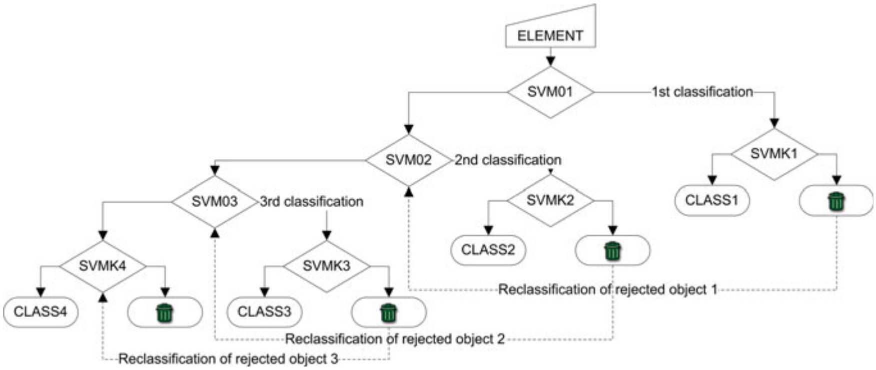


Fig. 4 Cascade configuration, reclassification of rejected elements with many classes of rejected documents

3.2 *Evaluation of Rejection Methods*

3.2.1 No Rejection

We assume that the set of recognized elements include classes $\mathbb{C} = \{\mathrm{C}_1, \mathrm{C}_2, \ldots, \mathrm{C}_\mathrm{n}\}$ of native elements to be recognized and the class C_0 of foreign elements to be rejected. Employing standard classification of formula Eq. 1, we get all elements classified to classes of $\mathbb{C}$. This classification is presented in Fig. 2. Also all foreign elements of the class C_0 will be incorrectly assigned to the classes of $\mathbb{C}$. Therefore, we can evaluate two factors of such classification: $T(rue)P(ositives)$ and $F(alse)P(ositives)$, i.e., correctly and incorrectly classified elements. These factors are defined by the following formulas:

$$TP_\Phi = \sum_{i=1}^{n} \sum_{\bar{x} \in C_i} \delta_{i\Phi_\alpha(\bar{x})} \qquad FP_\Phi = \textstyle\sum_{i=1}^{n} \sum_{\bar{x} \notin C_i} \delta_{i\Phi_\alpha(\bar{x})} \tag{6}$$

The factor FP includes all incorrectly classified native elements of classes $\mathbb{C}$ and all foreign elements of the class C_0. As outlined in Sect. 2.2, foreign elements of the class C_0 deteriorates classification rate. In order to improve classification, foreign elements should not be classified to any of the classes $\mathbb{C}$. Some attempts to such improvements are presented in the next sections.

3.2.2 Rejection

The first attempt to improve the classification presented in Fig. 2 relies on rejecting foreign elements at the beginning of classification process. Configurations of the simple classification with rejecting foreign elements are shown in Fig. 3. Now, these three configurations can be characterized by four factors: *TP*, *FP*, *TN* and *FN*. These factors are computed in formulas Eq. 7. In this formulas sgn denotes the signum function, i.e., sgn : $\mathbb{N} \rightarrow \{0, 1\}$, $\text{sgn}(x) = 0 \Leftrightarrow x = 0$ and $\overline{\text{sgn}} = 1 - \text{sgn}$.

$$TP_R = \sum_{i=1}^{n} \sum_{\overline{x} \in C_i} \delta_{i\Phi_\alpha(\overline{x})} \cdot \text{sgn}(R(\overline{x})) \qquad FN_R = \sum_{i=1}^{n} \sum_{\overline{x} \in C_i} \delta_{i\Phi_\alpha(\overline{x})} \cdot \overline{\text{sgn}}(R(\overline{x}))$$

$$FP_R = \sum_{i=1}^{n} \sum_{\overline{x} \notin C_i} \delta_{i\Phi_\alpha(\overline{x})} \cdot \text{sgn}(R(\overline{x})) \qquad TN_R = \sum_{\overline{x} \in C_0} \overline{\text{sgn}}(R(\overline{x}))\big) \tag{7}$$

Factors *TP*, *FP* are explained above. The factor *TN* counts all rejected foreign elements, while the factor *FN* counts all elements (native and foreign) incorrectly classified to classes of $\mathbb{C}$. Notice that these classifications' configurations decrease the factor *FP* by rejecting those foreign elements, which are incorrectly assigned to classes of $\mathbb{C}$ in the process of classification without rejection. Regrettably, the factor *TP* may also be decreased by incorrect rejection of native elements. An answer to the question about possible benefits of rejection depends on the quality of rejecting. If rejecting accuracy is high, then deterioration of correct classifications is smaller than improvements in incorrect classifications. In practice, rejection of foreign elements overheads incorrect rejection of native elements. Other two factors allow for better characterization of classification. In addition, it is worth to mention that construction and usage of such classifiers do not create difficulties comparing to standard classification without rejection.

3.2.3 Rejection and Reclassification in Cascade Configuration

Architectures of classification with rejection may allow for improving classification factors. In Fig. 4, rejected elements are subjected for reclassification. This solution is bounded to architecture and configuration of classification. Reclassification solutions discussed in this paper are based on cascade configuration shown in this Fig. 4. Below, the configuration is analyzed in detail.

In Fig. 4, elements rejected at the stage of class C_1 are subjected again to classification at the stage of class C_2 and further classes. Then, elements rejected at the stage of class C_2 are subjected again to classification at the stage of class C_3 and further classes, andnd so on.

Assume that we have n classes, $n - 1$ binary classifiers carrying the function $\boldsymbol{\Phi}_\alpha$ (i.e., SVM01, SVM02, ...) and n binary classifiers carrying the function $R_{\alpha,\beta}$ (i.e., SVMK1, SVMK2, ...). After the first reclassification, we have:

$$TP_{RR}^1 = TP_R + \sum_{i=2}^{n} \sum_{\overline{x} \in C_i} \delta_{1\Phi_\alpha(\overline{x})} \cdot \overline{\text{sgn}}(R_1(\overline{x})) \cdot \delta_{i\Phi_\alpha(\overline{x})} \cdot \text{sgn}(R(\overline{x}))$$

and after the first and second reclassification we have:

$$TP_{RR}^2 = TP_{RR}^1 + \sum_{i=3}^{n} \sum_{\bar{x} \in C_i} \delta_{1\Phi_\alpha(\bar{x})} \cdot \overline{\mathrm{sgn}}(R^1(\bar{x})) \cdot \delta_{2\Phi_\alpha(\bar{x})} \cdot \overline{\mathrm{sgn}}(R^2(\bar{x})) \cdot \delta_{i\Phi_\alpha(\bar{x})} \cdot \mathrm{sgn}(R(\bar{x}))$$

where $R^1(\bar{x})$ and $R^2(\bar{x})$ are rejecting classifiers corresponding to classes C_1 and C_2, respectively. Below, $R^j(\bar{x})$ denotes the rejecting classifier corresponding to the class C_j.

Finally and by analogy:

$$\begin{aligned}
TP_{RR} &= \sum_{k=1}^{n} \left(\sum_{i=k}^{n} \sum_{\bar{x} \in C_i} \left(\prod_{j=1}^{i-1} \delta_{j\Phi_\alpha(\bar{x})} \cdot \overline{\mathrm{sgn}}\left(R^j(\bar{x})\right) \right) \cdot \delta_{i\Phi_\alpha(\bar{x})} \cdot \mathrm{sgn}\left(R(\bar{x})\right) \right) \\
FP_{RR} &= \sum_{k=1}^{n} \left(\sum_{i=k}^{n} \sum_{\bar{x} \notin C_i} \left(\prod_{j=1}^{i-1} \delta_{j\Phi_\alpha(\bar{x})} \cdot \overline{\mathrm{sgn}}\left(R^j(\bar{x})\right) \right) \cdot \delta_{i\Phi_\alpha(\bar{x})} \cdot \mathrm{sgn}\left(R(\bar{x})\right) \right) \\
FN_{RR} &= FN_R - \sum_{i=1}^{n} \sum_{\bar{x} \in C_i} \sum_{k=i+1}^{n} \left(\prod_{j=i}^{k-1} \delta_{j\Phi_\alpha(\bar{x})} \cdot \overline{\mathrm{sgn}}\left(R^j(\bar{x})\right) \right) \cdot \delta_{k\Phi_\alpha(\bar{x})} \cdot \mathrm{sgn}\left(R(\bar{x})\right) \\
TN_{RR} &= TN_R - \sum_{\bar{x} \in C_0} \left(\sum_{i=2}^{n} \left(\prod_{j=1}^{i-1} \delta_{j\Phi_\alpha \bar{x}} \cdot \overline{\mathrm{sgn}}(R^j(\bar{x})) \right) \delta_{i\Phi_\alpha \bar{x}} \cdot \mathrm{sgn}(R(\bar{x})) \right)
\end{aligned} \tag{8}$$

Intuitively, quality of classification depends on the rate of correctly classified elements. In this case, it depends on the factors *TP* and *TN*, which should be maximized. On the other hand, minimization of factors *FP* and *FN* increases quality of classification. As it is seen in the above formulas, reclassification increases *TP* and decreases *FN* factors, what is desired. However, reclassification increases *FP* and decreases *TN* factors, what is undesirable. Therefore, quality of reclassification depends on significant overhead of desired factors over undesired ones. Fortunately, configurations presented in Fig. 4 naturally lead to such overhead. Moreover, experiments confirm this advantage of classification with rejection. Since this paper is focused on concepts and algorithms, we do not expand the experimental part of this study.

4 The Case Study

To show how the described models may work on real data we used *Wine Data Set* from the UCI Machine Learning Repository. These data are the results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The analysis determined the quantities of 13 constituents found in each of the three types of wines. The set contains 178 records.

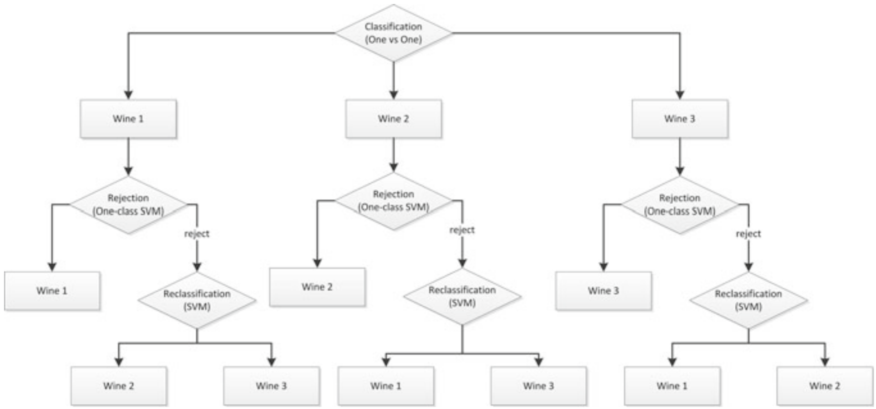


Fig. 5 The schema of the tested model

We randomly selected a half of records as the learning set. The rest of record was used in tests. In the test two kinds of SVM classifiers are used. The first kind are binary classifiers with the linear kernel and the cost C equals 1 [15]. The second kind are one-class SMV classifiers with the radial basis kernel and the following parameters: the kernel parameter $\gamma = 0.00001$, the cost for one-class classifier $\nu = 0.01$ [13]. In the whole system libsvm library was used [1].

The schema of the tested model is given in Fig. 5. The first layer of the model implements the classification without rejection model. The model was implemented as an ensemble of binary SVM classifiers. To solve the multi-class problem the One Against One strategy was used. The distribution of real classes (row) among recognized classes (column) is given below

$$c = \begin{pmatrix} 0.9655 & 0.0345 & 0.0000 \\ 0.0256 & 0.8974 & 0.0769 \\ 0.0000 & 0.0476 & 0.9524 \end{pmatrix}$$

After the classification, in the second layer of the tested model, for each class the one-class SVM classifier was created to reject foreign elements. The rejection accuracy for all classes is given below. On the diagonal the fraction of correct members not rejected by the one-class SVM classifier is given. The rest of cells contains the fraction of members from the class defined by the column rejected by the classifier created for the class given by the row.

$$r = \begin{pmatrix} 0.7931 & 0.9143 & 0.8333 \\ 0.7586 & 0.9714 & 0.0000 \\ 0.7931 & 0.3142 & 1.0000 \end{pmatrix}$$

Finally, in the last layer of the tested model, SVM binary classifiers were used to reclassify rejected elements. The obtained accuracy for all pairs of classes is given below.

$$rc = \begin{pmatrix} 0.0000 & 0.9846 & 1.0000 \\ 0.9846 & 0.0000 & 0.9322 \\ 1.0000 & 0.9322 & 0.0000 \end{pmatrix}$$

In the case of classification without rejection the obtained accuracy is 96, 90, and 95 % for the first, the second, and the third class respectively. To calculate the accuracy for the classification with rejection each element c_{ii} should be multiplied by the element r_{ii}. That gives 77, 87, and 95 % for the first, the second, and the third class respectively. For the third class the accuracy is the same as before. However, in the rest of cases we observe the reduction of the accuracy. In case of the first class the reduction is much bigger than the sum of misclassified observations from this class. Therefore it is useless to use the classification with rejection or the classification with reclassification of rejected elements in this case. For the second class the reduction is smaller than the sum of misclassified elements and the classification with reclassification of rejected elements may improve the accuracy.

To calculate benefits of the reclassification we have to estimate number of rejected members of foreign classes $c_{ij} * r_{ij}$ for each pair of different classes. After that we can estimate the number of correctly reclassified symbols. The number of correctly reclassified symbols is $c_{ij} * r_{ij} * cr_{jk}$ where $i \neq j$, $k \neq i$, and $k \neq j$.

In the discussed case, the accuracy for the third class was improved and it equals 96.76 % for the classification with reclassification of rejected elements. For the second class, the final accuracy is 89 % that is worse than the result of the classification without rejection. The result may be better, but the rejection method for the third class cannot reject any elements from the second class.

The presented discussion shows that the most important factor in the models is the rejection accuracy. This factor can decrease the classification results and can reduce the number of elements that may be reclassified.

5 Conclusions

In this paper, concepts and algorithms of classification with rejection of foreign elements are discussed. The discussion is based on three models of classification: classification without rejection, classifier with rejection, and classification with reclassification of rejected elements. All models of classification use hierarchical ensembles of binary classifiers. The study if focused on evaluation of quality of different classification models using cascade configuration of classification.

Evaluation of classification models shows how rejection and reclassification affects quality of classification. Comparing to standard classification, i.e., classification without foreign elements, rejection of foreign elements lowers the number of False Positives, but may also decrease the number of T(rue)P(ositives).

Reclassification can be used to reduce negative influence of rejections by reclassifying rejected elements. Such reclassification hunts up for F(alse)N(egatives), i.e., native elements incorrectly rejected, increasing quality of classification. Reclassification also may affect quality negatively. In this paper it is shown how to evaluate and compare quality of different models of classification. It is indicated that positive influence of rejection and classification overheads deterioration.

Future research and practical directions would include studies on different architectures of classifications based on binary classifiers, discussion on many classes classifiers, reinforcement of basic classification with rejection to design reclassification models.

Acknowledgments The research is supported by the National Science Center, grant No 2012/07/B/ST6/01501, decision no UMO-2012/07/B/ST6/01501.

References

1. Chang, C.-C., Lin, C.-J.: LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. **2**, 1–27 (2011)
2. Chow, C.: On optimum recognition error and reject tradeoff. IEEE Trans. Inf. Theory **16**(1), 41–46 (1970)
3. Fumera, G., Roli, F.: Analysis of error-reject trade-off in linearly combined multiple classifiers. Pattern Recogn. **37**, 1245–1265 (2004)
4. Graves, D., Pedrycz, W.: Kernel-based fuzzy clustering and fuzzy clustering: a comparative experimental study (2010)
5. Ha, T.M.: Optimum tradeoff between class-selective rejection error and average number of classes. Eng. Appl. Artif. Intell. **10**(6), 525–529 (1997)
6. Ishibuchi, H., Nakashima, T., Morisawa, T.: Voting in fuzzy rule-based systems for pattern classification problems. Fuzzy Sets Syst. **103**, 223–238 (1999)
7. Li, M., Sethi, I.K.: Confidence-based classifier design. Pattern Recogn. **39**(7), 1230–1240 (2006)
8. Lou, Z., Liu, K., Yang, J.Y., Suen, C.Y.: Rejection criteria and pairwise discrimination of handwritten numerals based on structural features. Pattern Anal. Appl. **2**(3), 228–238 (1999)
9. Luckner, M.: Comparison of hierarchical svm structures in letters recognition task. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Zurada, J. (eds.) Computational Intelligence: Methods and Applications, pp. 291–302. Challenging Problems of Science, Academic Publishing House EXIT, Warsaw (2008)
10. Mascarilla, L., Frélicot, C.: Reject strategies driven combination of pattern classifiers. Pattern Anal. Appl. **5**(2), 234–243 (2002)
11. Pierce, S.G., Worden, K., Manson, G.: Evaluation of neural network performance and generalisation using thresholding functions. Neural Comput. Appl. **16**, 109–124 (2007)
12. Pillai, I., Fumera, G., Roli, F.: A classification approach with a reject option for multi-label problems. In: Maino, G., Foresti, G.L. (eds.) ICIAP (1). Lecture Notes in Computer Science, vol. 6978, pp. 98–107. Springer (2011)
13. Schölkopf, B., et al.: Estimating the support of a high-dimensional distribution. Neural Comput. **13**(7), 1443–1471 (2001)
14. Simeone, P., Marrocco, C., Tortorella, F.: Design of reject rules for ECOC classification systems. Pattern Recogn. **45**(2), 863–875 (2012)
15. Vapnik, V.: Statistical Learning Theory. Wiley, New York (1998)
16. Xie, J., Qiu, Z., Wu, J.: Bootstrap methods for reject rules of fisher LDA. Pattern Recogn. 425–428 (2006)

Complementarity and Similarity of Complementary Structures in Spaces of Features and Concepts

Wladyslaw Homenda and Agnieszka Jastrzebska

Abstract The motivation for undertaking the research on relations, such as complementarity, is complexity of real-world phenomena. Authors aim at preparation of proper modeling tools that would enable to describe dependencies to the greatest extent. The objective of this article is to discuss relation of complementarity. Authors propose definition of complementarity for concepts described with features. Concepts represent phenomena, features describe concepts. The paper is oriented on research on similarity. Authors propose similarity relations for complementary structures in spaces of features and concepts.

Keywords Concepts · Features · Similarity · Complementarity

1 Introduction

Complexity of real-world phenomena makes it impossible to precisely model and predict processes like unemployment, market fluctuations, or consumer behavior. Plenty of contemporary models account only a fraction of influencing attributes, it is often impossible to identify them all. Moreover, it is often not doable to precisely express knowledge about attributes. As G. E. P. Box wrote: essentially, all models are wrong, but some are useful, [2].

The objective of this article is to discuss an approach to phenomena modeling based on their features. Concepts (phenomena) belong to the space of concepts. They are described by their features. Feature and concept spaces contain available knowledge and allow to characterize relations between features and relations between concepts. The nature of analyzed knowledge is imprecise. Features evaluations are fuzzified. In this paper, authors focus on positive information only.

W. Homenda · A. Jastrzebska (✉)
Faculty of Mathematics and Information Science,
Warsaw University of Technology, ul. Koszykowa 75, 00-662 Warsaw, Poland
e-mail: A.Jastrzebska@mini.pw.edu.pl
URL: http://www.mini.pw.edu.pl/~homenda

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_32

Of interest is structuring induced by the complementarity of features and similarity. Authors join these threads and discuss similarity of complementary features vectors. This paper is a revised and extended version of our former paper, [8], their objective is to define and discuss similarity measures for complementary structures.

Research on similarity of such structures is important from practical point of view. Similarity and dependencies like complementarity between real-world objects are of great interest in many areas of knowledge, including biology, computer vision, and economics. The inspiration for our research comes from observations of relations in the world of human beings. Hence, examples used in this paper are related to common life situations.

The article is structured as follows: in Sect. 2 feature and concept spaces are introduced. The distinction between the two of them is discussed. Similarity relation s_G allowing to compute similarity between features based on underlying concept space is presented. In Sect. 3 defined is complementarity. Section 4 covers constructed similarity relations.

2 Preliminaries

2.1 The Space of Features and the Space of Concepts

In many areas of science phenomena and relations between them are of greatest importance. Unfortunately, we are most likely unable to process phenomena directly. Therefore, we often describe them with their features and infer about their properties through comparison of features. In our study phenomena are called concepts and their descriptions are called features. We focus on characterization and structuring of the space of features and the space of concepts. In this paper, special interests are complementarity and similarity of complementary structures.

The space of features consists qualitatively of n features, which may be perceived also as labels or names.

$$\mathcal{F} = (f_1, \dots, f_n), \quad n \in \mathcal{N} \tag{1}$$

Features $f_1, \dots, f_n$ describe concepts. For example, if we consider features concerning a boy, we may be interested in features like f_{b1}—"brown hair," f_{b2}—"likes dogs," or f_{b3}—"plays basketball." Depending on application, we distinguish and analyze chosen subsets of all possible features. Also in this article, we operate on limited subset of feature space.

Particular features evaluations describe various possibilities. Features evaluations are scaled to the [0, 1] interval. Let us present the definition of the space of features $\mathcal{D}$, which is

$$\mathcal{D} = \{(\mu_1, \mu_2, \dots, \mu_n) : \mu_i \in [0, 1],\ i = 1, 2, \dots, n,\ n \in \mathcal{N}\} \tag{2}$$

Feature space consists of vectors of features evaluated with fuzzified positive information, expressed as numbers from the [0, 1] interval. Features vectors contain evaluations of corresponding attributes. Qualitative reference is $\mathcal{F}$. In the article, speaking about features, we mean features evaluations, not their labels.

An example, showing the distinction between the label and the evaluation is the following: assume we analyze a subset of $\mathcal{F}$ consisting of only one feature: f_{b1}—"brown hair". We may create three (or more) features vectors evaluations: $\mu_{Ab1} = [0]$, $\mu_{Bb1} = [0.7]$ and $\mu_{Cb1} = [1]$. These vectors correspond to evaluations of blond hair (μ_{Ab1}), dark red hair (μ_{Bb1}) and brown hair (μ_{Cb1}). Features evaluations vectors based on this example may be constructed so that they allow to represent different shades of hair colors.

Features are used to describe concrete concepts. By analogy, we define the space of concepts:

$$\mathcal{C} = \{c_1, c_2, \ldots, c_r\} \tag{3}$$

such that $r \in \mathcal{N}$ is the number of concepts in the space $\mathcal{C}$. Presented ideas are very general. Hence, our model may be adjusted to various areas of application.

Evaluated features may correspond to particular concepts. The scope of our study is to determine and characterize relations between features and between concepts through their features. A semantic mapping, which we call valuation mapping $\mathcal{V}$ generates subsets of the space of concepts satisfying certain features-related conditions. The valuation mapping assigns a set of concepts for every vector of features. We can formally describe such a mapping as

$$\mathcal{V} : \mathcal{D} \to \mathcal{P}(\mathcal{C}) \tag{4}$$

where $\mathcal{P}(\mathcal{C})$ is the family of all subsets of the space of concepts.

For instance, the simplest valuation assigns all concepts with a given vector of features $(\mu_1, \mu_2, \ldots, \mu_n)$ to this vector of features:

$$\mathcal{V}(\mu_1, \mu_2, \ldots, \mu_n) = \{c \in \mathcal{C} : d(c) = (\mu_1, \mu_2, \ldots, \mu_n)\} \tag{5}$$

where the mapping $d : \mathcal{C} \to \mathcal{D}$ defines features for every concept.

Valuation mapping $\mathcal{V}$ generates a subset of the space of concepts, such that each ith feature of selected concepts was evaluated exactly the same, as respective ith feature from analyzed input features vector $(\mu_1, \mu_2, \ldots, \mu_n)$. We may perceive mapping $\mathcal{V}$ as point-wise mapping, since it requires equality of concepts descriptions with pattern features.

We are interested in relations, that occur in subsets of the space of features and indirectly in the underlying concept space. The universe of discourse are concepts, but we want to process them through their descriptions. Approaches present in the literature are limited in this way. In further parts of this article, we present original methodology of the feature space structuralization.

First, we would like to highlight possible hierarchies in feature space. Hierarchy is induced through the generalization of chosen set of features. We are interested in hierarchies based on tree structures. For example:

- pills, ointments, and drops may be generalized to medicines (two-level tree),
- tigers and lions may be generalized to cat species (two-level tree),
- nurses and doctors may be generalized to health care staff (two-level tree),
- high school teachers and college teachers may be generalized to secondary-school teachers, teaching assistants and lecturers may be generalized to academic-level teachers, finally secondary-school teachers and academic-level teachers may be generalized to teachers (three-level tree).

Similar hierarchies are commonly present. Our idea is to use knowledge about hierarchical structure of features to characterize other dependencies in the space of features. Plain feature-wise comparison of features, that neglects data structures, omits extra knowledge, which we often have about certain attributes.

In our research, we focus on three types of structuring within the features space: exclusion, inclusion, and complementarity. In this paper, the great interest is complementarity and similarity of complementary features. We discuss it in Sects. 3 and 4. In this paper, we analyze simple two-level hierarchies. In a two-level tree, we have one root and one or more children. In future, we will investigate more complex structures.

2.2 Similarity in the Space of Features

A very important relation in the feature space is similarity. In the current subsection, basic concepts are introduced. We cover here briefly both literature based and our own approach to similarity modeling. There is substantial methodological difference between these two.

2.2.1 Similarity Based on Distance

Similarity in the feature space may be understood in a straightforward way. This approach utilizes distances between features vectors. Similarity then is a relation dual to the vectors' distance and is computed by comparison of each ith feature. The advantage of this approach is that there is a lot of defined distance measures, [1, 3, 9]. Even if we use another imprecise information representation model, including bipolar models, literature contains a lot of distance measures for intuitionistic fuzzy sets, type 2-fuzzy sets, [6, 7, 10] and so on.

Let us assume that we have two vectors of features evaluations:

$$\begin{aligned} \mu_A &= (\mu_{A1}, \mu_{A2}, \ldots, \mu_{An}) \\ \mu_B &= (\mu_{B1}, \mu_{B2}, \ldots, \mu_{Bn}) \end{aligned} \tag{6}$$

Then, we can apply Euclidean distance: $d_E(\mu_A, \mu_B) = \sqrt{\sum_{i=1}^{n}(\mu_{Ai} - \mu_{Bi})^2}$, Chebyshev distance: $d_{Ch}(\mu_A, \mu_B) = \max\left\{|\mu_{Ai} - \mu_{Bi}| \,:\, i = 1, 2, \ldots, n\right\}$, Manhattan distance: $d_M(\mu_A, \mu_B) = \sum_{i=1}^{n} |\mu_{Ai} - \mu_{Bi}|$ and many more. Distance measures satisfy axioms of non-negativity, symmetry, identity of indiscernibles and triangle inequality.

2.2.2 Similarity Based on the Underlying Concept Space

Here generalized similarity relation s_G of two vectors of features is presented.

The generalized similarity relation s_G of two vectors of features (μ_A and μ_B—see formula 6) is defined as follows:

$$s_G(\mu_A, \mu_B) = \frac{\left|\mathcal{V}(\mu_A) \cap \mathcal{V}(\mu_B)\right|}{\left|\mathcal{V}(\mu_A) \cap \mathcal{V}(\mu_B)\right| + |\mathcal{V}(\mu_A) \setminus \mathcal{V}(\mu_B)| + |\mathcal{V}(\mu_B) \setminus \mathcal{V}(\mu_A)|} \tag{7}$$

The key to the similarity measure s_G is the valuation mapping $\mathcal{V}$. Similarity measure s_G operates on features vectors, but the similarity value is in fact determined by the underlying space of concepts. The greater is the cardinality of the intersection of two sets extracted with the valuation mapping $\mathcal{V}$, the greater is the similarity. The greater is the number of concepts not shared by sets extracted with $\mathcal{V}(\mu_A)$ and $\mathcal{V}(\mu_B)$, the smaller is the similarity. Presented relation is inspired by the Tversky similarity ratio, [11]. Due to the space limitations we do not discuss here similarity measure s_G, but continue with the main topic, which is similarity between complementary structures of features.

3 Complementarity of Features

In this study interest is of features complementarity. It is a horizontal (or internal) relation occurring within nonempty set of features, a subset of the feature space.

Let us assume that we have sets of features F_1, F_2, …, F_m such that $F_1, F_2, \ldots, F_m \subset \mathcal{F}$ and they are nonempty and pairwise disjoint. We say that such sets of features are complementary to the threshold α if they satisfy the following condition:

$$\text{CompL}_\alpha(F_1, \ldots, F_m) \quad \Leftrightarrow \quad \text{s-aggr}\left(\,\text{aggr}(F_1), \text{aggr}(F_2), \ldots, \text{aggr}(F_m)\right) \geqslant \alpha \tag{8}$$

In this paper, complementarity relation is defined within features. In the formula above, s-aggr and aggr are aggregating operators with desired properties. Since we restrain the domain of interest to the [0, 1] interval, we treat aggregating operator as a mapping $\mathcal{A} : \bigcup_{n\in\mathcal{N}}[0, 1]^n \to [0, 1]$ such that:

1. $\mathcal{A}(x_1, \dots, x_n) \leqslant \mathcal{A}(y_1, \dots, y_n)$ whenever $x_i \leqslant y_i$ for all $i \in \{1, \dots, n\}$,
2. $\mathcal{A}(x) = x$ for all $x \in [0, 1]$,
3. $\mathcal{A}(0, \dots, 0) = 0$ and $\mathcal{A}(1, \dots, 1) = 1$
4. $\mathcal{A}(\dots, x_i, \dots, x_j, \dots) = \mathcal{A}(\dots, x_j, \dots, x_i \dots)$ whenever $0 \leqslant i, j \leqslant n$
5. $\mathcal{A}(x_1, \dots, x_n) \geqslant \max(x_1, \dots, x_n)$

Generally speaking, any aggregation operator $\mathcal{A}$ can be interpreted as a family of mappings $\{A_n : n \in \mathcal{N}\}$ such that:

$$A_n : [0, 1]^n \rightarrow [0, 1] \text{ and } A_n(x_1, \dots, x_n) = \mathcal{A}(x_1, \dots, x_n)$$

Of course, $A_1 \equiv id_{[0,1]}$.

In order to determine complementarity with CompL_α, aggr and s-aggr aggregation operators are used. aggr is any aggregation operator satisfying axioms 1–4 named above. An s-aggr aggregation operators satisfy axioms 1–5, i.e., the result of aggregation cannot be smaller than the greatest of the aggregated arguments.

Examples of aggr aggregation operators satisfying these properties are t-norms and t-conorms. Examples of s-aggr aggregating operators are t-conorms. Aggregating operators should be picked for each model individually, to adjust it to desired modeling purposes.

Complementarity can also be interpreted as a subset of the space $\mathcal{D}$ of features valuations with certain restrictions. Let us assume that

$$F_1 = \{f_{11}, f_{12}, \dots, f_{1k_1}\}, \ \dots, \ F_m = \{f_{m1}, f_{m2}, \dots, f_{mk_m}\}$$

Let us also assume that we have a vector $d = (\mu_1, \mu_2 \dots, \mu_n)$ of features evaluations and that $\mu_{11}, \dots, \mu_{1k_1}, \dots \mu_{mk_m}$ are different elements of this vector. Let us also denote:

$$\sigma_1 = \mathrm{aggr}(\mu_{11}, \dots, \mu_{1k_1}), \ \dots, \ \sigma_m = \mathrm{aggr}(\mu_{m1}, \dots, \mu_{mk_m})$$

We say that the vector $d = (\mu_1, \mu_2 \dots, \mu_n)$ of features evaluations satisfies the relation CompL of complementarity:

$$\mathrm{CompL}_\alpha(d) \quad \Leftrightarrow \quad \text{s-aggr}(\sigma_1, \dots, \sigma_m) \geqslant \alpha$$

Based on the above assumptions, the relation CompL of complementarity can be interpreted as follows:

$$\mathrm{CompL}(F_1, \ \dots, \ F_m) = \left\{ d \in \mathcal{D} : \mathrm{CompL}\big(d\big) \right\} \subset \mathcal{D} \tag{9}$$

Grouping features into more general sets $F_1, F_2, \dots, F_m$ introduces hierarchy to the data. As a result, we may perceive an aggregate F_1 as a representative of features $f_{11}, f_{12}, \dots, f_{1k_1}$. This is illustrated on Fig. 1.

Let us present an example of complementarity. In many countries, there are certain staffing requirements for nursing homes. Such places require doctors, nursing

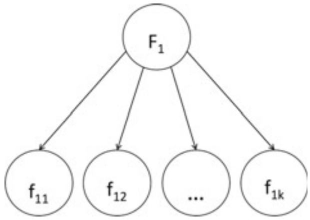


Fig. 1 Hierarchy in feature space

aides, and certified skilled nurses to be available 24 h a day. The number of employees with certain qualifications that have to be employed with a full-time position depends on the country, type of nursing home etc.

For the sake of this case study, let us assume that minimum staffing requirement for certain type and size of a nursing home is: 3 doctors, 10 skilled nurses, and 20 nursing aides. These numbers can be greater. Minimum staffing requirements just point out the critical (the lowest) level of personnel, which has to be maintained in order to run the nursing home.

Note that minimum staffing requirements for real-world organizations are often defined in a general fashion, recall the hierarchy and exemplar occupational groupings from Sect. 2. Regulations often group certain professions into classes. In this way, any kind of doctor (general practitioner, obstetrician, geriatric physician, etc.) can work at the nursing home and be included in the staffing minima.

With this example we discuss three distinct cases of complementarity relation:

- complementarity of nursing home employees on the doctor position,
- complementarity of nursing home employees on the certified skilled nurse position,
- complementarity of employed unlicensed assistive personnel (nursing aides positions).

Described complementarity relations are defined within the feature space (as is the complementarity definition in Formula 8). Complementarity with respect to the minimum staffing requirements for the nursing home may be satisfied by any organization, that fulfills conditions described above for the sake of simplicity we do not assume any other conditions.

In this case study concepts are nursing homes. Features that describe nursing homes are: offered health care services, available recreational facilities, employed staff, and enrolled residents. In this case study, we focus on the employed staff. We denote features labels as $f_{E1}, f_{E2}, \dots, f_{E6}$. Aggregated sets of features, which describe situation of interest are the following:

- F_{E1}: employed doctors—general practitioners,
- F_{E2}: employed doctors—geriatric physicians,
- F_{E3}: employed first-level nurses,
- F_{E4}: employed nurse practitioners,
- F_{E5}: employed nursing aides,
- F_{E6}: employed other unlicensed assistive personnel.

List above distinguish groups of features: F_{E1}, $F_{E2}, \dots, F_{E6}$. Each of named group may be seen as a generalization of more specific features describing health care personnel. Sets of features F_{E1}, $F_{E2}, \dots, F_6$ group all features such that:

- F_{E1}: contains features $f_{E11}, f_{E12}, \dots, f_{E1p}$, where each p-th feature corresponds to a person, who is a general practitioner and satisfies all requirements to be employed on a doctor position in the nursing home,
- F_{E2}: contains features $f_{E21}, f_{E22}, \dots, f_{E2s}$, where each s-th feature corresponds to a person, who is a geriatric physician and satisfies all requirements to be employed on a doctor position in the nursing home,
- F_{E3}: contains features $f_{E31}, f_{E32}, \dots, f_{E3t}$, where each t-th feature corresponds to a person, who is first-level nurse and satisfies all requirements to be employed in the nursing home,
- F_{E4}: contains features $f_{E41}, f_{E42}, \dots, f_{E4u}$, where each u-th feature corresponds to a person, who is a nurse practitioner and satisfies all requirements to be employed in the nursing home,
- F_{E5}: contains features $f_{E41}, f_{E42}, \dots, f_{E4v}$, where each u-th feature corresponds to a person, who is a nursing aid and satisfies all requirements to be employed in the nursing home,
- F_{E6}: contains features $f_{E51}, f_{E52}, \dots, f_{E5w}$, where each v-th feature corresponds to a person with is other unlicensed assistive personnel qualifications and who satisfies all requirements to be employed in the nursing home.

The number of features gathered in sets $F_{E1}, \dots, F_{E6}$ is very high. These sets contain all people qualified and legally allowed to work on given position.

Numerical representation of employment has to be scaled to the $[0, 1]$ interval and has to reflect modeled situation. Evaluations of features grouped in sets F_{E1}, $F_{E2}, \dots, F_{E6}$ are scaled with respect to the minimum staffing requirements for the nursing home. Evaluation has to be performed for features $f_{E11}, f_{E12}, \dots, f_{E1p}, \dots, f_{E61}, f_{E62}, \dots, f_{E6w}$. Evaluation will be separate for each of analyzed complementarity cases: doctors, certified nurses and uncertified caring staff.

First complementarity case: doctors. In this case in features evaluation the following numbers may be placed:

- in evaluation of features $f_{E11}, \dots, f_{E6w}$ if given person is qualified doctor and is employed as a doctor at the nursing home, which features vector we are evaluating, then on corresponding position we put $\frac{1}{3}$ otherwise 0.

Second complementarity case: certified nurses. In this case in features evaluation the following numbers may be placed:

- in evaluation of features $f_{E11}, \dots, f_{E6w}$ if given person is certified nurse (first-level or practitioner) and is employed on a nurse position in at the nursing home, which features vector we are evaluating, then on corresponding position we put $\frac{1}{10}$ otherwise 0.

Second complementarity case: masters. In this case in features evaluation the following numbers may be placed:

- in evaluation of features $f_{E11}, \ldots, f_{E6w}$ if given person qualifies as an unlicensed assistive personnel and is employed on an unlicensed assistive personnel position at the nursing home, which features vector we are evaluating, then on corresponding position we put $\frac{1}{20}$ otherwise 0.

3, 10, and 20 are numbers corresponding to minimum staffing requirement for doctors, certified nurses, and unlicensed assistive personnel. We may build many features evaluations vectors based on named features and evaluation scheme described above. Such features evaluation vectors will represent any hypothetical or real nursing home.

In analyzed case studies we propose the following aggregating operators and the threshold level α:

- bounded sum for aggr, i.e., a t-conorm defined as follows: $min\{a + b, 1\}$, where arguments are $a, b \in [0, 1]$,
- max for s-aggr,
- $\alpha = 1$.

Let us discuss an exemplar features evaluations vector E_1 describing certain nursing home for the first case example (complementarity of doctors):

$$E_1 = [\frac{1}{3}, \frac{1}{3}, 0, 0, \frac{1}{3}, 0, 0, 0, 0, 0, 0, 0, 0, 0, \frac{1}{3}, 0, \ldots, 0]$$

In the vector E_1, first 15 positions correspond to features $f_{E11}, f_{E12}, \ldots, f_{E115}$. Vector E_1 represents a nursing home, which is complementary with respect to the minimum staffing requirement for employees with doctor qualifications.

In an analogous way, we may construct features vectors representing minimum staffing requirement for certified and uncertified care staff.

In order to extract from the space of concepts (the space of all nursing homes) nursing homes which satisfy all staffing requirements to be operational, we have to find the intersection of three sets of nursing homes:

- nursing homes, that employ at least minimal number of doctors,
- nursing homes, that employ at least minimal number of certified nurses,
- nursing homes employing at least minimal number of unlicensed assistive personnel.

We may extend the case study and include also half-time employees.

Alternatively, if we may add certain additional restrictions to the complementarity relation, namely that aggregated are only these features, which are of higher than given β strength. In this way, we analyze complementarity only among these features, which were evaluated as greater than some level β and discard all features, which were evaluated as smaller than β. We may adjust the level of beta to the specific problem, that we want to model.

Definition of conditional complementarity looks as follows:

$$\text{CompLCond}_{\alpha}(F_{1c}, \dots, F_{mc}) \quad \Leftrightarrow \quad \text{s-aggr}\left(\text{aggr}(F_{1c}), \dots, \text{aggr}(F_{mc})\right) \geqslant \alpha \tag{10}$$

where:

$$F_{ic} = \left\{ f_{i1c}, f_{i2c}, \dots, f_{irc} : f_{i1c}, f_{i2c}, \dots, f_{irc} \geqslant \beta \right\}$$

A good example of conditional complementarity occurs in virtualized computer system environment. A system with virtualized resources has to balance performance with the ease of management. The more physical units are added to the system, the more costly and difficult is the administration. Therefore, adding a new physical unit with mediocre performance does not improve systems overall quality. Architectural consideration here is the balance between maintenance overhead and the performance. Systems are created with certain level of performance (greater than the desired level α), but they consist of the least possible number of physical equipment to achieve desired properties. Administrators do not add slow machines and require that added computing resources are more powerful than a necessary level β.

4 Similarity of Complementary Features

In our research, similarity of complementary features is very important. Due to space limitations, literature review on similarity measures has been shortened. Interesting research on similarity is presented not only, but also in: [1, 9, 11].

In this article discussed is similarity of vectors of complementary features. Complementarity occurs within both compared features vector. Vectors, which do not satisfy complementarity condition, are not analyzed in this paper.

Complementarity relation provides us with additional knowledge about information structure. Features are grouped in logical subsets $F_1, \dots, F_m$. We may use this knowledge and calculate similarity of such aggregates.

4.1 Similarity of Complementary Features Vectors Based on Distance

Let us start with the description of methodology for obtaining similarity measure between two vectors of complementary features evaluations. Similarity is analyzed here in isolation from the underlying concept space. We utilize distance measures, briefly mentioned in Sect. 2. Evaluations present in features vectors are compared directly. Of interest is metric distance between aggregated subsets of features evaluations. Aggregates are made according to logical division of feature space into $F_1, \dots, F_m$.

Let us assume that we have: complementary features sets $F_1, F_2, \ldots, F_m$ such that $F_1, F_2, \ldots, F_m \subset \mathcal{F}$ and two vectors of these features evaluations:

$$F_A = (F_{A1}, F_{A2}, \ldots, F_{Am})$$
$$F_B = (F_{B1}, F_{B2}, \ldots, F_{Bm})$$

Sets $F_1, F_2, \ldots, F_m$ consist of logically connected features, for example breakfast products, skiing equipment, etc. We may aggregate them, because of this additional knowledge, that features grouped in sets are somehow connected or replaceable. Naturally, we may analyze features grouped into one set, without any division. Features sets F_{A1}, F_{B1}, and so on are qualitatively corresponding.

In order to compute similarity of F_A and F_B, first we calculate Chebyshev distance between aggregated values of $F_{A1}, \ldots, F_{Am}$ and $F_{B1}, \ldots, F_{Bm}$:

$$d_{Ch}(F_A, F_B) = \max \left\{ |\mathrm{aggr}(F_{Ai}) - \mathrm{aggr}(F_{Bi})| \;:\; i = 1, \ldots, m \right\}$$

where aggr is a suitable aggregator. Authors suggest t-conorm. If features evaluations are restricted to [0, 1] and applied is a t-conorm aggregating function, then the distance measure above will always fall into the [0, 1] interval. In order to calculate similarity of F_A and F_B, we use the following formula:

$$s_{COMd}(F_A, F_B) = 1 - d_{Ch}(F_A, F_B) \tag{11}$$

Proposed similarity relation s_{COMd} is applicable in isolation from the underlying concept space. Relation between structures of complementary features is calculated even, if there are no objects, which may be described with given features. Instead of Chebyshev distance, different distance measure may be applied. Presented formulas take advantage of additional knowledge about features structure, namely its grouping into m subsets.

4.2 Similarity Based on the Valuation Mapping

Standard approaches to the similarity assume, that there is an underlying geometric representation of similarity. On this grounds similarity is treated rather superficially. Especially when we talk about comparing humans, there has to be a measure allowing to compare humans as entities characterized by some features, which distinguish us. Concepts, represented with their features not necessarily have to be in compliance with metric-based similarity measure. Strict axioms of distance-based approaches should be relaxed then. We have to use some generalizations of similarity measures, that will allow to model information such as judgments, attitudes, and so forth.

Therefore, we apply similarity relation s_G, which has been introduced in Sect. 2. It calculates coincidence between features vectors through numerical force of the underlying concept space.

Let us assume again that we have two complementary features sets: $F_A = (F_{A1}, F_{A2}, \ldots, F_{Am})$ and $F_B = (F_{B1}, F_{B2}, \ldots, F_{Bm})$. Subsets $F_{A1}, \ldots, F_{Am}, F_{B1}, \ldots, F_{Bm}$ consist of logically connected features. It is an additional knowledge, that we want to capture.

Similarity $s_{COM(plementary)}$ of two structures containing complementary features evaluations is calculated as follows:

$$s_{COM}(F_A, F_B) = s_G(Fa_A, Fa_B) \tag{12}$$

where:

$$\begin{aligned} Fa_A &= [\text{aggr}(F_{A1}), \text{aggr}(F_{A2}), \ldots, \text{aggr}(F_{Am})] \\ Fa_B &= [\text{aggr}(F_{B1}), \text{aggr}(F_{B2}), \ldots, \text{aggr}(F_{Bm})] \end{aligned}$$

Fa_A and Fa_B are vectors of aggregated values of the distinguished subsets of features. To calculate the similarity based on the underlying concept space, concepts representation (in the features vectors form) also has to be primarily aggregated in groups corresponding to sets $F_1, F_2, \ldots, F_m$. For the aggregation operator, aggr authors recommend t-conorm. s_G is a reflexive and symmetric similarity relation introduced in Sect. 2.2. Numerical properties of s_{COM} depend on applied aggregator.

Proposed similarity relation s_{COM} calculates similarity of complementary features structures through numerical power of the underlying concept space. As a result, we analyze concrete objects descriptions. We do not include vectors with features evaluations not connected to actual concepts. Additional knowledge about features, through prior features aggregation is also included.

5 Conclusions

This paper discusses structuralization in the feature and concept spaces. The nomenclature of formalized units of knowledge—space, which correspond to the real-world objects is introduced. Concepts are described by their features. Particular evaluations of features allow to describe and distinguish different objects. Feature evaluations are expressed as numbers from the $[0, 1]$ interval. Valuation mapping links space of descriptions with the space of concepts. Within such formal model, authors have developed similarity relation s_G, which allows to calculate similarity between two concepts.

In this study, the property of complementarity has been presented. Similarity of complementary structures was discussed. Two distinct methodologies were suggested. First, approach allows to compute similarity directly between features vectors. Second, methodology relies on the underlying concept space and utilizes similarity relation s_G and the valuation mapping $\mathcal{V}$. In the second approach objects' descriptions are compared indirectly, thorough the space of concepts.

In future research, authors plan to analyze different forms of structuring in the space of features. We will also focus on further generalization of our approach and on applying it with bipolar information representation models.

Acknowledgments The research is partially supported by the National Science Center, grant No 2011/01/B/ST6/06478.

References

1. Belanche, L., Orozco, J.: Things to know about a (dis)similarity measure. In: König, A., Dengel, A., Hinkelmann, K., Kise, K., Howlett, R.J., Jain, L.C. (eds.) Knowledge-Based and Intelligent Information and Engineering Systems. Lecture Notes in Computer Science, vol. 6881, pp. 100–109. Springer, Berlin (2011)
2. Box, G.E.P., Draper, N.R.: Empirical Model-Building and Response Surfaces. Wiley, New York (1987)
3. Goshtasby, A.A.: In: Singh, S., Kang, S.B. (eds.) Image Registration: Principles, Tools and Methods. Advances in Computer Vision and Pattern Recognition, pp. 7–66. Springer, London (2012)
4. Homenda, W.: Balanced fuzzy sets. Inf. Sci. **176**(17), 2467–2506 (2006)
5. Homenda, W., Jastrzebska, A.: Complementarity and similarity of complementary structures in spaces of features and concepts. In: Skulimowski, A.M.J. (ed.) 8th International Conference on Knowledge, Information and Creativity Support Systems, pp. 149–160. Progress & Business Publishers (2013)
6. Hung, W., Yang, M.: Similarity measures between type-2 fuzzy sets. Int. J. Uncertain. **12**(6), 827–841 (2004)
7. Hung, W., Yang, M.: Similarity measures of intuitionistic fuzzy sets based on Hausdorff distance. Pattern Recognit. Lett. **25**, 1603–1611 (2004)
8. Jastrzebska, A., Homenda, W.: Similarities in spaces of features and concepts: towards semantic evaluations. In: 2013 Federated Conference on Computer Science and Information Systems, pp. 257–260. IEEE Press (2013)
9. Klawonn, F., Kruse, R.: Similarity relations and independence concepts. In: Della Riccia, G., Dubois, D., Kruse, R., Lenz, H.J. (eds.) Preferences and Similarities, pp. 179–196. Springer, Wien (2008)
10. Szmidt, E., Kacprzyk, J.: Distances between intuitionistic fuzzy sets. Fuzzy Sets Syst. **114**, 505–518 (2000)
11. Tversky, A.: Features of similarity. Psychol. Rev. **84**(4), 327–352 (1977)

Modeling in Feature and Concept Spaces: Exclusion Relations and Similarities of Features Related with Exclusions

Agnieszka Jastrzebska and Wojciech Lesinski

Abstract The paper introduces definitions of exclusion relations in spaces of features and concepts. Concepts correspond to phenomena and they are described with their features. The objective of our research is to investigate and describe possible structuring and relations in the feature and concept spaces. In this article, three types of exclusions: weak, strict, and multiple qualitative are defined and discussed. The focus is on similarity of excluding structures. Two distinct methodologies are introduced. First approach relies on direct features comparison. Second methodology uses dedicated similarity relation rooted in the underlying concept space.

Keywords Concept space · Feature space · Similarity · Exclusion · Similarity of excluding features

1 Introduction

Formal models of knowledge and relations within the knowledge are one of major research streams in information sciences. The literature discusses various approaches to knowledge representation and structuralization, for example ontologies or cognitive maps. In this paper, authors present a model of concept and feature spaces. The aim is to investigate relations in feature and concept spaces. Among most interesting relations are similarity and exclusions, which are the key original topic of this article. The inspiration for his research is the world of consumers. Hence, examples are related to common life situations, for example purchasing choices. This paper is a revised and extended version of [6].

A. Jastrzebska (✉)
Faculty of Mathematics and Information Science, Warsaw University of Technology, ul. Koszykowa 75, 00-662 Warsaw, Poland
e-mail: A.Jastrzebska@mini.pw.edu.pl

W. Lesinski
Faculty of Mathematics and Computer Science, University of Bialystok, ul. Sosnowa 64, 15-887 Bialystok, Poland
e-mail: wlesinski@ii.uwb.edu.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_33

The paper is structured as follows. Section 2 covers a framework which formalizes feature and concept spaces. A vector-based representation model is also described. Valuation mapping, which allows to relate feature space with concepts, is introduced. Similarity relation customized for the developed model is presented. In Sect. 3, we address exclusion relations. Three distinct definitions of strict, weak and multiple qualitative exclusions are given. Section 4 discusses similarity of features satisfying weak or strict exclusion relations. Two distinct methodologies of calculating similarity for such features vectors are investigated.

2 Preliminaries

2.1 The Space of Concepts and the Space of Features

In this section, authors present concept and feature spaces.

The space of concepts

$$\mathcal{C} = \{c_1, c_2, \dots, c_r\} \quad r \in \mathcal{N} \tag{1}$$

is the universe of interest. Concepts are for example: human beings, molecules, organizations, computer systems, and so on. It is often impossible to process such entities directly, in their original form. Instead, it is a common practice to name and evaluate sets of features that describe entities. Features evaluation allows to describe and distinguish concepts. Concepts are comparable as long as they share some commonalities. Therefore, each concept from a given space of concept is described qualitatively by the same set of features. The differences between concepts are expressed with various features evaluations.

Features evaluations $\mathcal{D}$ are defined as

$$\mathcal{D} = \{(\mu_1, \mu_2, \dots, \mu_n) : \mu_i \in [0, 1]\} \quad i = 1, 2, \dots, n \quad n \in \mathcal{N} \tag{2}$$

Features evaluation vectors contain evaluations of corresponding features. In this study features evaluations are expressed as degrees of membership. Information representation scheme allows to express imprecise, positive gradual knowledge on the [0, 1] scale. The source of the knowledge may be for example an expert.

We can assume other information representation model, for example balanced fuzzy sets defined in [2] or Atanassov's intuitionistic fuzzy sets: [1]. Bipolar information representation models allow to illustrate more complex phenomena. Noteworthy are also asymmetric information processing schemes. Detailed discussion on bipolarity and asymmetry of data processing is in [3–5].

The space of features

$$\mathcal{F} = (f_1, f_2, \dots, f_n) \;\; n \in \mathcal{N} \tag{3}$$

contains names (or one may say labels) describing concepts. $\mathcal{F}$ is a qualitative reference. Numerical evaluations: $\mu_1, \mu_2, \ldots, \mu_n$ correspond to features named as: $f_1, \ldots, f_n$.

The following simple example illustrates the difference between concepts and features. Pupils from certain primary school constitute the space of concepts. One-element features vector represent each pupil. The only feature of interest is “likes singing”. One may construct a feature evaluation vector to describe a hypothetical student of a special interest. For example, such hypothetical feature evaluation vector is $\mu_P = [0.8]$ and it corresponds to a person, who potentially can be a member of a school choir.

2.2 The Valuation Mapping

Valuation mapping (V) generates a subset of the space of concept based on features evaluation vector given as an argument. It is a relation, that for every vector of features evaluations assigns a set of concepts:

$$V : \mathcal{D} \to 2^{\mathcal{C}} \tag{4}$$

Valuation mapping translates given features valuation vector into the underlying subset of existing concepts.

In this paper, the *filling-up valuation mapping* denoted as V_I is used

$$V_I(\mu_1, \ldots, \mu_n) = \{c \in \mathcal{C} : d(c) = (\mu_{c1}, \ldots, \mu_{cn}) \text{ and } \mu_{ci} \leqslant \mu_i\} \quad i = 1, \ldots, n \tag{5}$$

mapping $d : \mathcal{C} \to \mathcal{D}$ defines features for concepts. Mapping V_I generates a subset of the space of concept, in which each ith feature was not evaluated as greater than the corresponding feature in the features evaluation vector passed as an argument. Note that also different definitions of valuation mapping may be assumed.

V_I may be used to extract potential school choir members from the space of concepts (the primary school) in the toy case study introduced in the previous subsection.

2.3 Similarity of Features and Concepts

2.3.1 Brief Literature Review on Similarity

Similarity in the literature has been discussed in various contexts and applications. Noteworthy positions are not only but also [7–10]. In the literature, the concept of similarity is conventionally discussed in one of the three main streams

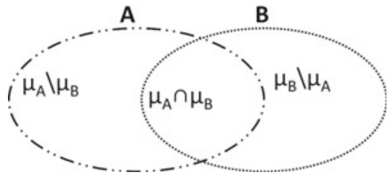


Fig. 1 Set theoretic approach to modeling similarity of **A** and **B**

- Distance-based similarity measures.
- Other measures of similarity, for example probabilistic-based approach to similarity, which includes correlation coefficients, f-divergence, and other.
- Set-theoretic similarity measures. Examples: Dice coefficient, Jaccard index, and Tversky index.

For this research most relevant is the set-theoretic approach. Set theoretic approach to similarity modeling was symbolically illustrated by Tversky with Fig. 1, [11].

In the set-theoretic approach similarity between two objects: A and B is obtained by analysis of their descriptions. Descriptions are sets of attributes determining objects' shapes (μ_A is A's attributes' set and μ_B is B's attributes' set). Similarity of A and B is determined with common features. Existence of features is binary (feature either exists or not). The more A and B are similar, the more overlapping are ellipses in Fig. 1. If objects are indistinguishable, ellipses lie on each other. Objects that do not share common features are disjoint. This very intuitive definition of similarity can be generalized onto many domains and transferred into plenty different similarity measures.

Exemplar set-theoretic similarity measures are: Jaccard index, Sorensen/Dice coefficient, and Tversky index. Similarity is a vast topic. Due to space limitations, the literature overview has been shortened.

2.3.2 Similarity in the Space of Features and Concepts

Similarity relation s_G, which is the main course of this subsection is inspired by the Tversky ratio. For details on Tversky ratio see [11]. Similarity relation s_G operates in spaces of concepts and features. s_G uses the valuation mapping linking both spaces. Relation s_G takes as arguments two features valuation vectors

$$\mu_A = (\mu_{A1}, \mu_{A2}, \dots, \mu_{An}), \quad \mu_B = (\mu_{B1}, \mu_{B2}, \dots, \mu_{Bn}) \tag{6}$$

and computes similarity between them. Similarity value depends on the cardinality of underlying concept subspaces, generated with the valuation mapping V_I:

$$s_G(\mu_A, \mu_B) = \frac{\left|V_I(\mu_A) \cap V_I(\mu_B)\right|}{\left|V_I(\mu_A) \cap V(\mu_B)\right| + \left|V_I(\mu_A) \setminus V(\mu_B)\right| + \left|V_I(\mu_B) \setminus V_I(\mu_A)\right|} \tag{7}$$

Similarity relation s_G determines the similarity of two features evaluation vectors based on the underlying concept space. Concept space has to be nonempty. The codomain of s_G is $[0, 1]$. The more numerous the intersection of two sets generated with valuation mapping V_I for vectors μ_A and μ_B, the greater the similarity. The greater the number of not shared concepts, the smaller the similarity.

Relation s_G is applied in the later parts of this paper, in Sect. 4, where the study on similarity of excluding features is presented.

3 Relation of Exclusion

This section introduces strict, weak and multiple qualitative relations of exclusion. Exclusion is a type of dependency, which we observe among features. It is an internal property of certain features evaluation vectors. Exclusion is qualitatively defined for features names and quantitatively satisfied in features evaluation vectors. Research on exclusion relations is original contribution of this article.

Let us discuss features vector F. Vector F contains names (labels) of n features: $F = [f_1, \dots, f_n]$. Corresponding features evaluation vector is denoted as: $\mu = [\mu_1, \dots, \mu_n]$.

We consider a set of n features gathered according to some logical key. Examples of logical (qualitative) distinction of features are the following: food (pork, ribs, cereals, cucumber, apple), clothing (trousers, skirts, suits), water sports (swimming, windsurfing, water-skiing), entertainment preferences (books, movies, opera) and so on. Moreover, groups of features have often nested internal groups. For example, food may be divided into vegetarian (cereals, cucumber, apple) and non-vegetarian (pork, ribs) products. Described property is a logical hierarchy of features.

Logical aggregation may involve many features. Let us denote sets of grouped features as $F_{G1}, \dots, F_{Gk}$. $F_{G1} \cup \dots \cup F_{Gk} = F$. Sets $F_{G1}, \dots, F_{Gk}$ split the set F to nonempty and pairwise disjoint subsets. Corresponding features evaluation vectors are denoted as: $\mu_{G1}, \dots, \mu_{Gk}$.

Exclusion relation occurs with respect to a set of features. A set of features excludes other features. Exclusion is a property, which can be satisfied in certain features evaluations vectors. For example, let us analyze a situation, when F_{G1} excludes $F \setminus F_{G1} = F_{G2} \cup \dots \cup F_{Gk} = \overline{F_{G1}}$. First, exclusion relation is defined within features names. Second, we check, which features evaluation vectors satisfy it.

We can depict exclusion relation with a labeled tree. In this paper, we discuss basic two-level structures. An example is in Fig. 2. It is based on features vector F (names) and its split into k nonempty subsets. The structure representing this relation is a two level tree. On the first level features from F_{G1} are placed. F_{G1} is a set of features, which excludes all other features from F. Features from set F_{G1} are aggregated into a single root node. Root's children are all other aggregated subsets of features ($F_{G2}, \dots, F_{Gk}$), which are excluded by the root.

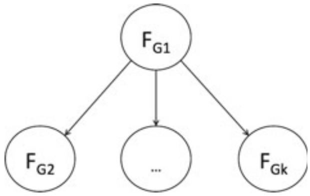


Fig. 2 F_{G1} excludes $F_{G2}, \ldots, F_{Gk}$

3.1 Strict Exclusion Relation

This study aims at distinguishing various types of exclusion relation. The distinction is important from applicational point of view. Exclusion is a relation commonly observed in the real world. Exclusion in this article is an internal, horizontal property of features evaluation vector. We have stressed in the previous subsection, that exclusion is defined qualitatively for features names. In the second step, it is necessary to check, if features evaluation vectors satisfy the exclusion relation.

First, we present strict exclusion. Let us assume features vector F, its nonempty subset of selected features of interest F_{Gi} ($F_{Gi} \in F$), and their corresponding features evaluation vectors denoted as μ and μ_{Gi}.

Definition 1 Strict horizontal exclusion within features evaluation vector μ with respect to features μ_{Gi} occurs when $aggr(\mu_{Gi}) \geqslant \alpha$ and $aggr(\mu \setminus \mu_{Gi}) = 0$, where $1 \geqslant \alpha \geqslant 0$.

Remark 1 In a special case, if strict exclusion is with respect to a single feature μ_i, the following conditions are satisfied: $\mu_i \geqslant \alpha$ and $\mu_1 = \ldots = \mu_{i-1} = \mu_{i+1} = \ldots = \mu_n = 0$.

Remark 2 Here, aggregating function *aggr* is a t-conorm. The choice of aggregating function depends on assumed information representation model.

In strict exclusion, if there is a nonzero evaluation of any feature from the excluding set F_{Gi}, then each feature from $\overline{F_{Gi}}$ has to be evaluated as 0.

For $\alpha = 0$ relation of exclusion is valid for a vector of 0's. Strict exclusion defined as above is antireflexive, symmetrical, and nontransitive.

Illnesses are very intuitive examples of strict exclusion. People with certain health problems are strictly prohibited to consume specific products. Illnesses strictly exclude certain foods. In the example below concepts are people, features are their illness and consumed foods.

Let us discuss galactosemia (feature denoted as μ_{GAL}), which is a genetic metabolic disorder that affects an individual's ability to metabolize the sugar galactose. In individuals with this disease, the enzymes needed for galactose metabolism are significantly diminished or missing entirely. The only treatment for galactosemia is strict elimination of lactose and galactose from the diet. Examples of food containing lactose and galactose: milk, honey, plums. These three products are features denoted as:

μ_m, μ_h, μ_p. A vector with chosen features is the following: $\mu_{Ex1} = [\mu_{GAL}, \mu_m, \mu_h, \mu_p]$. Next, based on this example, we may construct many features evaluation vectors. If there is nonzero evaluation of feature μ_{GAL}, it means that the corresponding person suffers from galactosemia. If there is a nonzero evaluation of feature μ_m, μ_h or μ_p, it means that the person consumes milk, honey or plums.

Presence of nonzero value in μ_{GAL} excludes μ_m, μ_h and μ_p. Each person sick with galactosemia has to have all features: μ_m, μ_h and μ_p evaluated as 0. The nature of exclusion is strict: μ_{GAL} rigorously prohibits possibility of appearance of nonzero values in other features. Analogously, presence of any nonzero evaluation in μ_m, μ_h or μ_p strictly excludes occurrence of galactosemia.

3.2 Weak Exclusion Relation

In this section weak exclusion relation is discussed. As before, weak exclusion relation is defined horizontally, within features F. In features evaluation vector μ a condition for weak exclusion is checked.

Definition 2 Weak horizontal exclusion within features valuation vector $\mu = [\mu_1, \dots, \mu_n]$ occurs when $aggr(\mu) \leqslant \alpha$, where $1 \geqslant \alpha \geqslant 0$.

Remark 3 As before, $aggr$ is a t-conorm. The choice of aggregating function depends on assumed information representation model.

In the weak exclusion relation, we do not distinguish any features subset. It is a property of a whole features evaluation vector. Weak exclusion relation is satisfied in each features evaluation vector filled with 0's.

Let us give an example of a weak exclusion within features vector describing on-budget selection of ski sets at a given ski shop. Concept space contains all customers at given ski shop. The following assumptions are made:

- Consumer is in a small ski shop to purchase a ski set.
- Consumer is bounded by the budget—he can spend no more than 1000 € on ski equipment.
- The ski shop has a (limited) offer of various ski products at different prices.
- Consumer picks available products.
- Cost of each product in the ski shop is divided per 1000, in order to scale this case study to the [0, 1] interval.

Features are 20 products available in this ski shop: six different skis ($Fs1, \dots, Fs6$), two different ski bindings ($Fs7, Fs8$), six different ski boots ($Fs9, \dots, Fs14$), two ski poles sets ($Fs15, Fs16$), one helmet ($Fs17$), one ski goggles ($Fs18$), and two different ski suits ($Fs19, Fs20$). General form of features evaluation vector is following: $\mu_{Ex2} = [\mu_{Fs1}, \mu_{Fs2}, \dots, \mu_{Fs20}]$, where $\mu_{Fs1}, \dots, \mu_{Fs20}$ correspond to products available in this shop. If a consumer chooses given product, in the corresponding position in his features evaluation vector a nonzero evaluation is placed. The values correspond to item's price divided per 1000.

For aggregation operator *aggr* arithmetic sum is assumed. For α 1 is assumed. Consumer can pick as many products as he can, as long as the total cost does not exceed his budget—sum cannot be greater than the threshold α. Ski products $\mu_{S1}, \dots, \mu_{T2}$ weakly exclude each other. Naturally, the consumer can make economic choices and save some money.

In each case study operators and threshold level have to be customized. This case is very simple on purpose. Our goal is to illustrate the idea of weak exclusion in the clearest way.

3.3 Multiple Horizontal Qualitative Exclusion Relations

The third exclusion relation discussed in this paper is multiple horizontal exclusion. It is a relation decomposable on the qualitative level into weak or strict exclusions. In this subsection, the description begins with an example of relation's appearance.

Let us discuss features vector: $\mu_{Ex3} = [\mu_{athome}, \mu_{atfriends}, \mu_{out}]$. Named features correspond to consumer's preferences towards the form of evening entertainment. Concepts are consumers. Consumer has to decide, which form of entertainment he prefers. A person cannot be in two places at the same time, so the choices are exclusive: consumer either stays at home, or goes to his friends, or goes out. Concurrent mutual exclusion of features occurs. Multiple horizontal qualitative exclusion is observed. Qualitatively speaking, in μ_{Ex3} there is are three different exclusion relations:

- staying at home excludes being at friend's and going out,
- being at friend's place excludes staying home and going out,
- going out excludes staying home and being at friend's.

In contrast to previous exclusion relations, here is multiple mutual dependency among features names (on the qualitative level). Example μ_{Ex3} and its decomposition into three exclusion relations is illustrated in Fig. 3.

Particular evaluations of features vectors with suspected multiple exclusion relation cannot be ambiguous. Each evaluation of such vector has to be in an indecomposable weak or strict exclusion relation.

In the example above features can be treated as sets of features. For example, feature μ_{athome} is a generalization of all leisure activities, that one can do while being at home, for example: reading a book, napping, etc. In a general case, multiple exclusion occurs among m mutually excluding sets of features and it can be decomposed into m exclusion relations (weak or strict, depending on the phenomena of interest).

In further parts of this paper, only features vectors satisfying indecomposable weak or strict exclusion relations are discussed.

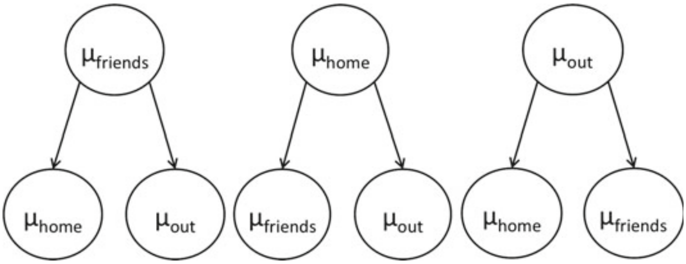


Fig. 3 Multiple qualitative exclusion of $\mu_{Ex3} = [\mu_{athome}, \mu_{atfriends}, \mu_{out}]$ decomposed into three exclusion relations

4 Similarity and Exclusion

Of main interest is similarity in feature and concept spaces. This section aims at defining similarity relations for features evaluation vectors satisfying indecomposable weak or strict exclusion relations. Two distinct methodologies for calculating similarity of such structures are analyzed. First, direct distance-based method is discussed. Second, an indirect approach based on similarity relation s_G is proposed.

4.1 Similarity Based on Direct Features Vectors Comparison

In this subsection, similarity of two features evaluation vectors in the relation of strict exclusion, as in Definition 1, is investigated. Let us compare two such structures: μ_A and μ_B

$$\mu_A = (\mu_{A1}, \mu_{A2}, \ldots, \mu_{An}), \quad \mu_B = (\mu_{B1}, \mu_{B2}, \ldots, \mu_{Bn}) \tag{8}$$

Each features evaluation vector contains n features. It is assumed that features are on corresponding positions. It means that μ_{A1} is qualitatively the same feature as μ_{B1}.

4.1.1 Strict Exclusion Relation

In the strict exclusion relation, nonzero evaluations appear only among features from excluding set, denoted as F_{Gi}. Set of excluding features F_{Gi} has to be nonempty. All other features are evaluated as 0.

It is possible to represent any features set, for example set F_{Gi}, with an aggregated value. For aggregation a t-conorm is applied. Aggregation operators are chosen with respect to assumed information representation model.

In this context, the property of *excluding features matching* is introduced.

Remark 4 Let us assume two features vectors: μ_A and μ_B both satisfying the relation of strict exclusion. Excluding features matching in vectors μ_A and μ_B occurs, if a set of excluding features from μ_A corresponds to the set of excluding features in μ_B.

In other words, excluding features matching occurs, if nonzero evaluations are expected at the same positions in both μ_A and μ_B. Example of matching: vector 1 describes exclusion of lactose and galactose in patient A sick with galactosemia, vector 2 describes exclusion of lactose and galactose in patient B sick with galactosemia. Example of non matching vectors: vector 1 describes exclusion of lactose and galactose in patientson A sick with galactosemia, vector 3 describes exclusion of windsurfing for a person, who does not know how to swim.

Sets of excluding features from μ_A and μ_B are denoted as F_{AGi} and F_{BGi} respectively. If sets of excluding features are not matching, similarity of μ_A and μ_B is arbitrarily set to 0. This is very intuitive property. If strict exclusion concerns qualitatively different features, then there is no similarity between such structures. "Strictness" of the strict exclusion relation is expressed by assuming similarity equal to 0, whenever excluding features are not matching.

In the case, if matching of excluding features occurs, distance between aggregated excluding features from μ_A and μ_B is computed. We use Chebyshev distance (d_{Ch})

$$d_{Ch}(\mu_A, \mu_B) = \max\left\{|\mu_{Aj} - \mu_{Bj}| \,:\, j = 1, 2, \ldots, n\right\} \tag{9}$$

Similarity s_{XDM} (X—exclusion, D—distance, M—matching) between two features evaluation vectors with matching excluding features is calculated:

$$s_{XDM}(\mu_A, \mu_B) = 1 - d_{Ch}(aggr(F_{AGi}), aggr(F_{BGi})), \tag{10}$$

where $aggr(F_{AGi})$ and $aggr(F_{BGi})$ are aggregated excluding features. *aggr* is a t-conorm, for example max. Chebyshev distance (see Formula 9) is used to calculate dissimilarity between matched aggregated values. Distance is computed between two numbers.

To sum up, in the case of strict exclusion relation, similarity s_{XD} (X—exclusion, D—distance) of two features evaluation vectors is calculated as follows:

$$s_{XD}(\mu_A, \mu_B) = \begin{cases} 0 & \text{excluding features } \underline{\text{not}} \text{ matching} \\ s_{XDM}(\mu_A, \mu_B) & \text{excluding features matching} \end{cases} \tag{11}$$

s_{XD}'s codomain is $[0, 1]$. Only if the same set of features is excluding other possibilities in both structures, there is certain similarity. In the case, if strict exclusion relation is with respect to not matching features, similarity is equal 0.

4.1.2 Weak Exclusion Relation

In the case of weak exclusion relation, the procedure for obtaining distance-based similarity is analogical. Let us again assume two features vectors μ_A and μ_B, which satisfy weak exclusion relation as in Definition 2. To compute distance-based similarity between two such features evaluation vectors following formula is applied:

$$s_{XDw}(\mu_A, \mu_B) = 1 - d_{Ch}(\mu_A, \mu_B), \tag{12}$$

where $d_{Ch}(\mu_A, \mu_B)$ is Chebyshev distance between two n-element vectors (for d_{Ch} see Formula 9). Codomain of s_{XDw} is $[0, 1]$. For perfectly similar vectors s_{XDw} is equal 1. The strength of similarity of two features evaluation vectors depends on the differences between corresponding evaluations in μ_A and μ_B.

Similarity measure based on distance does not require any information about the underlying concept space. It is more generic, because it requires only features evaluations. In the next subsection an alternative approach is presented.

4.2 Similarity of Excluding Features Based on Concept Space

The methodology, which is discussed in this subsection, relies not only on features vectors, but also on the underlying concept space. It uses similarity relation s_G and the valuation mapping V_I introduced in Sect. 2.

4.2.1 Strict Exclusion Relation

Let us assume two features evaluation vectors: μ_A and μ_B, both satisfying strict exclusion relation. Vectors μ_A and μ_B contain evaluations of features from the set F. Set F is divided logically into subsets of features: $F_{G1}, \ldots, F_{Gk}$. To compute similarity s_{XC} (X—exclusion, C—concept space) of two features evaluation vectors (μ_A and μ_B), firstly logical subsets of features are aggregated. As a result, two features aggregates vectors are obtained:

$$\begin{aligned} \mu_{Aag} &= \big(aggr(F_{AG1}), aggr(F_{AG2}), \ldots, aggr(F_{AGk})\big) \\ \mu_{Bag} &= \big(aggr(F_{BG1}), aggr(F_{BG2}), \ldots, aggr(F_{BGk})\big) \end{aligned} \tag{13}$$

where $aggr(F_{AG1}), \ldots, aggr(F_{AGk}), aggr(F_{BG1}), \ldots, aggr(F_{BGk})$ are aggregated with a t-conorm evaluations of features from $F_{G1}, \ldots, F_{Gk}$ from μ_A and μ_B.

It was assumed, that both μ_A and μ_B are in strict exclusion relation. Therefore, in vectors μ_{Aag} and μ_{Bag} nonzero evaluation can occur in only one position.

Consequently, operation of features sets aggregation with the same operator *aggr* has to be performed in the space of concepts. Finally, s_G is used to calculate similarity of μ_{Ax} and μ_{Bx} based on the concept space of aggregated values:

$$s_{XC}(\mu_A, \mu_B) = s_G(\mu_{Aag}, \mu_{Bag}) \tag{14}$$

If the exclusion relation is strict and excluding features do not match, then similarity is always equal to 0, because intersection $V_I(\mu_{Aag}) \cap V_I(\mu_{Bag})$ is empty. The same property was maintained for the relation based on distance.

4.2.2 Weak Exclusion Relation

If given features evaluation vectors μ_A and μ_B satisfy weak exclusion relation, similarity between them is computed with s_G without any prior modifications

$$s_{XCw}(\mu_A, \mu_B) = s_G(\mu_A, \mu_B) \tag{15}$$

Results obtained with similarity relation s_G rely on the content of the underlying concept space. Cardinalities of generated subsets of the concept space determine the strength of similarity of two features vectors μ_A and μ_B. Methodology based on relation s_G is compliant with the developed model of features and concept spaces.

5 Conclusions

The article is devoted to relation of exclusion within features evaluation vectors. Three distinct definitions of exclusion: strict, weak, and multiple qualitative were introduced. Differences between them were illustrated with case examples. Of main interest is similarity of features evaluation vectors satisfying weak or strict exclusion relation. In the paper research on two different approaches was presented: distance-based and concept space-based. To take full advantage of dependencies in the concept space, the second proposed methodology should be used. It calculates similarity of concepts' descriptions through the underlying concept space. The key is the valuation mapping, which relates features with concepts. An alternative and very flexible way of obtaining similarity of two vectors of features evaluations satisfying the relation of weak or strict exclusion is to calculate dependency directly with a distance-based measure.

Among possible applications of the proposed modeling framework are expert systems, most importantly decision aid systems. We believe that the approach on which we are working can be beneficial and improve efficient detection and description of relevant relations within data. The existing approaches to similarity modeling focus on an elementary level of abstraction. Similarity is typically considered as a relation for sets or objects. The proposed framework introduces structuring of concepts and therefore it operates on a higher level of abstraction.

In future, authors plan to continue their research on relations and structures in the feature and concept spaces. We are currently investigating relations of complementarity and inclusions. Our objective is to propose a comprehensive modeling framework to describe complex phenomena, where such relations play crucial role. The ultimate goal is to operate on such level of abstraction that we would be able to describe relations within related (structured) concepts. The proposed methods offer original perspective on knowledge mining. Among other, more technical topics that we will investigate are other information representation models that can be implemented in this framework.

Acknowledgments A. Jastrzebska contribution is supported by the Foundation for Polish Science under International PhD Projects in Intelligent Computing. Project financed from The European Union within the Innovative Economy Operational Programme (2007–2013) and European Regional Development Fund. W. Lesinski's research is supported by the National Science Center, grant No 2011/01/B/ST6/06478, decision no DEC-2011/01/B/ST6/06478.

References

1. Atanassov, K.T.: Intuitionistic fuzzy sets. Fuzzy Sets Syst. **20**, 87–96 (1986)
2. Homenda, W.: Balanced fuzzy sets. Inf. Sci. **176**(17), 2467–2506 (2006)
3. Homenda, W., Pedrycz, W.: Balanced fuzzy computing unit. Int. J. Uncertain. Fuzziness Knowl. Based Syst. **13**(2), 117–138 (2005)
4. Homenda, W., Pedrycz, W.: Balanced fuzzy gates. In: Greco, S., Hata, Y., Hirano, S., Inuiguchi, M., Miyamoto, S., Nguyen, H.S., Słowiński, R. (eds.) RSCTC 2006. LNCS (LNAI), vol. 4259, pp. 107–116. Springer, Berlin (2006)
5. Homenda, W., Pedrycz, W.: Symmetrization of fuzzy operators: notes on data aggregation. In: Halgamuge, S.K., Wang, L. (eds.) Computational Intelligence for Modelling and Prediction. Studies in Computational Intelligence, vol. 2, pp. 1–17. Springer, Berlin (2005)
6. Jastrzebska, A.: Similarity of exclusions in the concept space. In: Skulimowski, A.M.J. (ed.) 8th International Conference on Knowledge, Information and Creativity Support Systems, pp. 245–256. Progress & Business Publishers (2013)
7. Julian-Iranzo, P.: A procedure for the construction of a similarity relation. In: Magdalena, L., Ojeda-Aciego, M., Verdegay, J.L. (eds.) 12th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, pp. 489–496 (2008)
8. Klawonn, F., Kruse, R.: Similarity relations and independence concepts. In: Della Riccia, G., Dubois, D., Kruse, R., Lenz, H.J. (eds.) Preferences and Similarities, pp. 179–196. Springer, Wien (2008)
9. Lin, D.: An Information-theoretic definition of similarity. In: Shavlik, J.V. (ed.) 15th International Conference on Machine Learning, pp. 296–304. Morgan Kaufmann (1998)
10. Orozco, J., Belanche, L.: On aggregation operators of transitive similarity and dissimilarity relations. In: 13th IEEE International Conference on Fuzzy Systems, pp. 1373–1377. IEEE Press (2004)
11. Tversky, A.: Features of similarity. Psychol. Rev. **84**(4), 327–352 (1977)

E-Unification of Feature Structures

Petr Homola

Abstract Unlike structural unification, *E*-unification of feature structures has, to the best of our knowledge, never been used in natural language processing (NLP). We formalize the concept of *E*-unification for feature structures, present a universal *E*-unification procedure and discuss its computational tractability for arbitrary as well as linguistically motivated *E*-theories. A number of examples illustrate the usefulness of *E*-unification in the domain of NLP.

Keywords Feature structures · *E*-unification · Abstract rewriting systems · Knowledge representation and reasoning

1 Introduction

The use of *E*-unification in feature structure-based grammar formalisms is investigated in this paper. The reported results are an outgrowth of research on syntactically formed complex predicates [1, 2] and their linguistically motivated formal description and treatment without the use of the so-called restriction operator [9, 10, 17].

The notion of feature as a formal means of linguistic description originated in the tradition of the Prague school of linguistics in the interbellum [14]. The methods described in this paper can be used in any feature structure-based formalism such as Lexical-Functional Grammar [5, 7, 8] or Categorial Unification Grammar [15]. Most of the formalisms utilize a context-free or categorial grammar processed by a chart parser as a backbone. The use of charts and unification in natural language processing goes back to [6]. Kay [11, 12] was probably the first to use feature structures and the operation of unification on them in a formal grammar which is an analogue to

P. Homola (✉)
Codesign, s.r.o., Prague, Czech Republic
e-mail: phomola@codesign.cz

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_34

term unification in first-order logic. E-unification [3, 4] comes into play if we define equality (modulo a theory E) on feature structures. An example from the domain of NLP is presented in the next section.

2 Motivation

Informally, (structural) unification only adds information to feature structures. However, sometimes it seems necessary to "merge" two feature structures in a way that is incompatible with structural unification. A well investigated case are syntactically formed causatives in Romance languages [1, 2]. A Catalan example is given in (1).

$$\begin{array}{llll} \textit{El} & \textit{vaig} & \textit{fer} & \textit{riure} \\ \text{him} & \text{go.1}\textsc{sg} & \text{do.}\textsc{inf} & \text{laugh.}\textsc{inf} \end{array}$$
$$\text{"I made him laugh."} \tag{1}$$

The corresponding lexico-semantic value of the complex predicate is given in (2).

$$\begin{bmatrix} \text{FUNC} & \text{'cause'} \\ \text{ARG1} & \textsc{subj} \\ \text{ARG2} & \begin{bmatrix} \text{FUNC} & \text{'riure'} \\ \text{ARG1} & \textsc{obj} \end{bmatrix} \end{bmatrix} \tag{2}$$

Note that there is no way to obtain (2) from the feature structures of the two verbs in (1) as formed in the lexicon (given in (3)) by means of structural unification.

$$\begin{bmatrix} \text{FUNC} & \text{'cause'} \\ \text{ARG1} & \textsc{subj} \\ \text{ARG2} & f \end{bmatrix} \begin{bmatrix} \text{FUNC} & \text{'riure'} \\ \text{ARG1} & \textsc{subj} \end{bmatrix} \tag{3}$$

The operation that "merges" the two values is called "composition" by [1]. His "composition" operation ($=_H$) which uses the restriction operator [9, 10] can be implemented by means of E-unification of feature structures based on the following E-theory:

$$E = \left\{ \begin{bmatrix} \text{FUNC} & \text{'cause'} \\ \text{ARG1} & \textsc{subj} \\ \text{ARG2} & \begin{bmatrix} \text{FUNC} & \rule{2em}{0.4pt} \\ \text{ARG1} & \textsc{obj} \end{bmatrix} \end{bmatrix} \approx \begin{bmatrix} \text{FUNC} & \\ \text{ARG1} & \textsc{subj} \end{bmatrix} \right\} \tag{4}$$

E states that, for example, *He laughs* and *I made him laugh* are equal modulo the feature of causation. If formalized, an algorithm can be used that "merges" two feature structures according to a linguistically motivated declaratively described E-theory. In the next sections, we lay out a formal framework of E-unification of feature structures and discuss its computational properties.

3 *E*-Theories in Commutative Idempotent Monoids

Before a discussion of feature structures, we describe a more abstract case in algebraic terms and present a more general version of the corresponding algorithms regarding *E*-theories. How feature structures fit into the presently described algebraic structure is explained in Sect. 5.

Let $(M, \cdot)$ be a commutative idempotent monoid. We define a partial ordering on M as follows:

Definition 1 $a \preccurlyeq b \equiv a \cdot b = b$ for all $a, b \in M$

We assume that the semilattice $(M, \preccurlyeq)$ is atomistic and distributive. It follows from the distributivity (by the Kurosh-Ore theorem) that every non-atom can be expressed as a product of atoms which is unique up to reordering and redundancy.

Definition 2 An equational theory $\doteq_E$ over a set of equational axioms (identities) over M (i.e., $E = \{a_1 \approx b_1, \ldots, a_n \approx b_n\}$) is the least binary relation with the following properties:

1. $E \subseteq \doteq_E$,
2. $a \doteq_E a$ for all $a \in M$,
3. if $a \doteq_E b$ then $b \doteq_E a$ for all $a, b \in M$,
4. if $a \doteq_E b$ and $b \doteq_E c$ then $a \doteq_E c$ for all $a, b, c \in M$,
5. if $a \doteq_E b$ then $a \cdot c \doteq_E b \cdot c$ for all $a, b, c \in M$.

A thus defined relation is a congruence on $(M, \cdot)$. We write $E \vdash a \approx b$ if $a \doteq_E b$ can be derived from E by subsequent application of the following rules:

Reflexivity

$$\frac{\top}{\rho \approx \rho}$$

Symmetry

$$\frac{\rho \approx \sigma}{\sigma \approx \rho}$$

Transitivity

$$\frac{\rho \approx \sigma \wedge \sigma \approx \xi}{\rho \approx \xi}$$

Specialization

$$\frac{\rho \approx \sigma}{\rho\xi \approx \sigma\xi}$$

The specialization rule can be construed as an analogue of substitution.

A model of E is a set U together with an interpretation function $\mathcal{I} : M \to U$ such that

$$a \doteq_E b \equiv \mathcal{I}(a) = \mathcal{I}(b)$$

and we write $E \vDash a \approx b$ if $a \approx b$ is valid in all models of E. The four inference rules allow for enumerating equivalence classes modulo an equational theory E. The logic is sound and complete (we omit the proofs for lack of space).

Theorem 1 (Soundness) *$E \vdash a \approx b$ implies $E \vDash a \approx b$ for all E, a, b.*

Theorem 2 (Completeness) *$E \vDash a \approx b$ implies $E \vdash a \approx b$ for all E, a, b.*

Note that in conjunction with a function $I : M \to \{\mathbf{0}, \mathbf{1}\}$ the operation $\cdot$ on the monoid can be construed as unification: We say that $a, b \in M$ are unifiable if $I(a \cdot b) = \mathbf{1}$. Informally, I can be understood as a predicate that asserts that an element of M is "(semantically) interpretable" or "meaningful". In Sect. 5, we will formalize I as satisfiability of formulae in Wedekind's [16] attribute-value logic. E-unification is defined as follows:

Definition 3 Given a commutative idempotent monoid $(M, \cdot)$, a function $I : M \to \{\mathbf{0}, \mathbf{1}\}$ (such that $a \preccurlyeq b \wedge I(b) = \mathbf{1} \to I(a) = \mathbf{1}$) and an equational theory $\doteq_E$, we say that $a, b \in M$ are E-unifiable if there exist $c, d \in M$ such that $a \preccurlyeq c, b \preccurlyeq d, I(c) = \mathbf{1}, I(d) = \mathbf{1}$ and $c \doteq_E d$. The pair $\langle c, d \rangle$ is the E-unifier of a and b.

For our purposes in the domain of NLP, we are interested in a universal procedure that for $a, b \in M$ finds all c such that $a \doteq_E c$ and $b \preccurlyeq c$. Using the aforementioned derivation rules, we can enumerate all c such that $a \doteq_E c$ holds and check whether $b \preccurlyeq c$. In the next section, we will discuss an algorithm that enumerates all such c for E-theories provided an ordering on M can be found.

4 Computationally Tractable *E*-Unification via Rewriting

While the universal E-unification procedure is sound and complete, it is in general very inefficient and yields redundant results. In this section, we discuss techniques that allow for an efficient E-unification algorithm for a subclass of E-theories. The terminology in what follows as well as a lemma are taken from [3] (for this reason we omit the proofs).

Definition 4 A rewriting system R is terminating if there is no infinite chain $a_1 \to_R a_2 \to_R \cdots$.

Definition 5 b is a normal form of a if $a \overset{*}{\to}_R b$ and there is no c such that $b \to_R c$. A rewriting system R is called normalizing if every a has a normal form.

Definition 6 A rewriting system R is called locally confluent if for every a, b, c such that $a \to_R b, a \to_R c$ there exists a d and $b \overset{*}{\to}_R d, c \overset{*}{\to}_R d$.

Definition 7 A rewriting system R is called confluent if for every a, b, c such that $a \stackrel{*}{\rightarrow}_R b, a \stackrel{*}{\rightarrow}_R c$ there exists a d and $b \stackrel{*}{\rightarrow}_R d, c \stackrel{*}{\rightarrow}_R d$.

Lemma 1 (Newman's lemma) *A terminating rewriting system is confluent iff it is locally confluent.*

E-equivalence on M can be interpreted as a confluent rewriting system if an ordering $<$ defined on M can be found such that $a < b \rightarrow a \cdot c < b \cdot c$ for all $a, b, c \in M$ (i.e. $<$ is monotone) and the Knuth-Bendix completion procedure [13] succeeds. In such a case, we obtain a rewriting system R and $\doteq_R$ (defined by $a \doteq_R b \equiv a \stackrel{*}{\leftrightarrow}_R b$) is equivalent to $\doteq_E$.

Note that if there is an oriented rewriting system $\rightarrow_R$ equivalent to $\doteq_E$ that is not confluent, we can check E-equivalence of a and b by computing all normal forms $a \stackrel{*}{\rightarrow}_R a'$ and $b \stackrel{*}{\rightarrow}_R b'$ and checking whether $a' = b'$.

5 Formal Representation of Feature Structures

Wedekind's [16] logical representation of feature structures (attribute-value logic, henceforth AVL) is a special case of the monoid described in Sect. 3 which lets us use the presented apparatus to process f-structures obtained from an LFG [5, 8] parser.

In AVL, we have a set of constants $\mathcal{C}$ and a set of unary function symbols $\mathcal{F}_1$ ($\mathcal{C} \cap \mathcal{F}_1 = \emptyset$). The class of terms $\mathcal{T}$ contains all constants and if τ is a term and f a function symbol, $f\tau$ is also a term. The atomic formulae of AVL have the form $\tau_1 = \tau_2$ where $\tau_1, \tau_2 \in \mathcal{T}$ or $\bot$.[1] The formulae are formed recursively by means of the logical connectives $\neg$ and $\vee$.

Equivalence classes of certain AVL formulae can be seen as a commutative idempotent monoid. The atoms of the monoid are atomic formulae of the form $\sigma a = b$ where $\sigma \in \mathcal{F}_1^+$ and a and b are constants. The unit element is $\top$ and the operation $\cdot$ is $\wedge$. Subsumption is defined by $\varphi \sqsubseteq \psi \equiv \psi \rightarrow \varphi$. It is easy to see that a thus defined monoid is commutative and idempotent (further it is atomistic and distributive). The operator I can be formally defined on formulae as follows:

$$I(\varphi) = \begin{cases} \mathbf{1} & \text{if } \varphi \text{ is satisfiable} \\ \mathbf{0} & \text{otherwise} \end{cases}$$

Thus, $I(\varphi \wedge \psi) = \mathbf{1}$ if the feature structures expressed by φ and ψ are unifiable. Since AVL is decidable, I is computable for all formulae.

[1] We use the symbol $=$ instead of $\approx$ for identity in AVL as the latter symbol is already in use for equational axioms.

The equational axioms (the elements of E) have the form

$$\sigma_1 x \wedge \cdots \wedge \sigma_n x \approx \tau_1 x \wedge \cdots \wedge \tau_m x$$

where $\sigma_i, \tau_i \in \mathcal{F}_1^+$ and $x \in \mathcal{C}$ (i.e., x is a constant). Sometimes it may be useful to allow x to be any term, i.e., ηy where $\eta \in \mathcal{F}_1^*$ and $y \in \mathcal{C}$ (which gives us the possibility of replacing a substructure in a feature structure). It is not difficult to check that even with thus defined equational axioms, the relation $\doteq_E$ is a congruence and we can use the universal procedure provided previously to enumerate E-unifiers. The definition of $\doteq_E$ is extended as follows:

$$\xi \doteq_E \chi \text{ implies } \varphi \doteq_E \varphi[\chi]_v$$

where $\xi = \varphi|_v$ (i.e., ξ is a substructure of φ at position v that gets replaced by ξ). The corresponding inference rule for feature structures (represented as directed acyclic graphs) is

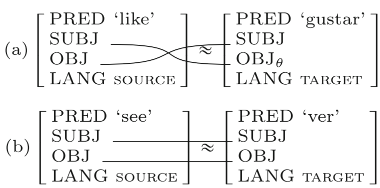
$$\frac{\xi \approx \chi}{\varphi \approx \varphi[\chi]_v}$$

where $\xi = \varphi|_v$. In the next section, we present two practical examples of a thus defined E-theory that can be implemented as a rewriting system.

6 Examples

6.1 Machine Translation as E-Unification

Future structures capture the linguistic meaning of sentences. They originate in the lexicon and grow in size during parsing. Rules in transfer-based machine translation (MT) systems can be described as E-theories that operate on feature structures. If each feature structure contains the attribute LANG with a value from {SOURCE,TARGET}, $\doteq_E$ can be converted to a rewriting system $\rightarrow_R$ since we can define $<$ by [LANG TARGET, …] $<$ [LANG SOURCE, …]. For example, the axiom given in (5a) "translates" *X like(s) Y* into Spanish *X*-DAT *gusta(n) Y*-NOM.



(5)

Note that the subject of the English sentence maps to the indirect object of the Spanish sentence and the direct object maps to the subject. This kind of "structural misalignment" is very frequent across natural languages. An example of an axiom without structural misalignment is given in (5b). Here the English subject and object map directly onto their Spanish counterparts.

Rewriting systems equal to *E*-theories for transfer-based MT systems are not confluent in general (but they are normalizing). A possible solution is to use a weighted rewriting system with weighted rewriting rules that assign a score to normal forms. The normal form with the highest weight is considered the best result. In conjunction with a depth-first search strategy and provided the rules are ordered according to their weights, the normal form with the highest weight can be obtained using a deterministic algorithm.

An open question is how the corresponding *E*-axioms (or rewrite rules) could be obtained in an unsupervised way. Even though axioms for lexical transfer can be obtained comparatively easily from bilingual dictionaries or corpora, hand-written axioms for structural transfer require deep knowledge of the grammar of both the source and target language. An automatic or at least semiautomatic creation of *E*-axioms is highly desirable. A detailed discussion cannot be given here due to lack of space, nevertheless let us mention that syntactically annotated parallel corpora would be extremely useful. Our initial experiments with a parallel syntactically annotated English-Aymara corpus reveal that the approach sketched above is viable.

6.2 Syntactically Formed Complex Predicates

We now come back to syntactically formed complex predicates described in Sect. 2. Consider the following Catalan sentence with an intransitive verb that governs a subject and a causee:

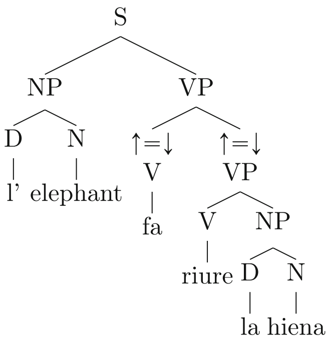
L'	*elefant*	*fa*	*riure*	*la*	*hiena*
the	elephant-MASC	make-PRES,3SG	laugh-INF	the-FEM	hyena

"The elephant makes the hyena laugh." (6)

The feature structures for individual words come from the lexicon, such as (7) for the verb *riure*. The slots for subject and object are empty as their values will be assigned later in the course of parsing.

$$\begin{bmatrix} \text{PRED} & \text{`riure'} \\ \text{SUBJ} & \\ \text{OBJ} & \\ \text{VFORM} & \textsc{infinitive} \end{bmatrix} \tag{7}$$

As illustrated in tree (8), the light verb and the main verb are coheads, that is, *E*-unification based on a theory that subsumes axiom (4) is needed to formally derive a feature structure that yields 'cause$\langle(\uparrow \text{SUBJ}), \text{riure}\langle(\uparrow \text{OBJ})\rangle\rangle$'.



(8)

As can be seen, the process of building up the syntax tree is monotonic. Intuitively, the corresponding feature structure incrementally absorbs information until the whole sentence is covered.

7 Conclusions

We showed how E-unification can be used in the domain of NLP. We have presented a universal E-unification procedure for arbitrary E-theories and an efficient E-unification algorithm for theories that can be converted into a rewriting system. Several linguistically motivated examples have been discussed. Since a declarative notation can be utilized, E-theories can be used to describe linguistic processes and constructions such as syntactically formed causatives in Romance languages or transfer rules in machine translation.

References

1. Alsina, A.: A theory of complex predicates: evidence from causatives in bantu and romance. In: Alsina, A., Bresnan, J., Sells, P. (eds.) Complex Predicates, pp. 203–246 (1997)
2. Alsina, A., Bresnan, J., Sells, P.: Complex predicates: structure and theory. In: Alsina, A., Bresnan, J., Sells, P. (eds.) Complex Predicates, pp. 1–12 (1997)
3. Baader, F., Nipkow, T.: Term Rewriting and All That. Cambridge University Press (1998)
4. Baader, F., Snyder, W.: Unification theory. In: Robinson, A., Voronkov, A. (eds.) Handbook of Automated Reasoning, pp. 439–526 (2001)
5. Bresnan, J.: Lexical-Functional Syntax. Blackwell Textbooks in Linguistics, New York (2001)
6. Colmerauer, A.: Les systèmes Q ou un formalisme pour analyser et synthétiser des phrases sur ordinateur. Technical Report, Mimeo, Montréal (1969)
7. Dalrymple, M.: Lexical Functional Grammar, Syntax and Semantics, vol. 34. Academic Press (2001)
8. Kaplan, R.M., Bresnan, J.: Lexical-functional grammar: A formal system for grammatical representation. In: Bresnan, J. (ed.) Mental Representation of Grammatical Relations. MIT Press, Cambridge (1982)
9. Kaplan, R.M., Wedekind, J.: Restriction and Structure Misalignment, ms, Xerox Palo Alto Research Center (1991)

10. Kaplan, R.M., Wedekind, J.: Restriction and correspondence-based translation. In: Proceedings of the 6th EACL Conference, pp. 193–202 (1993)
11. Kay, M.: Functional Grammar. In: Proceedings of the 5th meeting of the Berkeley Linguistics Society (1979)
12. Kay, M.: Functional Unification Grammar: a formalism for machine translation. In: Proceedings of the 10th International Conference on Computational Linguistics and 22nd annual meeting on Association for Computational Linguistics (1984)
13. Knuth, D., Bendix, P.: Simple Word Problems in Universal Algebras, pp. 263–297 (1970)
14. Rounds, W.C.: Feature Logics. In: van Benthem, J.F., ter Meulen, A., (eds.) Handbook of Logic and Language. Elsevier (1996)
15. Uszkoreit, H.: Categorial Unification Grammars. In: Proceedings of COLING (1986)
16. Wedekind, J.: Classical logics for attribute-value languages. In: Proceedings of the 5th EACL Conference, pp. 204–209 (1991)
17. Wedekind, J., Kaplan, R.M.: Type-driven semantic interpretation of f-structures. In: Proceedings of the 6th Conference of the European Chapter of the Association for Computational Linguistics (1993)

A Clustering-Based Approach to Reduce Feature Redundancy

Renato Cordeiro de Amorim and Boris Mirkin

Abstract Research effort has recently focused on designing feature weighting clustering algorithms. These algorithms automatically calculate the weight of each feature, representing their degree of relevance, in a data set. However, since most of these evaluate one feature at a time they may have difficulties to cluster data sets containing features with similar information. If a group of features contain the same relevant information, these clustering algorithms set high weights to each feature in this group, instead of removing some because of their redundant nature. This paper introduces an unsupervised feature selection method that can be used in the data pre-processing step to reduce the number of redundant features in a data set. This method clusters similar features together and then selects a subset of representative features for each cluster. This selection is based on the maximum information compression index between each feature and its respective cluster centroid. We present an empirical validation for our method by comparing it with a popular unsupervised feature selection on three EEG data sets. We find that our method selects features that produce better cluster recovery, without the need for an extra user-defined parameter.

Keywords Unsupervised feature selection · Feature weighting · Redundant features · Clustering · Mental task separation

R.C. de Amorim (✉)
School of Computer Science, College Lane Campus, University of Hertfordshire,
Hatfield AL10 9AB, UK
e-mail: r.amorim@herts.ac.uk

B. Mirkin
Department of Computer Science and Information Systems,
Birkbeck University of London, Malet Street, London WC1E 7HX, UK
e-mail: mirkin@dcs.bbk.ac.uk

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_35

1 Introduction

Reducing the cardinality of a data set, without losing information, has been a constant object of research effort for a considerable amount of time [19, 29]. The general aim of feature selection is to reduce such cardinality by removing any feature in a data set that are not of interest. For instance, those that are redundant containing the same information, those which are composed of noise, or simply those that have no relevance to the particular task being performed. There are indeed various benefits that may originate from the use of a feature selection algorithm, for instance (i) the amount an algorithm, being this in classification or clustering, takes to process a data set tends to decrease as the cardinality of a data set decreases; (ii) it may reduce the probability of a model describing an error or noise present in a given data set, instead of the underlying data structure, a problem commonly known as overfitting!; (iii) it may provide a general improvement to an algorithm used afterwards, being this in terms of accuracy or cluster recovery.

Feature selection either selects or deselects a particular feature $v \in V$ of a data set Y. It does so by providing each feature $v \in V$ with a weight of either one or zero. The generalization of this method, in which each feature $v \in V$, receives a weight in the interval $[0, 1]$ is known as feature weighting. Clearly, the weight w_v of a particular feature v aims to reflect the degree of relevance of v to the particular task at hand. The use of feature weighting does not conflict with feature selection as the former still allows a feature v to be deselected by setting $w_v = 0$. The major difference between feature selection and feature weighting is that the latter recognizes that even among relevant features that may be different degrees of relevance.

Feature weighting algorithms have been applied to hierarchical, as well as partitional clustering [2, 4, 11, 14–16, 34]. Hierarchical clustering algorithms aim to produce a set of clusters which have a tree-like relationship, demonstrable through a dendrogram, between them. These algorithms allow a given entity $y_i \in Y$ to be assigned to more than one cluster, as long as these clusters are related and the assignment happens at different levels of the tree. Partitional clustering algorithms take a different approach, originally they produce disjoint clusters, allowing a given entity $y_i \in Y$ to be assigned to a single cluster. Exceptions to this rule were introduced with Fuzzy C-Means [9] which allows a given entity to belong to all clusters with a degree of membership $u_{ik} \in [0, 1]$.

Here we expand our previous work [5] with a particular interest in improving Weighted K-Means (WK-Means) [11], and its L_p-based generalization, the intelligent Minkowski Weighted K-Means (iMWK-Means) [4] in terms of cluster recovery. Both algorithms apply cluster-based feature weight, allowing a feature v to have different weights at different clusters $k = \{1, 2, \ldots, K\}$, where K is the total number of clusters. This weight w_{kv} is calculated following the intuitive assumption that if a feature v has a high relative dispersion in a particular cluster S_k then its degree of relevance, and by consequence its weight w_{kv}, should be low. More details related to both algorithms can be found in Sect. 2.

The WK-Means, the iMWK-Means have been successfully applied in various scenarios [3, 4, 11, 20, 21]. However, these algorithms do introduce a new drawback. They both set w_{kv} by evaluating one feature $v \in V$ in each cluster $k = \{1, 2, \ldots, K\}$ at a time. Therefore, should a subset of features in V contain the same relevant information, none will be excluded by receiving a weight of zero. Quite the contrary, since they will all have similar small dispersions, each of their weights will be equally high.

In this paper we introduce a clustering-based method for feature selection that aims to reduce the number of redundant features in V. Our method, the intelligent K-Means for Feature Selection (iKFS), can be used to address the problem described above. The iKFS algorithm is used to cluster $v \in V$, rather than $y_i \in Y$. The number of clusters in V is found by using an anomalous pattern approach based on intelligent K-Means [28], as well as the maximum compression index (MIC) [29].

We evaluate the use of iKFS as a pre-processing step for both WK-Means and the iMWK-Means by clustering three data sets containing Electroencephalography (EEG) signals. These are high-dimensional data sets with 5,680 features each, with patterns that are difficult to discern. For comparison we run similar experiments using the features selected using the popular feature selection using feature similarity (FSFS) [29]. We find that iKFS tends to select a smaller amount of features that are in fact more relevant than those selected by FSFS, with the added benefit that iKFS does not require an extra user-defined parameter.

2 Background

Redundant features are those that contain similar information, by consequence when clustering a data set most such features are unnecessary. In order to reduce the number of redundant features in a data set our proposed method, iKFS, first aims to find clusters containing similar features. Clustering is the non-trivial task of creating K groups of entities so that those within the same group are similar and those between groups are dissimilar. Clustering algorithms have been used to solve problems in various fields of research, such as data mining, computer vision, bioinformatics, text mining, etc. [22, 28, 31, 36].

K-Means [8, 23] is among the most popular clustering algorithms. It performs partitional clustering, dividing a data set Y into K disjoint clusters $S = \{S_1, S_2, \ldots, S_K\}$. K-Means represents a given cluster S_k by its centre of gravity, the centroid c_k. Each feature $v \in V$ of a centroid, represented by c_{kv}, is equivalent to the average of v over each entity in S_k, $c_{kv} = \frac{\sum_{y_i \in S_k} y_{iv}}{|S_k|}$, assuming Euclidean distance. K-Means iteratively minimizes the sum of the distances between each entity $y_i \in Y$ and its respective centroid.

$$W(S, C) = \sum_{k=1}^{K} \sum_{y_i \in S_k} \sum_{v \in V} (y_{iv} - c_{kv})^2, \tag{1}$$

where $C = \{c_1, c_2, \ldots, c_K\}$. There are a number of reasons for the popularity of K-Means, among these, it is easy to implement, relatively fast algorithm, which is also intuitively easy to understand. In fact, the minimization of the K-Means criterion (1) has only the three steps below.

1. Assign the values of K random entities from Y to the initial centroids $c_1, c_2, \ldots, c_K$;
2. Assign each entity $y_i \in Y$ to the cluster represented by its closest centroid;
3. Update each centroid to the centre of gravity of its cluster, and go back to Step 2. Iterations cease when the algorithm converges.

The complexity of K-Means is of $\mathcal{O}(nKt)$, where t is the number of iterations K-Means takes to converge, and n the number of entities in Y. Although it is difficult to determine the value of t beforehand, we have shown that this tends to be small, particularly when K-Means is initialized with relevant, rather than random, centroids [1]. Other clustering algorithms can be much slower, for instance hierarchical algorithms have a complexity of at least $\mathcal{O}(n^2)$. Implementations of K-Means can be frequently found in software packages, such as MATLAB, R, SPSS, etc.

Although popular, K-Means does have drawbacks. Some of which have been target of research effort for a long time. For instance, K-Means requires K (the number of clusters in Y) to be known beforehand, and the clustering produced by K-Means can be heavily affected by the initial centroids used in its first step [10, 12, 24, 30, 33, 35].

Among the many algorithms addressing these two interrelated issues, intelligent K-Means (iK-Means) seems quite successful [1, 12, 28]. This algorithm finds the clusters in a data set by extracting one anomalous pattern at a time, as per below.

1. Set c_c, the centre of the data set Y.
2. Set a tentative centroid c_t, the entity $y_i \in Y$ that is the farthest from c_c.
3. Run K-Means on Y, using c_c and c_t as initial centroids. Do not allow c_c to move during the clustering.
4. If the $|S_{c_t}| \geq \theta$, add c_t to C_{init}, otherwise discard c_t. In any case, remove the entities in S_{c_t} from Y.
5. If there are entities in Y, go to Step 2.
6. Run K-Means, initialized with the centroids in C_{init} and $K = |C_{init}|$.

Recently, research effort has focused on the fact that K-Means treats all features $vinV$ equally, instead of taking into account that different features may have different degrees of relevance [17, 25]. Weighted K-Means (WK-Means) and the intelligent Minkowski Weighted K-Means (iMWK-Means) are of particular interest to us, because of their excellent ability to recover clusters [3, 4, 11, 20, 21].

The main difference between WK-Means and iMWK-Means is the distance measure in use. While the former applies the squared Euclidean distance, the latter makes a generalization of this by using the pth root of the Minkowski (L_p) distance. Both algorithms add a cluster-dependent weight, w_{kv} to the distance in use. To avoid linearity, this weight is put to the power of an user-defined exponent. The distance measure between an entity $y_i \in Y$ and a centroid $c_k \in C$ in WK-Means is then $d(y_i, c_k) = \sum_{v \in V} w_{kv}^{\beta} |y_{iv} - c_{kv}|^2$. In iMWK-Means, the distance exponent is the same as the weight exponent, making it possible to interpret w_{kv} as a feature re-scaling factor, for any exponent. The distance used by iMWK-Means is as follows.

$$d_p(y_i, c_k) = \sum_{v \in V} w_{kv}^p |y_{iv} - c_{kv}|^p. \tag{2}$$

Substituting the distance in the K-Means criterion (1) by the adjusted weighted distance (2) we obtain the iMWK-Means criterion.

$$W_p(S, C, w) = \sum_{k=1}^{K} \sum_{y_i \in S_K} \sum_{v \in V} w_{kv}^p |y_{iv} - c_{kv}|^p. \tag{3}$$

The exponent p is a user-defined parameter that affects equally the distance and the weights in (3). The calculation of w_{kv} for $k = 1, 2, \dots, K$ and each $v \in V$ follows the intuitive idea that if a feature v has a smaller relative dispersion in cluster S_k than a different feature $u \in V$, then v should have a higher weight in S_k than u. We formalize this with the equation below.

$$w_{kv} = \frac{1}{\sum_{u \in V} [D_{kvp}/D_{kup}]^{1/(p-1)}}, \tag{4}$$

where the dispersion of v at cluster S_k and specific p is given by $D_{kvp} = \sum_{y_i \in S_k} |y_{iv} - c_{kv}|^p$. For a given cluster S_k, the weights are subject to $\sum_{v \in V} w_{kv} = 1$, and a crisp clustering, in which an entity $y_i \in Y$ can only be assigned to a single cluster S_k. The equations for WK-Means are similar to all the above, but with a distance exponent always equal to two. We formalize the iMWK-Means algorithm below.

1. Obtain the initial centroids $C = \{c_1, c_2, \dots, c_K\}$ by applying the iK-Means algorithm, using the distance in Eq. 2. Set each cluster $S_k \leftarrow \emptyset$.
2. Assign each entity $y_i \in Y$ to the cluster S_k represented by the closest c_k, using (2). Should there be no change in S, stop.
3. Update each centroid in C to the Minkowski centre of their respective clusters.
4. Update each weight w_{kv}, using Eq. (4). Go to step 2.

The Minkowski centre of a given feature v for a cluster S_k can be found over each entity $y_{iv} \in S_k$ by using a steepest descent algorithm [4]. The WK-Means algorithm applies the squared Euclidean distance. In this we still have a user-defined exponent to set, but this is solely a weight exponent, the distance exponent is always two. The iMWK-Means is initialized with a modified version of the intelligent K-Means algorithm [28], which uses the weighted Minkowski distance (2).

The feature weighting procedure used by both WK-Means and iMWK-Means would not deal properly with a subset of V in which features contain relevant, but redundant information. Such features would have similarly low dispersions. Since a single weight w_{kv} is calculated at a time these features would have their weights set to a similarly high value. We believe that this issue makes the removal of redundant features prior to the use of either WK-Means or iMWK-Means beneficial.

Feature selection using feature similarity (FSFS) [29] is one of the most popular unsupervised algorithms that fits to our needs. In this the authors identify features containing similar information by using the maximum information compression index (MIC). This index is defined below, for the variables x and y.

$$2\lambda_2(x,y) = \\ var(x) + var(y) - \sqrt{(var(x) + var(y))^2 - 4var(x)var(y)(1 - p(x,y)^2)}, \quad (5)$$

where $p(x, y)$ is the correlation coefficient given by $\frac{cov(x,y)}{\sqrt{var(x)var(y)}}$. MIC has various interesting properties, such as being invariant to the rotation of the variables and to the translation of the data set [29]. FSFS applies MIC as applied as follows.

1. Choose an initial value for k, following the constrain $k \leq |V| - 1$. Set $R \leftarrow V$, standardize the features rather than entities (for details see Sects. 3 and 4).
2. For each feature $F_i \in R$, calculate r_i^k, the dissimilarity between F_i and its kth nearest neighbour feature in R, using Eq. 5.
3. Find the feature $F_{i'}$ for which r_i^k is minimum. Retain $F_{i'}$ and discard its k nearest features. Set $\varepsilon = r_{i'}^k$.
4. Adjust k in relation to the number of features. If $k < |R| - 1$, then $k = |R| - 1$.
5. If $k = 1$ stop and output R.
6. Adjust k in relation to the similarity. While $r_i^k > \varepsilon$
   (a) $k = k - 1$
   (b) $r_i^k = inf_{F_i \in R} r_i^k$
   (c) if $k = 1$ go to Step 5.
7. Go to Step 2.

The above is a popular and useful algorithm. However, we see two issues that deserve to be addressed: (i) FSFS does not take into account the structure of the data set while selecting features; (ii) FSFS requires the user to define a parameter, k, beforehand. This parameter may increase the algorithm's flexibility, but unfortunately we see no clear method to estimate it. These issues, added to the inability of WK-Means and iMWK-Means to deal with redundant features made us analyse the possibility of a clustering-based solution for feature selection that could be used as a pre-processing step. We present our method in the next section.

3 Algorithm

In this section we present our feature selection method, intelligent K-Means for feature selection (iKFS). Our aim is to cluster the features in V that are similar, rather than the entities. By assigning similar features to the same cluster we can identify and remove those that are redundant. Clearly, there are issues to address under this framework, for instance: (i) how many clusters of feature there would be in a given data set Y; (ii) given a cluster S_k of features, how many features should be selected from it.

Regarding issue (ii), it can be very tempting to keep a single feature from a cluster S_k of features, say the closest to the centroid c_k. However, we do not feel that each cluster should be treated the same, irrespective of its cardinality. With this in mind, we have decided to keep a subset of features of S_k, this subset cardinality is given by F_k.

$$F_k = \left\lceil \frac{|S_k|}{|Y|} * K \right\rceil, \tag{6}$$

where $|S_k|$ and $|Y|$ represent the cardinality of a given cluster of features S_k and the cardinality of the data set Y, respectively. One should note that since we are clustering features, the original data set has to be transposed, so the cardinality of Y is in fact the original number of features (Sect. 4 described experiments with 5,680 features). Equation (6) requires the number of clusters K to be known, taking us back to issue (i), its estimation. In our method we find K by using iK-Means, which can also be used to find good initial centroids for K-Means. The choice of iK-Means was based on its previous success as a clustering algorithm in different scenarios [3, 12, 28]. We introduce our method in full below.

1. Transpose the data set Y so that the original features become entities and then standardize the data set.
2. Apply the iK-Means algorithm setting $\theta = 0$.
3. For each cluster S_k, find F_k (Eq. 6) features that have the highest maximum information compression (Eq. 5) in relation to c_k. Put such features in R.
4. Output the features in R.

We are interested in selecting features that are dissimilar to all others, such features will most likely become singletons during the clustering process. In order to avoid disregarding such features we set $\theta = 0$.

4 Experiments

Electroencephalography (EEG) is the recording of high-dimensional noise-prone signals that can be captured from a brain via a non-invasive procedure. There is considerable research supporting the belief that these signals contain information about the current state or intention of a subject's mind [6, 7, 13, 18, 26].

We have recorded data from three healthy subjects (A, B and C) for our experiments, using five bipolar electrodes (five channels), and a sampling frequency of 250 Hz. These five electrodes were placed on the subjects head following the standard positions in the extended 10–20 system, using fc3 to pc3, fc1 to pc1, cz to pz, fc2 to pc2, and fc4 to pc4.

Our aim is to perform mental task separation, in other words, given a set of possible tasks we would like to know what particular task a subject is thinking about. We have three possible tasks: (i) movement of the left hand; (ii) movement of the right hand; (iii) movement of the feet. After visually suggesting what task the subject should be thinking about, we recorded the EEG data for eight seconds, constituting a trial. Here we intend to cluster trials into the right tasks. Hence, the number of clusters is known to be three. We have gathered data from 240, 120, and 350 trials for each subject, respectively. The difference in the number of trials relates solely to the availability of subjects and staff.

We have pre-processed our data sets in two steps. First, we transformed the data into its power spectrum density (PSD). EEG patterns are normally found in the frequency space rather than amplitude, and PSD helps us to identify periodicities in the data. This transformation has been successfully applied in previous research [6, 7, 13, 18, 26].

Second, having a trial represented by 71 time-related samples each with 80 PSD features, we generated a data matrix for each subject containing the respective number of trials (240, 120 and 350) over 5,680 features (71×80). We then standardized the data numerically.

$$y_{iv} = \frac{x_{iv} - \bar{x_v}}{0.5 * (max(x_v) - min(x_v))}, \quad (7)$$

where x_{iv} represents the PSD value of trial i in feature v, and $\bar{x_v}$ the average of feature v over all trials in the data set. The standard deviation is surely more popular in the standardization of data sets than the range. However, we have opted for the latter as the former favours unimodal distributions [27, 32].

The FSFS algorithm requires an user-defined parameter k. We have performed experiments with such parameter from 4,800 to 5,600 in steps of 100 for WK-Means and iMWK-Means independently. With this interval it is possible for FSFS to select a quantity of features close to that selected by iKFS. Note that the optimal k for WK-Means may not be the same as for iMWK-Means. Our method, iKFS, does not require any extra parameter, so the features used in WK-Means and iMWK-Means when the data is pre-processed with iKFS are exactly the same. Regarding the parameters required by the clustering algorithms themselves, WK-Means and iMWK-Means, we have run experiments from 1.0 to 5.0 in steps of 0.1. In this paper we do not deal with their estimation.

Since we have the labels for each trial in the data sets, we present the best possible results for each of these two algorithms in terms of their cluster recovery. This is calculated by using a confusion matrix.

We show the results of our experiments in Table 1. We are happy to see that in both algorithms, WK-Means and iMWK-Means, iKFS presents features that are more representative. This is visible thanks to the differences in cluster recovery when using FSFS and iKFS in both WK-Means and iMWK-Means. Table 1 also shows us that a much higher number of features does not necessarily means features that are more representative, nor better final accuracy.

Table 1 Cluster recovery of WK-Means and iMWK-Means using the features selected by iKFS and FSFS

	Feature selection method	Exponent		Accuracy		
		Distance	Weight	Mean	Std	Max
Subject A						
WK-Means	FSFS (25)	2.0	3.9	48.2	2.5	52.1
WK-Means	iKFS (20)	2.0	1.5	54.1	1.4	56.2
iMWK-Means	FSFS (28)	3.2	3.2	–	–	51.2
iMWK-Means	iKFS (20)	4.5	4.5	–	–	59.2
Subject B						
WK-Means	FSFS (472)	2.0	4.7	59.2	4.6	73.3
WK-Means	iKFS (9)	2.0	3.6	68.5	4.7	76.7
iMWK-Means	FSFS (393)	2.0	2.0	–	–	65.0
iMWK-Means	iKFS (9)	2.5	2.5	–	–	66.7
Subject C						
WK-Means	FSFS (33)	2.0	4.7	39.6	1.8	42.3
WK-Means	iKFS (11)	2.0	4.5	56.5	0.3	56.9
iMWK-Means	FSFS (33)	4.6	4.6	–	–	42.3
iMWK-Means	iKFS (11)	1.8	1.8	–	–	58.6

The number of features is in the parenthesis

Table 1 does not present average or standard deviation values for iMWK-Means because this is a deterministic algorithm, unlike WK-Means. This happens because iMWK-Means applies a version of iK-Means in its initialization, making it output the same clustering for a given data set irrespective of how many times it is run.

5 Conclusions

In this paper we introduced intelligent K-Means for feature selection (iKFS). This algorithm reduces the number of features that contain similar information, redundant features, in a given data set. It does so by generating clusters of features, instead of entities, using the anomalous pattern method of iK-Means. This anomalous pattern method is rather useful in finding the number of clusters of features in the data set, together with good initial centroids for such clusters. Features within the same cluster are then said to be similar, and by consequence redundant to some degree. Our method selects representative features from each cluster, with the exact quantity calculated based on the cardinality of each cluster and the features maximum information compression.

After explaining the details of our method in Sect. 3 we go to an empirical validation in Sect. 4. In this validation we perform a number of experiments using data sets comprised of Electroencephalography (EEG) signals originated from three healthy subjects. EEG data tends to be high-dimensional (our data sets have 5,680 features) and noisy, hence our choice to use this type of data in our experiments. We have experimented two feature weighting clustering algorithms, WK-Means [11] and iMWK-Means [4], as well as two feature selection algorithms used in the data pre-processing stage, our iKFS and feature selection using feature similarity (FSFS) method [29].

The WK-Means and iMWK-Means algorithms perform feature weighting which allow them to set different degrees of relevance to each feature, as well as simply remove features from a data set. However, the feature weights are set by analysing one feature at a time which means that if two features have the exact same relevant information none will be removed. In our experiments we have found that these feature weighting algorithms benefit from an extra data pre-processing step to reduce the quantity of redundant features in a data set. This is particularly true when iKFS is used in this extra pre-processing step since unlike FSFS, iKFS takes the data structure into account when removing features.

We do find the results we show here to be rather promising and our future research will aim to further optimize the features selected by our method, as well as experiment with data sets from other scenarios. We also intend to experiment with feature weighting algorithms using fuzzy logic that have been recently introduced [34].

References

1. de Amorim, R.C.: An empirical evaluation of different initializations on the number of k-means iterations. Lect. Notes Comput. Sci. **7629**, 15–26 (2013)
2. de Amorim, R.C.: Feature relevance in Ward's hierarchical clustering using the Lp norm. J. Classif. **32**(1) (to appear in 2015)
3. de Amorim, R.C., Komisarczuk, P.: On initializations for the Minkowski weighted k-means. Lect. Notes Comput. Sci. **7619**, 45–55 (2012)
4. de Amorim, R.C., Mirkin, B.: Minkowski metric, feature weighting and anomalous cluster initializing in k-means clustering. Pattern Recogn. **45**(3), 1061–1075 (2012)
5. de Amorim, R.C., Mirkin, B.: Removing redundant features via clustering: preliminary results in mental task separation. In: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems (KICSS), November, pp. 7–9. Krakow, Poland (2013)
6. de Amorim, R.C., Mirkin, B., Gan, J.Q.: A method for classifying mental tasks in the space of EEG transforms. Technical report, Technical Report BBKS-10-01, Birkbeck University of London, London (2010)
7. de Amorim, R.C., Mirkin, B., Gan, J.Q.: Anomalous pattern based clustering of mental tasks with subject independent learning-some preliminary results. Artif. Intell. Res. **1**(1), 46–54 (2012)
8. Ball, G.H., Hall, D.J.: A clustering technique for summarizing multivariate data. Behav. Sci. **12**(2), 153–155 (1967)
9. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Kluwer Academic Publishers, Norwell, MA (1981)
10. Celebi, M.E., Kingravi, H.A., Vela, P.A.: A comparative study of efficient initialization methods for the k-means clustering algorithm. Expert Syst. Appl. **40**(1), 200–210 (2013)

11. Chan, E.Y., Ching, W.K., Ng, M.K., Huang, J.Z.: An optimization algorithm for clustering using weighted dissimilarity measures. Pattern Recogn. **37**(5), 943–952 (2004)
12. Chiang, M.M.T., Mirkin, B.: Intelligent choice of the number of clusters in k-means clustering: an experimental study with different cluster spreads. J. Classif. **27**(1), 3–40 (2010)
13. Chiappa, S., Bengio, S.: HMM and IOHMM modeling of EEG rhythms for asynchronous BCI systems. In: European Symposium on Artificial Neural Networks, ESANN, pp. 193–204 (2004)
14. De Soete, G.: Optimal variable weighting for ultrametric and additive tree clustering. Qual. Quant. **20**(2–3), 169–180 (1986)
15. De Soete, G.: OVWTRE: a program for optimal variable weighting for ultrametric and additive tree fitting. J. Classif. **5**(1), 101–104 (1988)
16. DeSarbo, W.S., Carroll, J.D., Linda, C.A., Green, P.E.: Synthesized clustering: a method for amalgamating alternative clustering bases with differential weighting of variables. Psychometrika **49**(1), 57–78 (1984)
17. Frigui, H., Nasraoui, O.: Unsupervised learning of prototypes and attribute weights. Pattern Recogn. **37**(3), 567–581 (2004)
18. Gan, J.Q.: Self-adapting BCI based on unsupervised learning. In: 3rd International Workshop on Brain-Computer Interfaces, pp. 50–51 (2006)
19. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. J. Mach. Learn. Res. **3**, 1157–1182 (2003)
20. Huang, J.Z., Ng, M.K., Rong, H., Li, Z.: Automated variable weighting in k-means type clustering. IEEE Trans. Pattern Anal. Mach. Intell. **27**(5), 657–668 (2005)
21. Huang, J.Z., Xu, J., Ng, M., Ye, Y.: Weighting method for feature selection in k-means. In: Computational Methods of Feature Selection, pp. 193–209 (2008)
22. Jain, A.K.: Data clustering: 50 years beyond k-means. Pattern Recogn. Lett. **31**(8), 651–666 (2010)
23. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281–297. California, USA (1967)
24. Maitra, R., Peterson, A.D., Ghosh, A.P.: A systematic evaluation of different methods for initializing the k-means clustering algorithm. Trans. Knowl. Data Eng. 522–537 (2010)
25. Makarenkov, V., Legendre, P.: Optimal variable weighting for ultrametric and additive trees and k-means partitioning: methods and software. J. Classif. **18**(2), 245–271 (2001)
26. Millan, J., Mouriño, J.: Asynchronous BCI and local neural classifiers: an overview of the adaptive brain interface project. IEEE Trans. Neural Syst. Rehabil. Eng. **11**(2), 159–161 (2003)
27. Milligan, G.W., Cooper, M.C.: A study of standardization of variables in cluster analysis. J. Classif. **5**(2), 181–204 (1988)
28. Mirkin, B.: Clustering for Data Mining: A Data Recovery Approach, vol. 3. CRC Press (2005)
29. Mitra, P., Murthy, C.A., Pal, S.K.: Unsupervised feature selection using feature similarity. IEEE Trans. Pattern Anal. Mach. Intell. **24**(3), 301–312 (2002)
30. Pena, J.M., Lozano, J.A., Larranaga, P.: An empirical comparison of four initialization methods for the k-means algorithm. Pattern Recogn. Lett. **20**(10), 1027–1040 (1999)
31. Steinbach, M., Karypis, G., Kumar, V.: A comparison of document clustering techniques. In: KDD Workshop on Text Mining, pp. 525–526. Boston (2000)
32. Steinley, D.: Standardizing variables in k-means clustering. In: Classification, Clustering, and Data Mining Applications, pp. 53–60. Springer (2004)
33. Steinley, D., Brusco, M.J.: Initializing k-means batch clustering: a critical evaluation of several techniques. J. Classif. **24**(1), 99–121 (2007)
34. Svetlova, L., Mirkin, B., Lei, H.: MFWK-Means: Minkowski metric fuzzy weighted k-means for high dimensional data clustering. In: IEEE 14th International Conference on Information Reuse and Integration (IRI), pp. 692–699. IEEE (2013)
35. Tibshirani, R., Walther, G., Hastie, T.: Estimating the number of clusters in a data set via the gap statistic. J. R. Stat. Soc.: Ser. B (Stat. Methodol.) **63**(2), 411–423 (2001)
36. Xu, R., Wunsch, D.: Survey of clustering algorithms. IEEE Trans. Neural Netw. **16**(3), 645–678 (2005)

Part VI
Music and Video

Building Internal Scene Representation in Cognitive Agents

Marek Jaszuk and Janusz A. Starzyk

Abstract Navigating in realistic environments requires continuous observation of a robot's surroundings, and creating internal representation of a perceived scene. This incorporates a sequence of cognitive processes, including attention focus, recognition of objects, and building internal scene representation. The paper describes selected elements of a cognitive system, which implement mechanisms of scene observation based on visual saccades, followed by creating the scene representation. Such internal representation is a foundation for scene comparison, based on a distance matrix. This, in turn, allows for recognizing known places, changes in the environment, or structure of complex objects.

Keywords Visual saccades · Scene representation · Episodic memory

1 Introduction

One of our natural skills is the ability of intelligently acting in complex environments. For instance, we can find our destination in a city, interact with other people to exchange information or arrange objects in a room in a suitable way. In these tasks, we outperform current man-built systems such as mobile robots. Thus, it is highly desirable to develop artificial agents, which will be able to support us in performing rich variety of tasks, both in our daily environment, as well as in dangerous and inaccessible places.

M. Jaszuk (✉) · J.A. Starzyk
University of Information Technology and Management,
ul. Sucharskiego 2, Rzeszów, Poland
e-mail: marek.jaszuk@gmail.com

J.A. Starzyk
School of Electrical Engineering and Computer Science, Ohio University,
Athens, OH 45701, USA
e-mail: starzykj@ohio.edu

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_36

For this purpose, we devise computer models of intelligent information processing called cognitive systems [6]. Such systems are expected to collect and process stream of sensory data to create internal representation of the environment, and autonomously undertake the proper actions. The components of cognitive systems mimic different mental functions. Among their most important elements are: sensory and motor functions, as well as different kinds of memory, including semantic, episodic, procedural, and working memory.

To carry out successful navigation in complex environments, mobile robots must acquire and maintain internal representation of the environment. This is not a trivial task and many factors affect reliability of such models. In the presented paper, we try to solve this problem, by using a saccade-based algorithm of scene observation and building internal scene representation. We also introduce a scene comparison method, based on distances between scene elements. An earlier version of this research was presented at the International Conference KICSS2013 [5].

The paper is organized as follows: Sect. 2 discusses visual saccades and the designed cognitive system architecture; In Sect. 3 we discuss building the scene model from the data delivered by a sequence of visual saccades; Sect. 4 presents a method designed for comparing scenes; In Sect. 5 we present the VEEMA simulation, where the experiments are carried out; In Sect. 6 we present sample experimental results demonstrating the performance of the scene comparison method.

2 The Fundamentals

2.1 Visual Saccades

Visual saccade is a fast eye movement, from one focus point to another [10]. The main purpose of saccadic movements is to focus cognitive attention on objects potentially important to the observer. The visual data from the environment are subconsciously filtered in order to identify regions with higher saliency. Next, attention is switched rapidly between such regions, resulting in a sequence of saccadic movements. After focusing attention on a given region, the observer is able to acquire more precise visual data needed to recognize the object of interest. Additionally, the observer should be able to assess the distance between particular attention focus points. Humans realize this task through stereovision. In robots, it can be realized using several techniques depending on the robot's construction.

In robotic systems there are two possible ways of implementing visual saccades. The first of them relies on camera movements, which resembles biological reality [2]. A simpler approach is based on software analysis of images from a static camera mounted on a robot. This approach does not require any sophisticated hardware, allows for high speed saccades and can be used practically on all mobile robots equipped with a camera. A robot can still move its camera to follow changes in the environment, but with a lower speed than the one required for saccadic motion.

2.2 *The Cognitive System*

The presented work focuses on selected parts of a cognitive system named MLECOG (Memory Learning and Cognition). The system is based on motivated learning algorithm [9]. This learning method as well as the whole system is founded on the idea of embodied intelligence, where whole learning comes from interaction with the environment instead of scripted algorithms designed by engineers [3]. The structure of the discussed system represented on a general level is shown in Fig. 1. Its central part form emergent motivations, and the goal creation system, which are responsible for stimulating the agent to act and learn new skills. The set of sensors acquires the raw stream of data from the environment. Subconscious attention switching responds to salient features of the sensory data and drives the agent's attention. Semantic memory stores general facts about the world, learned from repeating experiences, while episodic memory is responsible for storing particular experiences and situations. Working memory is responsible for processing filtered and semantically recognized data, building internal representation of experiences, and storing them in the form of episodes.

Figure 2 shows in more detail elements of the system responsible for building the environment representation, and storing it in the form of sequences of episodes in episodic memory. The video stream registered by a camera contains huge amount of data, with various level of significance. To protect the system against the flood of unimportant details, the stream needs to be filtered in order to find the elements worth the agent's attention. Thus, the video data need to be initially processed in order to identify regions with high visual saliency. The agent then follows the visually salient elements in a sequence of saccadic movements. After focusing attention on a given

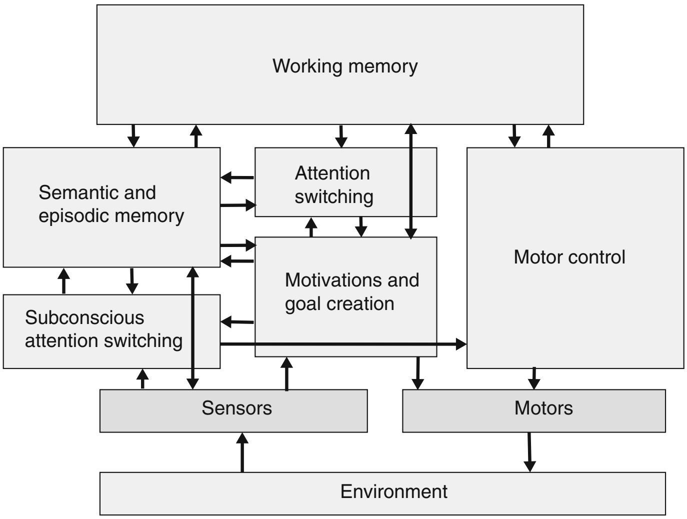
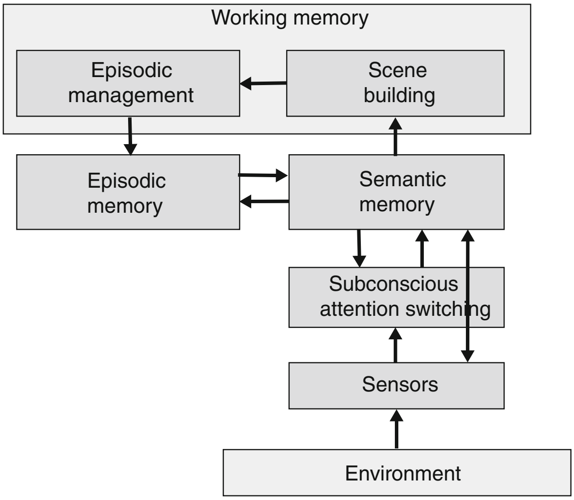


Fig. 1 The general structure of the MLECOG system

Fig. 2 The structure of the cognitive system elements responsible for building the internal scene representation



region, the content of this region is analysed and recognized cognitively using the semantic memory. The sequence of recognized objects is delivered together with spatial location to the scene building module, where the scene model is created. The scene memory is the key element for the episodic management module, which identifies changes within the scenes, and decides about saving valuable experiences to the episodic memory.

There are a number of methods for reconstructing 3D geometry of the environment. Among the most commonly used are those based on multiple view techniques [4]. Our approach, however, is different, because it incorporates the element of active scene observation and cognitive recognition. When a visually attractive region is identified, the part of the scene view is analysed and transformed into a set of features. For this purpose, we use the SURF algorithm [1] due to its scaling and rotation invariance. In this way, we reduce computationally expensive processing of raw pixel data. After recognizing the SURF features by semantic memory the scene data are replaced by a relatively small number of cognitively recognized objects. In this way, recognition of known scenes is performed in a similar way like humans do—by comparing a set of characteristic objects represented on symbolic level. Also storage of such symbolic data is much more efficient.

3 Building Internal Scene Representation from a Sequence of Saccades

Building the scene representation requires identifying objects within the environment, and finding their spatial placement (location map). The objects, perceived in a single saccade, are relatively small, which resembles human perception in which

Fig. 3 A single object under attention focus with *circles* indicating SURF keypoints

the area of visual attention focus is quite limited. If a scene object is large, it has to be observed in a sequence of saccades. Such object can be decomposed and represented in the agent's memory as a complex object. An example of an object being the subject of attention focus is shown in Fig. 3. The image comes from the virtual 3D environment in which our experiments are carried out. This environment is discussed in more detail in Sect. 5.

The circles in Fig. 3 indicate the keypoints resulting from analysis of the image with the SURF algorithm. We are using the implementation of SURF available in the OpenCV library [8]. The radius of particular circles indicates the size of respective visual features indicated by the keypoints. The semantic memory is composed of 3D models of objects characterized by the SURF descriptors. Such representation allows for recognizing objects irrespectively of the direction of observation. It is enough to match the features detected within the environment view, to the features of objects from the memory.

To illustrate the mechanism of creating the internal scene representation we will analyze a sample sequence of saccadic movements. For simplification, we limit the considerations to a 2D environment map. This analysis can later be extended to the 3D case.

When an agent enters a new scene, he starts from focusing attention on the object, which is the most visually salient. We use a simple method based on histograms to asses visual saliency. Although this method is not too precise, we use it because it is fast and allows the system to work in real time.

For practical reasons, location of the first object perceived in the scene is the most convenient choice for the origin of the scene coordinates Fig. 4a. The agent can assess the distance a_1 to the object, but it is currently not of much use, because the object is located in coordinates irrespective of the agent location. In physical robots

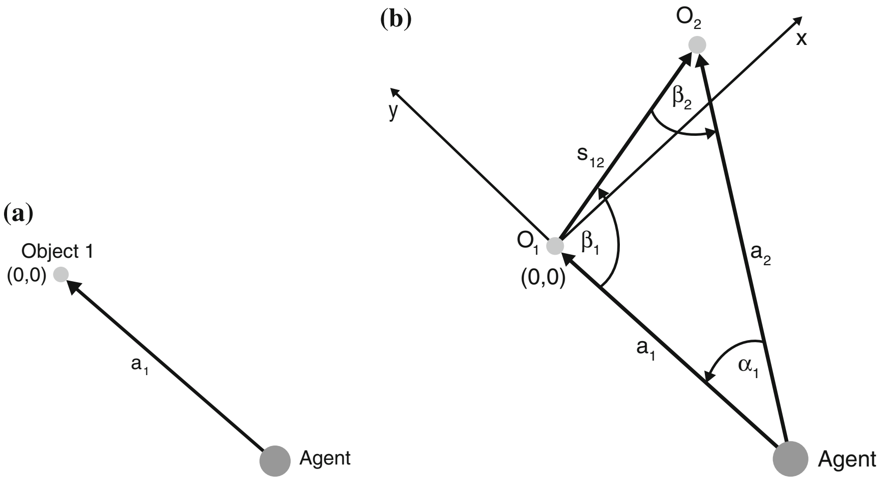


Fig. 4 The first stage of building the scene map: **a** perception of the first object—origin of the local coordinate system, **b** first saccade and location of the second object

the assessment of the distance can be realized using stereovision, range laser or other sensors, depending on the robot's design.

The next step is focusing attention on the second scene object. This is again done on the basis of visual saliency. The transition between the first and the second object is the first saccade made by the agent during observing the scene. After focusing attention on the second object, and recognizing it cognitively, the agent has to locate it within his internal scene representation. To do this it is necessary to determine the relative location of the first and second object. The distance is not measured directly, but it can be computed given that the agent can measure the distance from his temporary location to both the objects (a_1 and a_2), and the angle α_1 between the two directions (Fig. 4b).

The distance between O_1 and O_2, which is the length of saccade S_{12}, can be obtained from the law of cosines applied to the triangle with vertices in O_1, O_2, and the agent location:

$$s_{12}^2 = a_1^2 + a_2^2 - 2a_1 a_2 \cos \alpha_1. \tag{1}$$

To locate the second object in the internal scene representation, the direction of the axes of local scene coordinates is needed. We can choose the direction of the y axis as an extension of the line between the agent and O_1 (Fig. 4b). This defines the direction of the x axis as perpendicular to y. Given the axes we can compute the location of the second object within the local coordinates:

$$x_{O2} = s_{12} \cos \left(\beta_1 - \frac{\pi}{2} \right), \tag{2a}$$

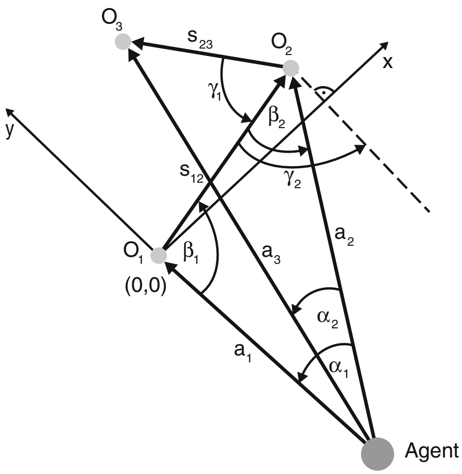


Fig. 5 Adding the third object to the scene map

$$y_{O2} = s_{12} \cos\left(\pi - \beta_1\right), \tag{2b}$$

To make the description of the scene building mechanism complete, we have to demonstrate adding yet another object to the scene. Locating the new object in the local coordinates is based on known positions of the two previously memorized objects. The agent, as in previous cases, locates a visually salient object and focuses attention on this object. He also measures the distance to this object, and the angle α_2 between the directions of observation of object 2 and object 3 from the agent's position (Fig. 5). Given the data, the length of saccade s_{23} can be computed analogously as in the first saccade (Fig. 5).

To position object 3 in the local coordinates we need to know the angle $\gamma_1 + \gamma_2$. This will allow for computing the projections of s_{23} on the x and y axes. To find this angle we start from using the law of cosines to compute β_2 and $\beta_2 + \gamma_1$ directly, which after subtracting gives γ_1. We already know coordinates of object 2 so the γ_2 angle can also be computed from one of trigonometric functions. In this way, we obtained $\gamma_1 + \gamma_2$. Now the coordinates of object 3 can be computed as follows:

$$x_{O3} = x_{O2} - s_{23} \cos\left(\gamma_1 + \gamma_2 - \frac{\pi}{2}\right), \tag{3a}$$

$$y_{O3} = y_{O3} + s_{23} \sin\left(\gamma_1 + \gamma_2 - \frac{\pi}{2}\right), \tag{3b}$$

The procedure described for object 3 is repeated for every new saccade. To locate the new object within the local scene coordinates only the most recent saccade is needed. Although some imprecisions are possible during measuring distances between objects within the scene, the described procedure allows to avoid cumulating errors in the scene representation building. Moreover, the scene map can be updated while the agent explores the environment.

Building 3D scene representation can be done by extending the described methodology. The general procedure in the 3D case is very similar to the one already presented. In every step of the scene observation, the agent focuses attention on an object with high visual saliency, and measures distances to the perceived objects, together with angles between directions of observation of most recently perceived objects. The first observed object becomes the origin of the scene, and the subsequent objects are added in relation to the most recently added ones. The additional dimension results in necessity of using data from three most recent saccades, to position a new object within the scene memory (in 2D scene only 2 are required). The detailed discussion of saccades in 3D space, does not bring much new to understanding of discussed mechanism, and thus is omitted here.

4 Distance-Based Scene Comparison

Let us assume that D_1 and D_2 are two matrices of distances between objects in scenes SC_1 and SC_2, respectively, and that the order of rows and columns corresponds to equivalent symbols in both scenes. In case a symbol is unique to a scene, we add a new row and column to both matrices. If one of the compared scenes is missing an object, its distance matrix will have the distance to this object set to some large value like 2× the size of the scene diagonal. In this way both matrices always have the same size, even if they contain different collections of objects. Notice that D_1 and D_2 are symmetrical matrices with zeros on diagonal (an object has 0 distance to itself). In addition to the distance matrices we assign some significance to the scene objects. The significance results from the agent's internal needs, and motivations, and allows for steering the agent's attention not only by external objective factors, but also by internal perception of object significance. Every object has some significance and its significance may differ, depending on the scenes in which it appears. For the purpose of scene comparison we define the mutual significance matrix as follows:

$$S_M = \left[s_{ij}\right]_{m\times m}, \text{where} \quad s_{ij} = \prod_{k=1,2} s_{ki}s_{kj} \tag{4}$$

where s_{ki} is ith object significance in kth scene. To normalize the distance similarity measure we calculate a distance difference matrix ΔD:

$$\Delta D = \left[\delta_{ij}\right]_{m\times m} = \begin{cases} \frac{\left\| d_{1,ij} - d_{2,ij} \right\|}{d_{diag}}, & \text{if} \quad d_{1,ij} \le d_{diag} \wedge d_{2,ij} \le d_{diag} \\ 1, & \text{if} \quad d_{1,ij} > d_{diag} \vee d_{2,ij} > d_{diag} \end{cases} \tag{5}$$

where $d_{diag} = max\left(d_{diag1}, d_{diag2}\right)$ is the maximum (diagonal) distance of both scenes. This guarantees that $0 < \delta_{ij} < 1$ for all i, j. If at least one of the two elements for which the distance difference is computed, does not exist in one of the scenes, such distance difference takes the maximal value equal to 1. Then we calculate the normalized scene similarity as

$$S_{SC} = \frac{\sum_{i=1}^{m} \sum_{j=i+1}^{m} \left(1 - \delta_{ij}\right)^{p} s_{ij}}{\sum_{i=1}^{m} \sum_{j=i+1}^{m} s_{ij}}. \tag{6}$$

This scene similarity has values between 0 (dissimilar) and 1 (identical). The p factor is an empirical constant, which is adjusted to control the sensitivity of the similarity measure—the larger p, the more sensitive the measure is to differences in the compared scenes.

When the objects are not unique (i.e., they are indistinguishable), a similar approach can be applied. However, in this case there is no unique ordering of the objects due to repetitions. The scenes can be compared only on the basis of geometrical relations. To do this we must either examine all combinations of ordering of the repeated objects that is NP hard, or use a heuristics, where we assign object pairs based on similarities of their corresponding distance vectors $d_{1,ij}$ and $d_{2,ij}$ within uniquely defined scene elements. The procedure that we use is numerically efficient, and has a quadratic computational cost.

In realistic scenario, the scene will be a mixture of both unique objects, as well as some sets of indistinguishable objects (e.g., a sequence of windows of the same shape and size in a house). Thus we can summarize the whole procedure of scene comparison in the following steps:

1. Obtain distance matrices D_1 and D_2 for the two scenes.
2. Order columns according to objects with unique symbols. If a symbol has several instances, each instance corresponds to a separate column.
3. Reorganize columns within each group of identical symbols in increasing order of distance values.
4. Within each group of identical symbols repeat heuristic procedure to obtain object instance assignment.
5. Find normalized scenes similarity using (6).

5 The VEEMA Simulation Environment

Our current implementation of the cognitive agent is designed to cooperate with a virtual environment called VEEMA (Virtual Environment with Embodied Motivated Agent). This environment is based on the NeoAxis 3D Game Engine [7] (Fig. 6). The agent is able to explore the virtual world similarly as a robot could do it in the real world. The advantage of using the virtual environment is full control over the environment.

Fig. 6 A third person camera view on the VEEMA environment. The *white circle* indicates the agent

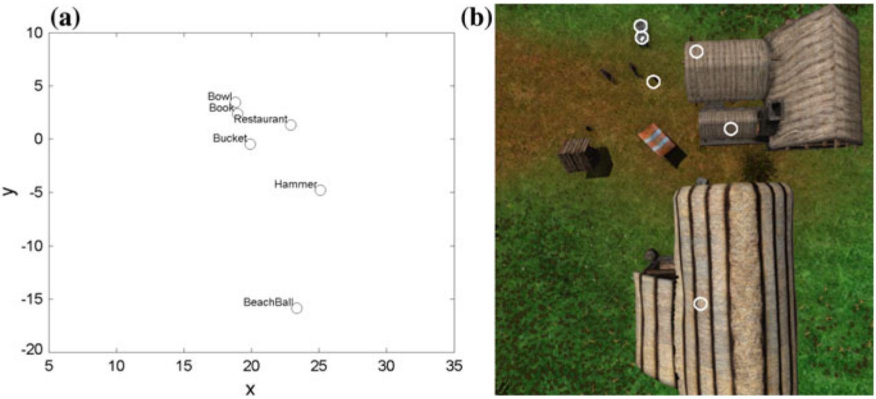


Fig. 7 The layout of objects in the considered learning scenario: **a** projection on the x-y plane, **b** view from the *top* camera with indicated locations of objects of agent's interest

In the current version, the environment explored by the agent is relatively simple, and far from complexity of real-world environments. It was adapted to a motivated learning scenario [9], in which the agent searches among a collection of resources, and learns skills required to survive in the environment. The agent can focus attention on objects placed in the environment, and reconstruct internally the scene layout. A sample placement of objects seen by the agent is shown in Fig. 7.

6 Experimental Results

To test performance of the scene comparison algorithm described in Sect. 4 we constructed numerous experiments, in which two scenes with different locations and types of objects were compared. Here, we present two sample experiments. The first one shows how the similarity value changes, when the first scene consists of a fixed set of objects (6 unique objects), and the second scene consists initially of 2 objects, and then subsequent objects are added to the scene in the same positions as in the first scene (Fig. 8a). Objects from the first scene are indicated by crosses, while objects from the second scene are indicated by circles. The numbers in the figure indicate the type (symbols) of objects. When the number of objects in the second scene reaches the number of objects in the first scene (similarity measure = 1), a sequence of new objects (not present in scene 1) was added until their number reached 11 Fig. 8b. The positions of newly added objects were chosen randomly.

The set of similarity values obtained in the experiment is shown in Fig. 9a. As one can see, initially similarity increases monotonically with the increasing number of objects, and after reaching maximum (=1) for identical scenes, decreases with the number of objects added to the second scene. This is what we expected from the similarity measure. The presented results are obtained for $p = 10$ in Eq. 6. For this, and the other experiments, the value seems quite accurate, because it allows for sharp distinctions between the scenes. However, it is possible that for more complicated scenes, different values will be necessary.

The second experiment is quite similar to the first one, but it explores the similarity of scenes composed of identical objects. In this case, we apply the heuristic procedure of object matching, and the scene comparison is based solely on the geometry of object's locations. As expected, the maximal similarity is reached, when the number of objects in the second scene equals the number of objects in the first scene. However, the changes in similarity are more sharp and less regular, than in the previous experiment.

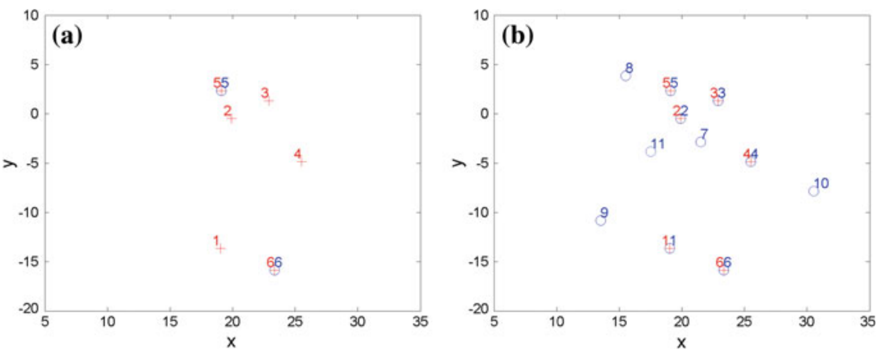


Fig. 8 The layout of objects in the experiment: **a** the initial stage of experiment, **b** the final stage of experiment

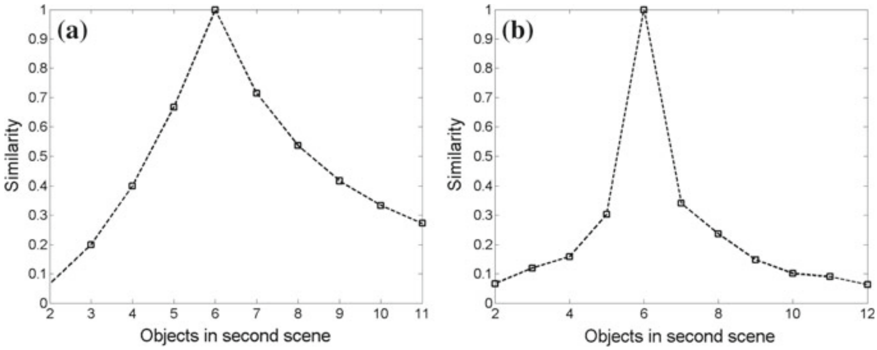


Fig. 9 Similarity of scenes with respect to changing number of objects: **a** scenes with unique objects, **b** scenes with indistinguishable objects

7 Conclusions

In the paper, we introduced a method of building internal environment representation within a cognitive agent. Also we introduced a distance-based similarity measure for scene comparison. Sample experiments based on a virtual 3D environment were performed, in which the performance of the proposed measure was demonstrated. The results show, that the proposed methodology gives satisfying results, however, the scene created in the virtual environment is simplified, and of much smaller complexity, than real-world scenes. Thus, the aim of our future work will be extending the system to be able to process all the data delivered by a video stream, either registered in the virtual simulation or in real environment.

Acknowledgments The research was supported by The Polish National Science Centre, grant No. 2011/03/B/ST7/02518.

References

1. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: SURF: Speeded Up Robust Features. Comput. Vis. Image Und. **110**(3), 346–359 (2008)
2. Begum, M., Karray, F.: Visual attention for robotic cognition: a survey. IEEE Trans. Auton. Ment. Dev. **3**(1), 92–105 (2011)
3. Brooks, R.: Intelligence without representation. Artif. Int. **47**, 139–159 (1991)
4. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge (2003)
5. Jaszuk, M., Starzyk, J.A.: Building internal scene representation in cognitive agents. In: Skulimowski, A.M.J. (ed.) Looking into the Future of Creativity and Decision Support Systems: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems, Kraków, Poland, 79, Nov 2013. Advances in Decision Sciences and Future Studies, vol. 2, pp. 279–290. Progress & Business Publishers, Kraków (2013)

6. Langley, P., Laird, J.E., Rogers, S.: Cognitive architectures: research issues and challenges. Cogn. Syst. Res. **10**, 141–160 (2009)
7. The NeoAxis Game Engine. http://www.neoaxis.com/
8. The OpenCV Computer Vision Library. http://opencv.org/
9. Starzyk, J.A., Graham, J.T., Raif, P., Tan, A.-H.: Motivated learning for the development of autonomous systems. Cogn. Syst. Res. **14**, 10–25 (2012)
10. Yarbus, A.: Movements of the Eyes. Plenum Press, New York (1967)

Optical Music Recognition as the Case of Imbalanced Pattern Recognition: A Study of Single Classifiers

Agnieszka Jastrzebska and Wojciech Lesinski

Abstract The article is focused on a particular aspect of classification, namely the imbalance of recognized classes. The paper contains a comparative study of results of musical symbols classification using known algorithms: k-nearest neighbors, k-means, Mahalanobis minimal distance, and decision trees. Authors aim at addressing the problem of imbalanced pattern recognition. First, we theoretically analyze difficulties entailed in the classification of music notation symbols. Second, in the enclosed case study we investigate the fitness of named single classifiers on real data. Conducted experiments are based on own implementations of named algorithms with all necessary image processing tasks. Results are highly satisfying.

Keywords Pattern recognition · Classification · Imbalanced data

1 Introduction

The problem of pattern recognition is a data mining branch studied and developed for many years now. In a number of its applications, satisfying results have already been achieved, however, in many fields it is still possible to obtain better results. Certainly, one of the possible research fields is the issue of imbalanced data. For the purposes of this paper, the issue of imbalanced data can be defined as a case in which there are one or more of the following characteristics:

1. there are significant differences in the number of elements between classes;
2. elements within the same class have shapes that do not overlap;

A. Jastrzebska · W. Lesinski (✉)
Faculty of Mathematics and Information Science, Warsaw University of Technology, Plac Politechniki 1, 00-660 Warsaw, Poland
e-mail: wlesinski@ii.uwb.edu.pl

W. Lesinski
Faculty of Mathematics and Computer Science, University of Bialystok, ul. Sosnowa 64, 15-887 Bialystok, Poland

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_37

3. objects belonging to different classes are very different in size; and
4. objects in different classes are both simple and complex.

In this paper, the issue of imbalanced data is illustrated on the example of music notation symbols. The characters on the score have all of the four characteristics given above.

Musical notation symbols appear with varied frequency. Some of them, such as quarter and eighth notes, are very common, often appearing several times within a single line of the score. Other (including rest and accidentals) occur frequently, but still much more rarely. There are also symbols (breve note, longa note), which appear occasionally in few musical compositions.

The problem of incompatible shapes within the same class applies to part of the examined symbols. These include, among others, arcs, crescendo, and diminuendo. Different shapes of arcs are shown in Fig. 1.

Objects belonging to particular classes of musical notation strongly differ in their size. Figure 2 illustrates this diversity. It shows big, medium, and small symbols. These terms are conventional, of course. Certainly, we can count arc into large

Fig. 1 Incompatible shapes of the "arc" symbol

Fig. 2 Different sizes of chosen musical notation symbols. Starting from the *top* placed are: *dot*, *flat*, *G clef* and *arc*

objects which is well visible on Fig. 2. Into large characters treble clef can be also included. In the terms of size, dot is the opposite of the arc. It is the smallest character that occurs in the score. Into small symbols, but much larger than the dot, we can also count some of the accents, the whole note, or a whole note rest. Comparison of the dots and the arc perfectly shows poor balance of musical notation, in terms of size. Between the extreme values, we can find many characters of intermediate sizes including quarter, flats, and naturals.

General methodology of optical music recognition has been already researched and described in [10]. We would like to highlight that studied problem of imbalance of classes is an original contribution to the field of music symbols classification. The aim of our study is to investigate how well-known classification tools deal with imbalanced data. In this paper, presented are single classifiers only. The research is based on actual opuses. Applied classification algorithms have been implemented in C++. Developed program works with both high- and low-resolution images of musical symbols.

The paper is organized as follows. Section 2 lists the basic information about the classification and used classifiers. In Sect. 3 the learning and testing sets are outlined. Section 5 describes empirical tests.

2 Preliminaries

2.1 Classification

Pattern recognition is usually performed on observed features, which characterize objects, rather then on objects directly. Therefore, we distinguish a mapping from the space of objects $\mathbb{O}$ into the space features $\mathbb{X}$, i.e., $\phi : \mathbb{O} \rightarrow \mathbb{X}$. This mapping is called *features extractor*. Then, we consider a mapping from the space of features into the space of classes $\psi : \mathbb{X} \rightarrow \mathbb{C}$. Such a mapping is named *classifier*. It is worth to notice that the term classifier is used in different contexts: classification of objects and classification of features. Meaning of this term can be concluded from context. Therefore, we will not distinguish explicitly, which meaning is taken.

Composition of the above two mappings constitute the classifier: $\Psi = \phi \circ \psi$. In other words, the mapping $\mathbb{O} \stackrel{\Psi}{\rightarrow} \mathbb{C}$ is decomposed to $\mathbb{O} \stackrel{\phi}{\rightarrow} \mathbb{X} \stackrel{\psi}{\longrightarrow} \mathbb{C}$.

The space of features $\mathbb{X}$ is usually the Cartesian product of features $X_1, X_2, \ldots, X_n$, i.e., $\mathbb{X} = X_1 \times X_2 \times \cdots \times X_n$. Therefore, mapping Ψ and ϕ operate on vectors $x_1, x_2, \ldots, x_n$, where x_i is a value of the feature X_i for $i = 1, 2, \ldots, n$. For simplicity, a vector of values of features will be simply called a vector of features.

Summarizing, we explore pattern recognition problem searching for an (object) classifier

$$\Psi : \mathbb{O} \rightarrow \mathbb{C} \tag{1}$$

which is decomposed to a feature extractor

$$\phi : \mathbb{O} \to \mathbb{X} \quad (2)$$

and a (features) classifier or a classification algorithm

$$\psi : \mathbb{X} \to \mathbb{C} \quad (3)$$

The classifier ψ divides features' space onto so-called decision regions:

$$D_X^{(i)} = \psi^{-1}(i) = \{x \in X : \psi(x) = i\} \quad \text{for every} \quad i \in M \quad (4)$$

and then, of course, the features extractor splits the space of objects into classes

$$O_i = \phi^{-1}(X^{(i)}) = \{o \in \mathbb{O} : \phi(o) \in X^{(i)}\} \quad \text{for every} \quad i \in M \quad (5)$$

or equivalently

$$O_i = \Psi^{-1}(i) = (\phi \circ \psi)^{-1}(i) = \phi^{-1}\left(\psi^{-1}(i)\right) \quad \text{for every} \quad i \in M \quad (6)$$

We assume that the classification algorithm splits the space of features' values, i.e., it separates the whole space X into pairwise disjoint subsets, which cover the whole space X

$$(\forall i, j \in M, i \neq j) \;\; D_X^{(i)} \cap D_X^{(j)} = \emptyset \quad \text{and} \quad \bigcup_{i \in M} D_X^{(i)} = X \quad (7)$$

In our current research investigated are classification schemes based on classical sets theory. In such case belongingness is crisp. An element either belongs to given class or not. There are fuzzy generalizations of classical approach to classification. In the next step of our research, we will take a closer look at other information representation models and we will investigate their suitability to optical music recognition. We would like to verify, if other approaches, especially ones involving bipolarity (c.f. [8, 9, 11]) may enhance classification results with imbalanced data.

2.2 *Classifiers*

In current section, we describe briefly applied classifiers, that is: k-Nearest Neighbors, k-means, Mahalanobis minimal distance, and decision tree.

2.2.1 k-Nearest Neighbors

The k-nearest neighbors algorithm [4] is among the simplest one of all machine learning algorithms. For a given object, which is being classified, its k-nearest neighbors are extracted from the learning set. The object of interest is classified to the class having majority in the set of its k nearest neighbors. k is a positive integer, typically small. If k is equal to 1, then the object is simply assigned to the class of its nearest neighbor. Advantage of the k-nearest neighbors algorithm is its high recognition rate. Among disadvantages, the most important one is kNN's high time complexity. Its big computation overload is a significant disadvantage.

2.2.2 K-Means

K-means is an algorithm similar to k-Nearest Neighbors, which attempts to solve the problem of large amount of calculations in recognition stage, what is typical for k-NN classifier. In the learning stage, this classifier divides every class from the learning set into k clusters with the use of k-means algorithm [7]. The classifier does not store whole of the training set, but only the calculated centroids, thus the expenditure of calculations is reduced in the recognition stage. In the recognition stage, a search for the nearest centroid is performed. The class, to which given centroid belongs is the answer of the algorithm. The distance between two elements in the recognition stage is determined by previously settled metric.

2.2.3 Mahalanobis Minimal Distance

The Mahalanobis minimal distance classifier can be interpreted as a modification of the naive Bayes algorithm. In this method we assume, that a priori likelihood for every class is equal each to other, i.e., $\pi_1 = \pi_2 = \cdots = \pi_M$, and all observations come from normal distribution with the same covariance matrices. With this assumption, the Bayes rule assumes the following form:

$$(x - m_k)^T \sum{}^{-1} (x - m_k) \tag{8}$$

where x is classified object, m_k is a mean of C_k class calculated from training set, and given below

$$m_k = \frac{1}{n_k} \sum_{i=1}^{n_k} x_{ik} \tag{9}$$

and $\sum$ is a covariance matrix defined by Eq. 10

$$\sum = \frac{1}{n - M} \sum_{k=1}^{M} \sum_{i=1}^{n_k} (x_{ki} - m_k)(x_{ki} - m_k)^T \tag{10}$$

where $x_{1k} \ldots x_{nk}$ are vectors representing objects from class C_k, m_k is a mean vector from this class, n_k—number of elements in class C_k, n—number of all elements and M number of classes. An Eq. 8 is called Mahalanobis distance. In this method, an object x is classified to the class j if square of Mahalanobis distance for this class is the smallest one.

2.2.4 Decision Trees

A decision tree is a decision support tool that uses a tree structure for decision making and classification [2, 14]. Popular algorithms used for construction of decision trees have inductive nature. Top-bottom tree building scheme is used. In this scheme, building a tree starts from the root of the tree. Then, a feature for testing is chosen for this node and the training set is divided into subsets according to values of this feature. For each value there is a corresponding branch leading to a subtree which should be created on the basis of the proper testing subset. This process stops when a stop criterion is fulfilled and the current subtree becomes a leaf.

The stop criterion indicates when the construction process needs to be brought to a standstill, which is when for some set of samples we should make a leaf, not a node. An obvious stop criterion could be situation when

- a sample set is empty,
- all samples are from the same class, and
- attributes set is empty.

In practice, criteria given above sometimes make the model overfitted to learning data. So, other stop criteria or mechanisms, such as pruning, are necessary in order to avoid the overfitting problem.

Finally, classification of a given object is based on finding a path from the root to a leaf along branches of the tree. Choices of branches are done by assigning tests' results of the features corresponding to nodes. The leaf ending the path gives the class label for the object.

3 Data Set

The recognized set of music notation symbols had about 27,000 objects in 20 classes. There were 12 classes defined as numerous and each of them had about 2000 representatives. Cardinality of other eight classes was much lower and various in each of them. Part of the examined symbols was cut from chosen Fryderyk Chopin's compositions. Other part of the symbols' library comes from our team's other research projects. They were divided into two groups: regular and irregular (rare) classes. Regular classes include flat, sharp, natural, G and F clefs, piano, forte, mezzo-forte, quarter rest, eight rest, sixteenth rest, and flagged stem. Irregular classes consist of accent, breve note, C clef, crescendo, diminuendo, fermata, arc, and 30 s rest. Image sets coming from regular classes consisted of 2000 objects each. Sets of irregular classes are significantly smaller.

4 Feature Extraction

Classification was done on features characterizing every symbol. Vectorized and numerical features characterizing symbols were defined based on the experience of authors. The following features were used in the experiment:

- histograms, i.e., relations between the number of pixels with a given value of a feature and the number of all pixels. Histogram of black pixels was used in the experiment,
- horizontal and vertical projections (also known as histograms), i.e., numbers of black pixels in rows (for horizontal projection) and in columns (for vertical projection),
- horizontal and vertical transitions, i.e., the number of pairs of neighboring white/ black pixels in every row for horizontal transitions and in row for vertical transition. Transition allows defining objects with complicated shapes, for example treble clef,
- left, right, top and bottom margins, i.e., for every row it is the number of white pixels counted from the left edge of the image to the first black pixel (left margin), for every column it is the number of white pixels counted from the top edge of the image to the first black pixel (upper margin) etc. This feature shows the symbol's position on the image. It is useful, for instance, to distinguish natural from sharp,
- directions, i.e., for a given pixel it is the longest segment of black pixels in given directions (usually horizontal, vertical, left, and right diagonal directions are considered) which include given black pixel,
- moments,
- average 3, i.e., the average value calculated of three neighboring elements in features' vector,
- average 5, i.e., the average value calculated of three neighboring elements in features' vector,
- difference, i.e., differences between two consecutive values of in features' vector,
- numerical features got as casted numerical parameters of the above vectorized featured:
  - max, i.e., the maximum value of all values in features' vector,
  - min, i.e., the minimum value of all values in features' vector,
  - ave, i.e., the average value of all values in features' vector,
  - maxPos, minPos, avePos, i.e., positions of given max, min, and ave values in features' vector.

5 Experiment and Results

In order to determine and compare the quality of described methods, we have used

- accuracy calculated by the equation:

$$acc = \frac{number\ of\ well\ recognized\ objects}{number\ of\ all\ objects} \tag{11}$$

- classifier's error:

$$err = \frac{number\ of\ objects\ recognized\ incorrectly}{number\ of\ all\ objects} \tag{12}$$

In the second stage irregular classes, were added to the previously recognized classes. At this point, attention was paid to changes in the efficiency of recognition and recognition of particular irregular classes. Apart from *acc* and *err*, measures showing the influence of classes counting less elements should be used for classifiers assessment.

To evaluate the classifiers, two measures were calculated: sensitivity $(TP)/(TP + FN)$ and precision $(FP)/(TP + FP)$. For these calculations, our multiclass problem was turned to m two class problems (*one class contra all others*). All measures were calculated for each class. In the end, average measure was determined.

5.1 Recognition of Regular Classes

The experiment was divided into two parts. In the first one, only elements belonging to the regular classes were being recognized. It allowed to determine the appropriate structure of classifiers. For each classifier, the appropriate training set size was established. Dependency of the classifier efficiency upon the size of the training set was examined. The tests were performed for learning sets counting 1, 10, 20, 50, 100, 200, and 400 in each class. Intuition suggests that it should increase with the growing number of learning symbols. To evaluate the classifiers, we used accuracy described in Sect. 5.

Best results were obtained by kNN and decision tree. Tests show that learning set counting 400 elements in each class is enough to achieved good outcome. All results are shown in Table 1.

For some classifiers, other parameters were examined too. In the case of kNN factor k was tested. The k parameter was examined with the training set counting 400 elements for every recognized class. Tests were performed for $k = 1, 2, 3, 5, 10, 15, 20$. The worst performance this classifier obtained at $k = 1$. It was 95 %. The highest efficiency, 98 %, was achieved with $k = 5$ and $k = 10$. For this reason, for further examinations, the tested parameter will take value that generates less computational costs, which is 5.

Table 1 The effectiveness of recognition of regular classes

Classifier	1	10	50	100	200	400
kNN	70.21	88.34	93.42	95.07	97.93	98.14
K-means	65.45	82.86	88.49	91.20	92.72	93.27
Mahalanobis	62.63	82.02	87.75	90.37	92.52	93.22
Decision tree	35.69	82.23	92.64	96.28	97.92	98.29

For k-means classifier parameter, k was determined. Similarly to kNN, the analysis was carried out for the learning set containing 400 symbols in each class. Tests were performed for $k = 1, 2, 3, 5, 10, 15, 25, 50$. The worst performance classifier obtained with $k = 1$. With the growing number of clusters for given class, the efficiency of the method rises. Unfortunately, at the same time computational complexity is increased and the duration of algorithm's run is prolonged. The increase of effectiveness stopped after k reached 10.

5.2 *Irregular Classes*

Establishing the correct structure of classifiers was an introduction to recognize the whole issue. Classes determining the imbalance have been added into regular classes. In this case, apart from the global effectiveness, worth noticing is the recognition within irregular classes. Regular classes had 400 representatives in the training set. The number of other classes is presented in Table 2. The results for all of the classes are shown in the Table 3. Figure 3 shows sensitivity for all classes, Fig. 4—precision.

k-Nearest Neighbors: Tests were performed for $k = 1$ and $k = 5$. Global effectiveness compared to the results described in Sect. 5.1 slightly decreased (difference of 0.5). Unfortunately, recognition effectiveness of irregular classes is significantly lower than regular classes. The best performance out of irregular classes was obtained by C clef. This is due to the relatively large training set and the shape of

Table 2 Learning and testing sets for irregular classes

Class	Learning set	Testing set
Accent	30	65
Breve	1	2
Crescendo	55	100
Diminuendo	51	97
Fermata	35	46
Clef C	100	178
Arc	100	155
Thirty-second rest	20	35

Table 3 Effectiveness of recognition with irregular classes

Class	kNN		K-means		Mahalanobis		Decision tree	
	Sens.	Prec.	Sens.	Prec.	Sens.	Prec.	Sens.	Prec.
F clef	99.19	99.50	98.63	98.94	99.31	99.19	99.19	99.50
G clef	99.13	99.31	98.63	98.14	98.69	98.63	99.13	99.19
Flat	96.69	98.22	88.50	92.49	88.82	92.81	96.73	98.10
Natural	97.00	96.16	90.81	87.79	91.23	88.32	96.82	96.09
Sharp	97.00	96.46	86.56	86.13	88.34	86.38	96.68	96.27
Forte	99.94	99.32	99.69	97.49	99.69	97.50	100.00	99.32
Mezzo-forte	100.00	100.00	100.00	99.19	100.00	99.81	100.00	99.50
Piano	100.00	99.48	100.00	99.44	100.00	96.64	100.00	99.26
Quarter rest	96.63	97.48	87.31	88.25	88.46	87.95	97.02	97.49
Eight rest	97.32	96.59	90.00	88.62	90.00	89.33	97.19	96.58
1/16 rest	97.38	97.68	91.00	90.77	91.75	90.67	97.25	97.74
Flagged stem	97.50	98.17	88.56	94.72	87.82	94.43	98.34	98.30
Clef C	98.88	96.70	91.01	95.29	98.88	97.21	98.88	96.67
Arc	98.06	98.70	87.74	97.84	94.84	98.66	94.19	98.65
Crescendo	88.00	88.89	85.00	85.00	91.25	94.79	87.25	88.78
Diminuendo	85.77	85.57	83.51	79.41	92.78	92.78	85.48	83.67
Accent	90.77	88.06	81.54	80.30	100.00	92.86	88.06	87.69
Fermata	86.96	97.56	89.13	89.13	97.21	89.80	90.19	91.49
1/32 rest	80.00	77.78	71.43	71.43	82.40	78.38	84.12	87.88
Breve	0.00	ud.	0.00	ud.	0.00	ud.	0.00	ud.

Sens.—Sensitivity, Prec.—Precision, ud.—undetermined

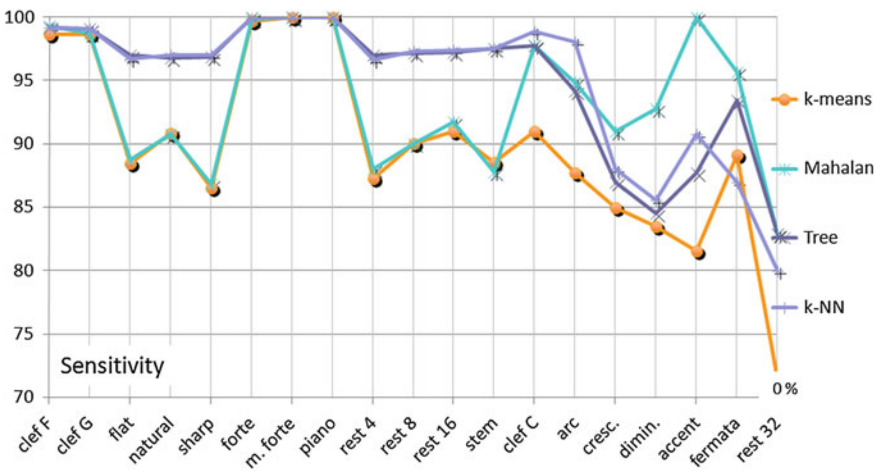


Fig. 3 Recognition of all classes—sensitivity

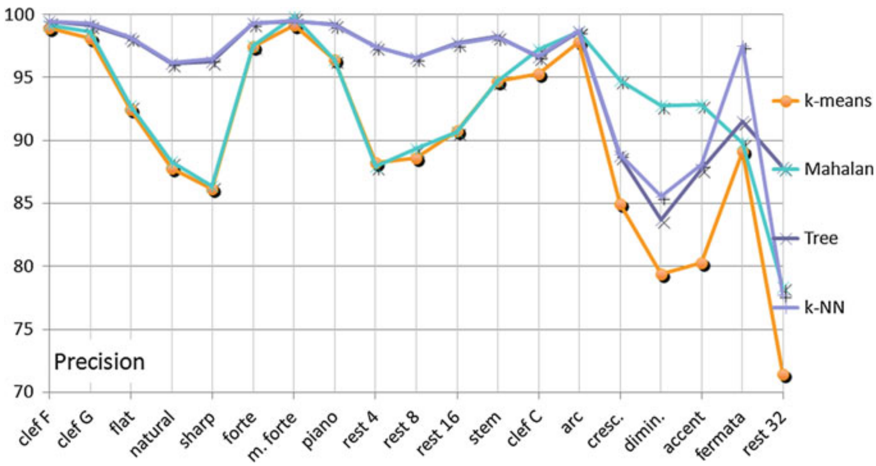


Fig. 4 Recognition of all classes—precision

this symbol which distinguishes it from the others. The arc was also recognized with high efficiency. This symbol also had a large training set. Another group of symbols to be considered is the accent, crescendo, and diminuendo. These symbols are very similar to each other and are mistaken for one another. The breve note is also noteworthy. This symbol hardly occurs in modern musical scores, so it is difficult to find many of its copies. This note at $k = 5$ is not recognized at all, whereas for $k = 1$ its recognition is equal 100 %.

K-means: The study on this classifier was carried out with $k = 5$ ($k = 1$ for breve note). Also in this case, increasing the number of classes resulted in a slight decrease of global effectiveness. In the instance of irregular classes, effectiveness of this method was below k-Nearest Neighbors classifier. Again, the most recognized classes were the C clef and the arc. Note: if the class is counting less elements, it is recognized worse. Classification errors occurred more or less in the same places in which they appeared for the k-Nearest Neighbors, but it was more of them.

Mahalanobis: Global result of the classifier using Mahalanobis distance was also reduced with an increase in the number of analyzed classes. The overall result of this method turned out to be worse than in the k-nearest neighbors method. However, interestingly, particular rare symbols were recognized better by this method. The algorithm was especially able to isolate accent, which has been recognized with a 100 percent efficiency. Unfortunately, crescendo and diminuendo were still mistaken for one another. Problems also appeared with distinguishing 1/32 rest from 1/16 rest. The breve note remained unrecognizable.

Decision tree: Previous tests have shown that a decision tree is sensitive to the size of the training set. Unsurprisingly, symbols from irregular classes were recognized poorly. The global efficiency of this classifier dropped by two percent after adding new classes. Unfortunately, the results of the irregular symbols recognition were not satisfying. Like in previous methods, classified the best were C clef and the arc.

This is probably due to their significant representation in the training set. Also here appeared the distortion: accent—diminuendo and diminuendo—accent. What is odd, tree often mistaken fermata symbol for the mezzo-forte. There were also errors in recognition 1/32 rest. The breve note remained unrecognizable too.

6 Conclusions

The article discusses results of classification task performed on images of musical symbols. Difficulty of undertaken research lies in imbalance of classes and high variability of objects. We have concisely illustrated most important issues, which we have had to address in order to proceed with optical music recognition and classification.

In summary, 20 classes of musical notation symbols were classified. 12 of them have been considered as a regular classes, the other as irregular classes. The recognition effectiveness of regular classes was satisfying. k-Nearest Neighbors classifiers and decision tree recognized them with the efficiency of 98 %. Slightly worse results were obtained by the classes counting less elements. The less symbols class contained, the worse were the results.

In order to improve the results, features vector could be modified or other methods of classification could be applied. It is possible to use complex classifiers such as bagging and random forest in further works.

Acknowledgments The research is supported by the National Science Center, grant No 2011/01/B/ST6/06478, decision no DEC-2011/01/B/ST6/06478.

References

1. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer (2006)
2. Breiman, L., Friedman, J., Stone, C.J., Olshen, R.A.: Classification and Regression Trees. Wadsworth, Belmont (1984)
3. Chen, C., Liaw, A., Breiman, L.: Using random forest to learn imbalanced data. Technical report, Dept. of Statistics, U.C. Berkeley (2004)
4. Cover, T.M., Hart, P.E.: Nearest neighbor pattern classification. IEEE Trans. Inf. Theory **1**(13), 21–27 (1967)
5. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley, New York (2001)
6. Garcia, V., Sanchez, J.S., Mollineda, R.A., Alejo, R., Sotoca, J.M.: The class imbalance problem in pattern recognition and learning. In: II Congreso Espanol de Informatica, pp. 283–291 (2007)
7. Hartigan, J.A., Wong, M.A.: A k-means clustering algorithm. Appl. Stat. **28**, 100–108 (1979)
8. Homenda, W., Pedrycz, W.: Balanced fuzzy gates. In: Greco, S., Hata, Y., Hirano, S., Inuiguchi, M., Miyamoto, S., Nguyen, H.S., Słowiński, R. (eds.) RSCTC 2006. LNAI, vol. 4259, pp. 107–116. Springer, Heidelberg (2006)
9. Homenda, W., Pedrycz, W.: Symmetrization of fuzzy operators: notes on data aggregation. In: Halgamuge, S.K., Wang, L. (eds.) Computational Intelligence for Modelling and Prediction. Studies in Computational Intelligence, vol. 2, pp. 1–17. Springer, Heidelberg (2005)

10. Homenda, W.: Optical music recognition: the case study of pattern recognition. In: Kurzynski, M., Puchala, E., Wozniak, M., Zolnierek, A. (eds.) Computer Recognition Systems. Proceedings of the 4th International Conference on Computer Recognition Systems CORES05, Part VII, pp. 835–842 (2005)
11. Homenda, W.: Balanced fuzzy sets. Inf. Sci. **176**, 2467–2506 (2006)
12. Kurzynski, M.: Statistical Pattern Recognition. Publishing House of Wroclaw University of Technology, Wroclaw (1997) (in Polish)
13. Lesinski, W., Jastrzebska, A.: Optical music recognition as the case of imbalanced pattern recognition: a study of single classifiers. In: Skulimowski, A.M.J. (ed.) 8th International Conference on Knowledge, Information and Creativity Support Systems, pp. 267–278. Progress & Business Publishers (2013)
14. Quinlan, J.R.: Induction of decision trees. Mach. Learn. **1**, 81–106 (1986)
15. Rebelo, A., Fujinaga, I., Paszkiewicz, F., Marcal, A.R.S., Guedes, C., Cardoso, J.S.: Optical music recognition: state-of-the-art and open issues. Int. J. Multimedia Inf. Retrieval **1**, 173–190 (2012)

Modeling and Recognition of Video Events with Fuzzy Semantic Petri Nets

Piotr Szwed

Abstract This paper addresses the problem of modeling and automated recognition of video events. We propose to use Linear Temporal Logic as a language for events specification and Fuzzy Semantic Petri Nets (FSPN) as a tool for their recognition. FSPN are Petri nets coupled with an underlying fuzzy ontology. The ontology stores assertions (facts) concerning classification of objects and detected relations. Fuzzy predicates querying the ontology content are used as guards of transitions in FSPN. Tokens carry information on objects participating in a scenario and are equipped with weights indicating likelihood of their assignment to places. In turn, the places correspond to scenario steps. We describe a prototype detection system consisting of an FSPN interpreter, the fuzzy ontology, and a set of predicate evaluators. Initial tests yielding promising results are reported.

Keywords Petri nets · Fuzzy ontology · Event recognition · Temporal logic

1 Introduction

Recognition of video events is an important research area in computer vision. Developed methods may have many potential applications: intelligent video surveillance, video indexing engines, and various systems in which human–computer interactions are based on the interpretation of video content. Practical implementations of event recognition systems face two problems [12]. The first is related to extraction of features from input video allowing for identification and tracking of objects. The second pertains to an approach to event modeling. Specifications of events, to be meaningful and manageable, should preferably be decoupled from code performing low-level recognition tasks and use semantic description of objects and their relations.

A typical video event recognition system includes several processing steps: background maintenance, object detection, tracking, object classification, and finally

P. Szwed (✉)
AGH University of Science and Technology, Kraków, Poland
e-mail: pszwed@agh.edu.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_38

events inferencing. In this paper, however, we focus on the last stages of event recognition chain. We propose to define event scenarios as Linear Temporal Logic (LTL) formulas and then, to translate them to Fuzzy Semantic Petri Nets (FSPN) formalism [18, 19]. Predicates appearing in LTL formulas become guards of transitions in the resulting Petri net, whereas, places correspond to scenario steps. Recognition of events is performed during FSPN execution. Flowing tokens carry information on objects participating in an event and are equipped with fuzzy weights indicating likelihood of the event occurrence. Results of an analysis of the current scene are stored in a fuzzy ontology comprising identified objects and fuzzy relations between them. The ontology content is updated on arrival of each video frame, then all enabled transitions in FSPN are fired and the reached Petri net marking is analyzed. If a token appears in a place modeling an important scenario step, such situation is to be reported as a *semantic output*.

The conceptual layout of a proposed video event recognition system is depicted in Fig. 1. We focus on components marked in gray: *Fuzzy Ontology* constituting an intermediate abstraction layer between description of tracked objects, *Scenario Specification* expressed in Linear Temporal Logic (LTL), and *Fuzzy Semantic Petri Nets*, which are obtained by translating LTL formulas.

The proposed approach stems from an observation that in the domain of formal software verification, LTL has been successfully applied to specify and check temporal requirements related to the behavior of concurrent programs. Hence, an idea of applying LTL to detection of video events occurrences. However, in opposition to models of programs, whose behavior can be observed with the 100 % accuracy, results of video content analysis are inherently uncertain, as they are calculated in several steps, including background subtraction, object recognition, classification, and tracking. Each of them may produce small cumulative errors. Moreover, semantic scenario specifications may suffer from vagueness, by referring to such terms as *near*, *walking*, and *running*, which can be differently understood and defined. Hence, fuzziness is the proposed mean to manage uncertainty and vagueness.

The paper is organized as follows: Sect. 2 reports known approaches to the specification and analysis of events. Section 3 discusses the application of Linear Temporal

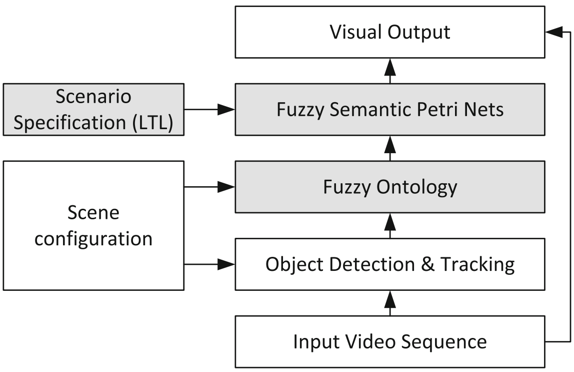


Fig. 1 Conceptual model of a system for semantic video events analysis

Logic in this field. It is followed by Sect. 4 which describes the fuzzy ontology. FSPN are defined in Sect. 5. A system implementing the proposed approach is presented in Sect. 6 and finally Sect. 7 gives concluding remarks.

2 Related Works

Recognition of video events has been intensively researched over last 15 years. A large number of methods are reported in recent surveys: [2, 12]. Systems for video recognition usually have a layered architecture [7, 16], in which lower level layers provide an *abstraction* of meaningful aspects of video sequences, whereas higher level layers are related to formalisms used for *event modeling* and algorithms that detect events based on formal specifications. The greatest diversity can be observed in approaches to event modeling.

Probabilistic state-based methods use models comprised of states and transitions, in which transitions are attributed with probability factors learned from annotated video. During the analysis of an input video sequence a likelihood of a situation is computed. This group includes methods based on neural networks [5], Hidden Markov Models, Dynamic Bayesian Networks [1], and stochastic Petri nets [13].

In grammar-based methods, complex activities are represented by production rules that generate strings of atomic actions. Hence, complex events can be recognized by language parsing techniques [10]. In the review [2], a limitation of these methods as regards concurrent activities was indicated. The criticism seems to be founded in a case, where sequences of single actions are analyzed. However, in a more general setting, e.g., this is provided by the Kripke structure [11], each string element is a set of low-level events occurring parallelly and, in consequence, the concurrent events can be tracked (see Sect. 3).

Description-based approaches specify events and scenarios using high-level languages, either textual or graphical as situation graph trees [16, 17] and Petri nets [4, 9, 13]. The methods falling into this category are considered *semantic*, as specifications are prepared by experts, who give meaningful names to events, engaged objects, actions, and conditions. Descriptions are often hierarchical: complex events can be expressed as graphs of subevents. Models also may include constraints and knowledge about scene objects, e.g., in [16] they are expressed as formulas of a Fuzzy Metric-Temporal Horn Logic. In some approaches events and their ingredients, the types of participating objects and relations are defined as ontologies [3].

Petri Nets (PNs) are applied in the field of event detection in two modes [12]. In the first mode of *object PNs* tokens represent objects and places object states and transitions events of interest. Such an approach was applied to the surveillance of traffic [9] and people [6]. In the second mode of *plan PNs* places correspond to subevents building up a plan. A presence of a token in a place indicates that a particular event assigned to the place is occurring. The latter approach was applied in [4] to people surveillance.

3 Scenario Specification with Temporal Logic

Temporal logic [15] is a symbolic language allowing to express temporal (unquantified) relationships between events or conditions. Formulas in temporal logic are constructed from propositions linked by classical logic operators and *temporal operators*. For Linear Temporal Logic (*LTL*), most often used operators are: $\Box q$ (the formula q is *always* true starting from a certain moment in time) and $\Diamond p$ (the formula q will *eventually* became true in the future).

To give a simple example, the temporal formula $a \Rightarrow \Diamond b \Rightarrow \Box c$ specifies, that at the beginning a is satisfied, then b happens, and finally c becomes always true.

Semantics of LTL is defined with a model called *Kripke structure* [11], which can be defined formally as a tuple $K = (Props, S, T, s_0, L)$, where *Props* is a set of atomic propositions, S is a set of states, $T \subseteq S \times S$ is a flow relation, s_0 is an initial state and $L: S \rightarrow 2^{Props}$ is a function, that assigns *sets* of true propositions to states. For LTL the flow relation T together with the initial state s_0 defines a linear sequence of states (worlds): $s_0, s_1, s_2, \ldots, s_n, \ldots$.

In case of a formal verification conducted by model checking, the states represent snapshots of program memory in consecutive time moments. For the intended application of LTL to specify event scenarios, the sequence of states corresponds to a video sequence or, more precisely, to the sets of objects recognized in particular frames, their properties, and relations. Hence, in formal specifications of events we replace propositions by unary and binary predicates in Fuzzy Description Logics [14] querying those relations.

We will discuss a scenario specification on an example of graffiti painting event shown in Fig. 2. The event develops in the following steps (subevents): (i) *init*: a person appears on a scene, (ii) *move*: the person moves towards the wall, (iii) *front*: the person is in front of the wall, (iv) *appear*: a graffiti appears on the wall, (v) *remain*: the graffiti remains on the wall.

The scenario of the event can be formalized as an LTL formula comprised of propositions corresponding to subevents (1). It should be noted that to some extent these lower level events can overlap in time, e.g., when a graffiti appears the person is probably still in front of the wall.

$$init \Rightarrow \Diamond move \Rightarrow \Diamond front \Rightarrow \Diamond appear \Rightarrow \Box remain \qquad (1)$$

Fig. 2 Graffiti event steps

Fig. 3 Scene configuration: the wall and the windows boundaries are marked

In the next stage, the initial scenario specification is refined by replacing propositions with formulas composed from unary and binary predicates. The resulting specification for the discussed event is given by the formula (2).

$$\begin{aligned}
&Person(p), atBorder(p), Wall(w)\\
&\Rightarrow \Diamond(movesTowards(p,w))_{\{3,\infty\}}\\
&\Rightarrow \Diamond(inFrontOf(p,w))_{\{10,\infty\}}\\
&\Rightarrow \Diamond(newObject(g), inside(g,w), notInsideSomeWindow(g))_{\{3,\infty\}}\\
&\Rightarrow \Box(inside(g,w), isStill(g), notInsideSomeWindow(g))_{\{3,\infty\}}
\end{aligned} \tag{2}$$

The scenario references three objects: p of type *Person*, w of type *Wall*, and g, an object that is introduced in the fourth step. The specification relies on classification of objects: p is a *Person*, their relations, e.g., graffiti g is on a wall w, and a scene configuration that should provide information on shapes of walls and windows. The later for the discussed example is given in Fig. 3. Integer numbers in curly brackets define time intervals $[t_l, t_h]$. A subevent should last at least t_l time units to be accepted as a scenario step occurrence. Then it can be reported, but also it enables a transition to a next subevent. Compound events older then t_h are ignored.

4 Fuzzy Ontology

The fuzzy ontology constitutes an intermediate layer between information on tracked objects and fuzzy Petri nets. Whereas objects within the tracking model are described with numeric values, like size, distance, or speed, the ontology provides a kind of abstraction referencing linguistic terms, e.g., *a person*, *an object is inside other object*, or *a person is in front of other object*.

Ontologies are often described as unions of two layers: terminological: *TBox*, which comprises concepts and relation types (including taxonomic relations between concepts) and assertional: *ABox* gathering facts about individuals and their relations.

For *fuzzy ontologies* and corresponding Fuzzy Description Logics, these relations are extended by adding weights being real numbers from [0, 1]. They can be used to express uncertainty, e.g., with respect to class membership or relation occurrence. The formalization of fuzzy ontology language including fuzzy classes, roles (object properties), and datatypes can be found in [14].

In the presented solution the TBox is limited to fuzzy concepts, like *Person*, *Wall*, taxonomic relations and object properties: *inside*, *notInsideSomeWindow*. The ABox is comprised of individuals including tracked objects and predefined scene objects, fuzzy class membership relations that are represented by unary predicates returning values from [0, 1], e.g., *Person*(x) and asserted fuzzy relations between individuals, e.g., *inFrontOf*(x, y). It should be noted that a unary predicate describing property of an object, e.g., *isStill*(x) can be considered equivalent to the class membership axiom: $x \in$ *isStill*.

The ontology supports queries for objects present in ABox and their relations. Assertions on relations in ABox are made with *evaluators*, functions (or more precisely delegates in an object-oriented implementation) that examine object model and calculate fuzzy weights of predicates. In opposition to approach proposed in [14], evaluators are external entities beyond the ontology. This allows for greater flexibility in their construction. In many cases, they have a form of fuzzy membership functions described by line segments, similar to these discussed in [17], but they can be also based on other features, as Jaccard metrics applied to object areas.

The predicate *newObject*(g) references temporal information stored in the underlying frame sequence model. Its evaluator shown in Fig. 4a uses a membership function that takes as argument the difference between current frame number and the frame, in which the object g appeared.

The predicate *inside*(x, s) (Fig. 4b) divides the object x into a grid of cells and calculates how many of them overlap with a scene object s. It is used internally by the predicate *notInsideSomeWindow*(x) which denotes a class of objects satisfying the axiom $\neg\exists w \cdot Window(w) \sqcap inside(x, w)$. The condition was introduced to prevent from classifying as objects left on the wall visual changes occurring inside the windows, e.g., reflections on the glass, window opening, and people moving behind. In this case the predicate value is calculated by the implemented reasoner according to the formula: $1 - \max\{w \in Window : inside(x, w)$. For the example in the Fig. 4b the evaluator *notInsideSomeWindow*(g) yields the value $22/25 = 0.88$.

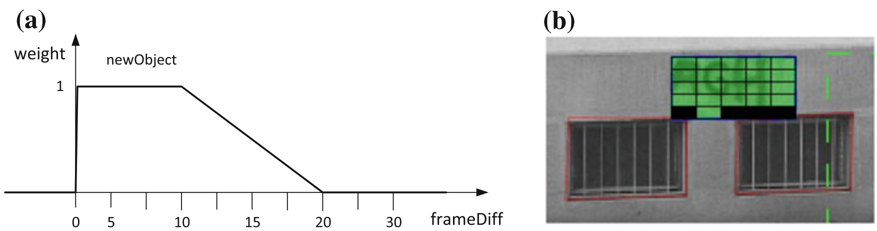


Fig. 4 Examples of evaluators: **a** newObject(x), **b** inside(x,s)

5 Fuzzy Semantic Petri Nets

Classical method of verification, whether a sequence of worlds satisfy an LTL formula, consists in converting it into more manageable representation, namely a Büchi automaton [8]. Such automaton accepts an infinite sequence of symbols, each of which can be considered as subsets of logical propositions having true value in the subsequent worlds forming a semantic model.

In the presented approach, we use Fuzzy Semantic Petri Nets as a tool for scenario analysis. The nets have a typical structure of Büchi automata, however, they can process multiple tokens simultaneously. This feature is particularly important, because it allows for reasoning about overlapping scenario occurrences, in which various combinations of objects participate.

To manage uncertainty of an input data and vagueness of specifications, fuzzy predicates returning values from $[0, 1]$ are used as transition guards. These values are then combined with the weights of tokens flowing through a net. Tokens, in turn, represent event occurrences. This enables monitoring occurrences of scenario steps and reasoning about their likelihood. Moreover, sequences of accepted states strictly defined with LTL formulas can to some extent be interleaved with states not satisfying the specified conditions. In such a case, the weight of a token expressing the scenario likelihood decreases.

Formal definition of FSPN comprises three concepts: Petri net structure, a binding, and fuzzy marking. We start with some auxiliary definitions. Unary predicate is defined as a pair (n, v_s) where, n is a predicate name and v_s is a variable name referring to a *subject* of the predicate. Binary predicate is a triple (n, v_s, v_o); the variable v_o is a predicate *object*. Set of all unary and binary predicates is denoted by *Preds*. By $Vars(p)$ we denote a set of variables appearing in the predicate p. Analogously, for a set $C \subseteq Preds$ we define $Vars(C)$, as $\bigcup_{p \in C} Vars(p)$.

Definition 1 (*Petri net structure*) Petri net *PN* is a tuple $(P, T, F, Preds, G, L, H)$, where P is a set of places, T is a set of transitions, P and T are satisfying $P \cap T = \emptyset$ and $P \cup T \neq \emptyset$. $F \subseteq P \times T \cup T \times P$ is a set of arcs (flow relation), and *Preds* is a set of unary and binary predicates. $G\colon T \rightarrow 2^{Preds}$ is a guard function that assigns sets of predicates to transitions. $L\colon P \rightarrow \mathbb{N} \cup \{0\}$ is a function assigning lower bound to a place; this value defines how long a token should stay in a place to be allowed to leave it. $H\colon P \rightarrow \mathbb{N} \cup \{\omega\}$ assigns upper bound to a place. The symbol ω represents infinity.

The set of input places for a transition $t \in T$ is denoted as $\bullet t = \{p \in P\colon (p, t) \in F\}$ and the set of output places as $t\bullet = \{p \in P\colon (t, p) \in F\}$

Definition 2 (*Binding*) Let V be set of variables and I a set of objects. *Binding* b is defined as a partial function from V to I. A variable v *is bound* for a binding b, iff $v \in \mathbf{dom}\, b$. A set of all bindings is denoted by B.

Let $p \in Preds$ a predicate and $b \in B$ be a binding. Predicate value for a binding $val\colon Preds \times B \rightarrow [0, 1]$ is a function that assigns value from the interval $[0, 1]$ to a pair (p, b). If $Vars(p) \setminus \mathbf{dom}\, b \neq \emptyset$, then $val(p, b) = 0$.

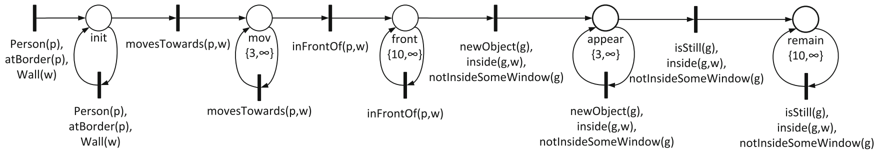


Fig. 5 Fuzzy Semantic Petri Net representing the graffiti painting event

Definition 3 (*Fuzzy marking*) A set of fuzzy tokens FT is defined as $FT = B \times \mathbb{R} \times (\mathbb{N} \cup \{0\}) \times (\mathbb{N} \cup \{0\})$. Components of a token tuple $(b, w, c, \tau) \in FT$ are the following: $b \in B$ denotes a binding, $w \in [0, 1]$ is a fuzzy weight, $c \geq 0$ is a counter storing information, how long the token rests in a place and τ is a time stamp. For a Petri net $PN = (P, T, F, Preds, G)$ *fuzzy marking* $FM : P \rightarrow 2^{FT}$ is defined as a function that assigns sets of fuzzy tokens to places of the net.

The definition of FSPN is too general with respect to the structure that is required to represent a scenario expressed with an LTL formula. In fact, we use state machines, i.e., Petri nets satisfying $|\bullet\, t| \leq 1$ and $|t\, \bullet| = 1$ for each $t \in T$.

Figure 5 gives an example of a net representing the formula (2). Logical conditions appearing in subformulas become guards for two transitions: the first leading to a place and the second forming a self-loop (a transition, for which $\bullet t = t\bullet$ holds). The net in Fig. 5 represents a particular simple case. Translation of more complex formulas, e.g., containing disjunctions, may result in multiple transitions linking places or forks.

The behavior of FSPNs defined in previous section differs from the standard semantics for Petri nets, as they are not intended analyze such issues as concurrency or conflicts, but to perform a kind of fuzzy reasoning and classification of sequences of events. A single step of FSPN execution is comprised of three basic stages:

1. *Firing enabled non-initial transitions and generating new tokens*. During this stage, for each pair (t,ft) consisting of a transition t and a token ft in its input place, the transition guard is evaluated, then aggregated with the token weight and assigned to a new token ft' introduced to the transition's output place. The new token obtains a timestamp equal to the iteration number. There are, however, some variations to the above procedure: created tokens must have weight above a certain threshold (we used 0.25 in experiments); in the case, where the transition guard contains a free variable, it must be bound to an object in the ontology.
2. *Removing old tokens*. It is assumed, that a creation of a new token ft' from ft consumes a portion of its weight. If this value falls below a certain threshold, the token ft is removed. Also in this step, multiple tokens sharing the same binding and assigned to the same place are aggregated.
3. *Firing initial transitions*. Finally, new tokens are introduced into the net by firing initial transitions (i.e., satisfying $\bullet t = \emptyset$). For each initial transition, variables appearing in its guard are bound to objects, then the guard value is calculated and

used as a weight of new tokens. A threshold (0.2) preventing creation of tokens with a small weight is used as well as there is implemented a mechanism, which does not allow introducing tokens with a binding already present in the net.

The semantics of Petri nets proposed in this paper is close to referenced in Sect. 2 *plan PNs*, as tokens represent combination of objects participating in scenarios. There are, however, some salient differences. (1) In probabilistic PNs discussed in [4] in case of a conflict (e.g., two enabled transitions sharing input place with a single token) only one transition with a higher learned probability would fire, whereas in our model they both can be executed and produce two tokens. This allows to reason concurrently about scenario alternatives. Moreover, a weak initial likelihood of a scenario branch can be amplified by future events. (2) In our approach all enabled transitions are executed in a single parallel step. (3) Petri nets modeling scenarios are actually state machines. Their structure is sufficient to construct the Büchi automaton [8] representing an LTL formula.

6 Event Detection System and Initial Experiments

In this section, we describe a prototype system allowing to test defined events. The system takes at input an annotated video sequence defining tracking information. For each frame a list of segments and identified objects are provided. The data does not represent ground truth, but in real the output from experimental tracking algorithms [19] developed within a project SIMPOZ[1] aiming at implementation of an automated video surveillance system.

The architecture of the system is presented in Fig. 6. Main components are: the *Fuzzy ontology*, a set of *Evaluators*, i.e., functions calculating fuzzy values of predicates from low-level features of tracked objects, and the *Fuzzy Semantic Petri Net*. The system is also equipped with a GUI providing visual output.

The control flow during a single iteration was marked in Fig. 6 with numbers in circles.

1. On arrival of a new frame, asserted relations between objects are removed from the ontology, then new identified objects are added as individuals.
2. In the next step all enabled transitions in Petri nets defining scenarios are fired. Preparation of transitions requires calculation of guards that are conjunctions of unary and binary predicates.
3. In order to obtain weights of predicates, appropriate queries are made to the ontology. If a predicate value for a certain binding was evaluated earlier, it is immediately returned.
4. Otherwise, appropriate *evaluators* assigned to predicates are called and returned values are asserted in the ontology as weights of fuzzy relations.

[1] http://www.simpoz.pl.

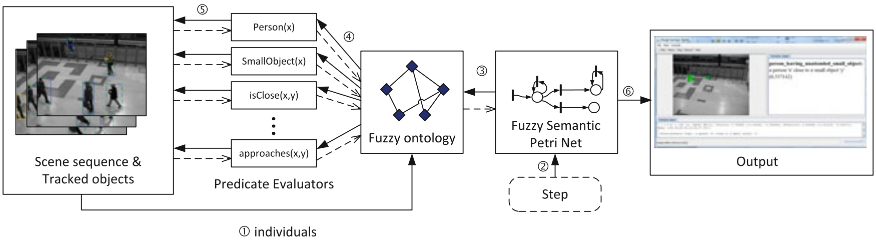


Fig. 6 Architecture of the event recognition system

5. Evaluators read data available at input, they analyze such features as location or size, but also estimated state variables, e.g., velocity.
6. After the net state is updated, the reached marking is analyzed. If a token stays in one of the indicated places long enough (the period is defined by the lower bound parameter assigned to the place), its presence is reported as a recognized scenario step (or its final stage).

To facilitate the evaluation of a FSPN at the *design* time, the framework collects analytic information related to weights of tokens and their flows. This information helps to evaluate the recognition capabilities of the defined FSPN and implemented evaluators. We will discuss this with an example given by the formula (2) and corresponding Petri net in Fig. 5.

Figure 7 presents in form of a Gannt chart weights of tokens assigned to net places at consecutive frames. For the purpose of presentation their values were shifted by adding 2, 4, 6, and 8 for tokens in *move*, *front*, *appear*, and *remain*. Hence, each elevation above a baseline represents a subevent occurrence. The expected and successfully recognized event occurrence is accomplished within the frames 286–316. It can be observed that compound subevents partly overlap, moreover, multiple transitions (marked with gray arrows) occur. Subevents *init*, *front*, *appear*, and *remain* are stable, both as regards duration and amplitude. The presented time series exhibit interesting

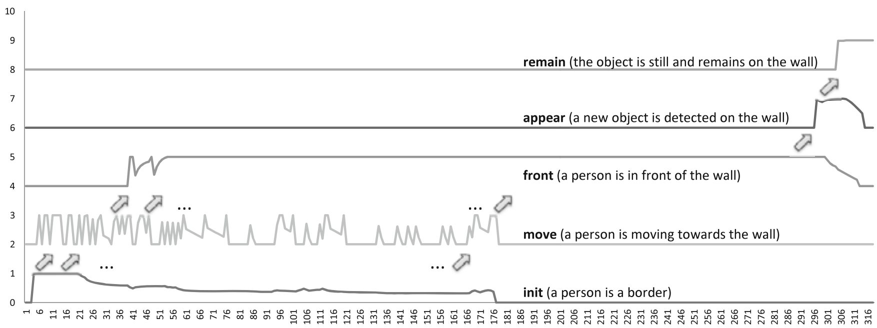


Fig. 7 Weights of tokens assigned to places at consecutive frames

property related to reusing of ontology and predicate *evaluators*. The *movesTowards* evaluator that was used in specification of *move* event (see formula 2) was actually prepared for other experiment, during which people violating a virtual zone on a floor were detected. As it can be noticed, the evaluator is inappropriate in the case of interactions with vertical objects, because it cannot produce stable events. In the further development, it was replaced by a new corrected implementation: *movesTowardsVerticalObject*.

Analogous experiments were conducted for several event recognition tasks including abandoned luggage and detecting violation of a surveillance zone. Tests for the abandoned luggage and graffiti painting events yielded 100 % correct results (true positives). For a zone violation, the recognition ratio was about 76 %. Detailed analysis revealed that in this case the lower performance was due to tracking problems (lost of identity in case of occlusion and in some cases invalid segmentation).

7 Conclusions

In this paper, we address the problem of modeling and recognition of video events. To summarize our contribution: first, we propose to apply a temporal logic formalism to specify event scenarios and further to translate them to Petri net structures; second, we introduce Fuzzy Semantic Petri Nets; finally, we describe a proof of concept prototype system that interprets a data resulting from a tracking algorithm, represents it as a content of a fuzzy ontology, and detects event occurrences with a FSPN interpreter. An advantage of FSPN is their capability of detecting concurrently occurring events, in which participate various combinations of objects, analyze scenario alternatives, and their likelihoods. Petri nets state (marking) gives general overview of a scenario stage. A presence of a token in a place can be reported as *semantic output*, e.g., to a surveillance system operator.

Acknowledgments This work is supported from AGH University of Science and Technology under Grant No. 11.11.120.859.

References

1. Aggarwal, J., Park, S.: Human motion: modeling and recognition of actions and interactions. In: Proceedings of the 2nd International Symposium on 3D Data Processing, Visualization and Transmission, 2004. 3DPVT 2004, pp. 640–647 Sept 2004
2. Aggarwal, J., Ryoo, M.S.: Human activity analysis: a review. ACM Comput. Surv. (CSUR) **43**(3), 16 (2011)
3. Akdemir, U., Turaga, P., Chellappa, R.: An ontology based approach for activity recognition from video. In: Proceedings of the 16th ACM international conference on Multimedia, pp. 709–712. ACM (2008)

4. Albanese, M., Chellappa, R., Moscato, V., Picariello, A., Subrahmanian, V.S., Turaga, P., Udrea, O.: A constrained probabilistic Petri net framework for human activity detection in video. IEEE Trans. Multimedia **10**(8), 1429–1443 (2008)
5. Barnard, M., Odobez, J.M., Bengio, S.: Multi-modal audio-visual event recognition for football analysis. In: IEEE 13th Workshop on Neural Networks for Signal Processing, 2003. NNSP'03. 2003, pp. 469–478, Sept 2003
6. Borzin, A., Rivlin, E., Rudzsky, M.: Surveillance event interpretation using generalized stochastic Petri nets. In: Eighth International Workshop on Image Analysis for Multimedia Interactive Services, 2007. WIAMIS'07, pp. 4–4. IEEE (2007)
7. Brémond, F., Thonnat, M., Zúniga, M.: Video-understanding framework for automatic behavior recognition. Behav. Res. Methods **38**(3), 416–426 (2006)
8. Büchi, J.R.: On a decision method in restricted second-order arithmetic. In: International Congress on Logic, Methodology, and Philosophy of Science, pp. 1–11. Stanford University Press (1962)
9. Ghanem, N., DeMenthon, D., Doermann, D., Davis, L.: Representation and recognition of events in surveillance video using Petri nets. In: Conference on Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04, pp. 112–112, June 2004
10. Joo, S.W., Chellappa, R.: Attribute grammar-based event recognition and anomaly detection. In: Conference on Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06, pp. 107–107, June 2006
11. Kripke, S.: Semantical considerations on modal logic. Acta philosophica fennica **16**(1963), 83–94 (1963)
12. Lavee, G., Rivlin, E., Rudzsky, M.: Understanding video events: a survey of methods for automatic interpretation of semantic occurrences in video. IEEE Trans. Syst. Man Cybern. Part C Appl. Rev. **39**(5), 489–504 (2009)
13. Lavee, G., Rudzsky, M., Rivlin, E., Borzin, A.: Video event modeling and recognition in generalized stochastic Petri nets. IEEE Trans. Circuits Syst. Video Technol. **20**(1), 102–118 (2010)
14. Lukasiewicz, T., Straccia, U.: Managing uncertainty and vagueness in description logics for the semantic web. Web Semant. Sci. Serv. Agents World Wide Web **6**(4), 291–308 (2008)
15. Manna, Z., Pnueli, A.: Temporal logic. In: The Temporal Logic of Reactive and Concurrent Systems, pp. 179–273. Springer, New York (1992)
16. Munch, D., Jsselmuiden, J., Arens, M., Stiefelhagen, R.: High-level situation recognition using fuzzy metric temporal logic, case studies in surveillance and smart environments. In: IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 882–889 Nov 2011
17. Nagel, H.H.: Steps toward a cognitive vision system. AI Mag. **25**(2), 31 (2004)
18. Szwed, P.: Video event recognition with Fuzzy Semantic Petri Nets. In: Gruca, D.A., Czachórski, T., Kozielski, S. (eds.) Man-Machine Interactions 3, Advances inIntelligent Systems and Computing, vol. 242, pp. 431–439. Springer International Publishing (2014)
19. Szwed, P., Komorkiewicz, M.: Object tracking and video event recognition with Fuzzy Semantic Petri Nets. In: Ganzha, M., Maciaszek, L.A., Paprzycki, M. (eds.) FedCSIS, pp. 167–174 (2013)

Self-localization and Navigation in Dynamic Search Hierarchy for Video Retrieval Interface

Tomoko Kajiyama and Shin'ichi Satoh

Abstract We have improved on the previously proposed graphical search interface, 'Revolving Cube Show,' for multi-faceted metadata. This interface can treat discrete, continuous, and hierarchical attributes, enabling users to search flexibly and intuitively by using simple operations to combine attributes. We have added two functions—one for displaying a search hierarchy as a guide tree and one for moving to a specific position in the search hierarchy, to solve problems identified through user testing. We created a video retrieval application using the improved interface for the iPad and tested it using data for 10,352 Japanese TV programs. The improved interface enabled users to easily understand the current position in the overall hierarchy and to quickly change specific attribute values.

Keywords Multi-faceted navigation · Dynamic hierarchy · Video collection · Interactive visualization · iPad application

1 Introduction

A graphical search interface called *Revolving Cube Show* (*Cube* for short) for multi-faceted metadata has been proposed [10]. *Cube* can treat discrete, continuous, and hierarchical attributes, enabling users to search flexibly and intuitively by combining attributes with simple operations. A video navigation system implementing this interface for use with Japanese TV programs has been developed; it

T. Kajiyama (✉)
Aoyama Gakuin University, 5-10-1 Fuchinobe, Chuo-ku, Sagamihara-shi,
Kanagawa 252-5258, Japan
e-mail: tomo@ise.aoyama.ac.jp

S. Satoh
National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku,
Tokyo 101-8430, Japan
e-mail: satoh@nii.ac.jp

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_39

supports five attributes: channel, time zone, performer, genre, and airtime. Users were able to search intuitively by creating dynamic search hierarchies. However, *Cube* had two problems: (i) a user sometimes, misunderstood the current position in the overall hierarchy because *Cube* displays only the previous and next values, and (ii) a user had to change the attribute values in the order specified by the system. That is, the user could not specify an attribute other than the previous or next attribute as a search key. To solve these problems, we added two functions: one for displaying a search hierarchy as a guide tree and one for moving to a specific position in the search hierarchy.

2 Related Work

Evaluation has shown that users sometimes select the wrong facets in category search [3], so that multi-faceted navigation in which users select and combine facets to create dynamic hierarchies is considered one solution [6]. Several faceted navigation approaches have also been reported: using questions to expand the user's thinking [4], combining the facets the user wants to use for searching by buttons or category names [6, 7], and using a faceted browsing interface for larger datasets [14]. Previous works are focused on how to specify queries, e.g., using icons representing facet names clearly for children [9], showing category hierarchy in 3D [5], showing the number of retrieved results for the next level of facets [14] and so on. One of the general purposes of using faceted navigations is to support users with ambiguous information needs. For this situation, it is important to provide functions for users to continue searching intuitively without concentrating on specifying queries.

A lot of research is undertaken for video navigation interfaces and it basically falls into two topics: how to specify queries and how to display videos. In the former topic, methods for supporting non-text search [2], using annotation or labels for browsing [4, 12] and providing faceted search [13] are proposed. This research uses many different types of operations, e.g., selecting facets by adding panels or clicking a button, selecting attribute values by using dropdown box and so on. In the latter topic, methods for layout and quantity of displaying videos are evaluated [8]. To support users with ambiguous information needs and treat multi-faceted hierarchical data, navigation interfaces should provide unity among operations for each function and display retrieved results with no obligation for browsing.

3 Revolving Cube Show

3.1 Overview of the Basic Interface

Cube provides users with three capabilities in particular: (i) users can combine the various kinds of attributes (discrete, continuous, and hierarchial) in one operation and create dynamic search hierarchies, (ii) users can browse the next set of retrieved results or the results retrieved for another set of attribute values, and (iii) users can quickly recognize the current search conditions. We constructed a video navigation system implementing *Cube* [11].

Figure 1 shows a screenshot of Japanese TV programs and five attributes: channel, time zone, performer, genre, and airtime. The five attributes are shown as faceted navigation tabs at the top. When the user selects an attribute as a search facet, a corresponding colored key frame appears in the window. In the example shown in Fig. 1, the outer key frame is for airtime, and the inner one is for genre. The retrieved results are displayed in real time inside the frames. The attribute values used as the current search keys are shown at the bottom center of the frame, and the previous and next attribute values are shown respectively, at the bottom left and right of the frame. Users can browse retrieved results for the next/previous attribute value at the same level by flicking the right/left side of the frame to the left/right and for the next lower/upper attribute value in the hierarchy by flicking the top/bottom of the frame down/upward. They can continue to search by flicking

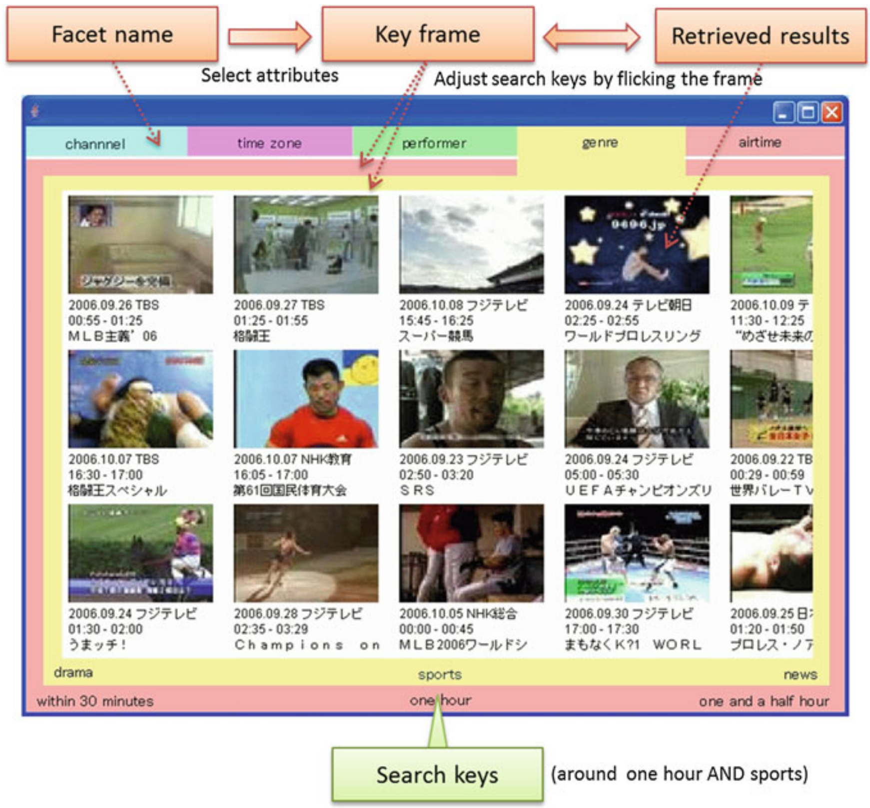


Fig. 1 An overview of *Cube* interface

the frame like moving on a cube surface and browse the next candidates by scrolling the retrieved results. *Cube* supports AND search by enabling users to specify multiple key frames. In the example shown in Fig. 1, the search keys are ‘around 1 h’ and ‘sports’ videos.

3.2 New Functions

We tested this video navigation system with 15 people and identified two problems: displaying the search hierarchy and adjusting the attribute value used as a search key. The user had to mentally visualize the overall search hierarchy because only the previous and next values were displayed in the bottom corners of the corresponding frame. It was, thus difficult for users to understand their current position in the hierarchy. Moreover, a user did not know which attribute values would come next when moving to a lower level of attributes. While *Cube* provided seamless display of retrieved results for an attribute value at the same level, users had to browse all retrieved results between the present attribute values and the attribute values defined by the system because the operation for changing attribute values was simply frame flicking. This prevented users from quickly browsing relevant retrieved results. To solve the problems, we added two functions: one for displaying a search hierarchy as a guide tree and one for moving to a specific position in the search hierarchy. Figures 2 and 3 show screenshots of a video retrieval application with our improved interface.

3.2.1 Displaying Search Hierarchy

When the user changes the search keys, a guide tree shown in the screenshot at the bottom of Fig. 2 is displayed at the right top of the window for 2 s To facilitate understanding of the search hierarchy, attributes on the same level are displayed with the same width, which is greater than that of the attributes at lower levels. The two previous and the two subsequent attribute values are also shown. If the attribute value specified previously also had attribute values, the attribute value obtained by flicking the frame downward and two previous and two subsequent values for that attribute are also displayed. Note that we also changed the names of two attributes for clarity: ‘time zone’ is now ‘start time’ and ‘airtime’ is now ‘duration.’

The algorithm for creating a guide tree for self-localization was as follows.

(1) The attribute for the outer frame are shown at the top center of the tree and displayed in green.
(2) The attribute value of its attribute is vertically aligned under the attribute and shown in pink.
(3) Two previous and two subsequent values of the attributes are horizontally aligned next to the attribute value displayed in step (2) and shown in brown.

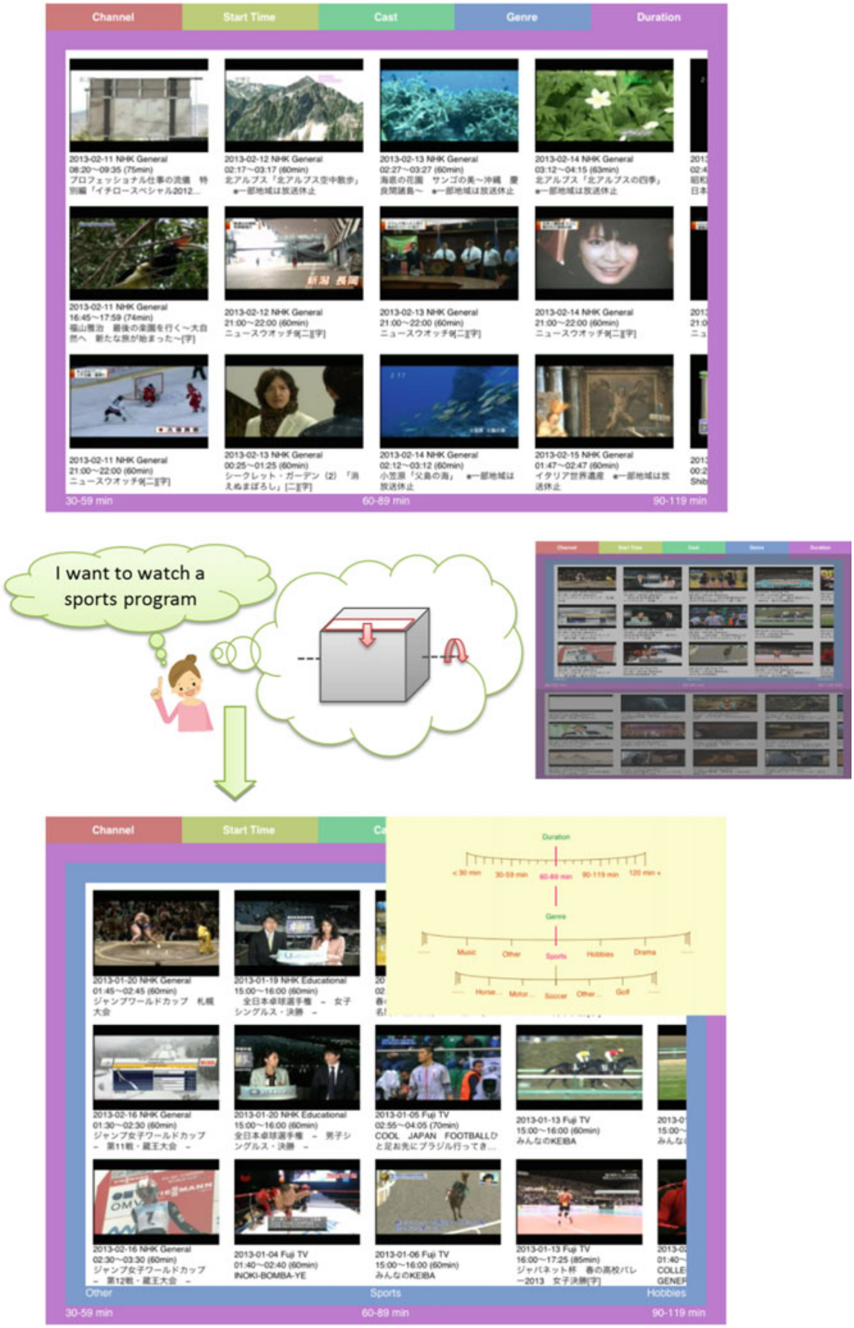


Fig. 2 Function for displaying a search hierarchy

The first attribute value is treated as the next attribute values of the last attribute value so that a user can browse retrieved results in the same direction.

(4) The links between the attribute and attribute values are displayed in brown. If the attribute displayed in step (1) is continuous, four more lines are shown like a ruler.
(5) If there is another attribute which is not displayed yet as search keys, the attribute is vertically aligned under the attribute value displayed in step (2) and go to step (2).
(6) If the attribute value for the most inner frame is displayed in step (1) and has subgenre, the attribute value obtained by flicking the frame downward and two previous and two subsequent values for that attribute are displayed with the same way described above.

Figure 2 shows the status when a user wants to watch a sports program between 60 and 89 min, and taps 'genre'. The search keys are 'duration: between 60 and 89 min and the first attribute value of the 'genre', 'sports' program, and retrieved results are displayed. While changing search keys, the frame for 'genre' appears inside the window like rotating a cube downward. The user could understand the current position by seeing a guide tree that appeared at the right top of the window after stopping animation.

The last search key specified by the user, 'sports,' has subgenres such as 'soccer' and 'golf'. Because not only the two previous and the two subsequent attribute values for the selected facet but also the attribute vales obtained by flicking the frame downward such as 'soccer' and 'golf' were displayed on the guide tree, the user could know that the 'sports' has subgenre and the part of candidates specified as next search key.

In a previous search interface [10], it was difficult for a user to understand the current position in search hierarchy and a user could not know whether the search facet was of a hierarchy attribute. In a usability test with one participant, she understood how to add search facets; however, she could not understand the transition of her search hierarchy changing. The following were her actions in the test.

(1) She wanted to watch a short local drama program.
(2) She selected 'duration' and specified 'within 30 min'.
(3) She selected 'genre', specified 'drama' and browsed retrieved results.
(4) She thought that she had to browse only local drama and continued to flick the frame of 'genre'.
(5) She noticed that search keys changed full circle.
(6) She gave up searching a short local drama program and changed her information needs to search a short music program.
(7) She flicked the frame of 'genre' eight times and browsed retrieved results.
(8) She decided on one program to watch.

To search for her relevant drama program, she had to specify 'duration: within 30 min' and 'genre: 'local' under 'drama'.' However, she could not understand the current position in the hierarchy and did not notice that 'drama' had subgenres such as 'local', 'period', and 'foreign.' If she would use our improved interface, she might understand the current position in the hierarchy by a guide tree and notice that 'drama' had subgenre 'local' in the step (4). She might not give up searching for relevant short local programs. This function enabled users to understand the current position in the search hierarchy.

3.2.2 Moving to a Specific Position

The user can also see the whole search hierarchy by performing a single tap with two fingers. This causes all attribute values at the same level of the search keys to be displayed full screen. In other words, the abbreviated attribute values in Fig. 2 are displayed. Users can then directly move to a specific position in the search hierarchy by tapping the name of the attribute value. The structure of the search hierarchy was kept and only the tapped attribute value was changed. If the attribute value for the most inner frame was tapped and had subgenre, the bottom part of the search hierarchy displayed as a guide tree would change.

In the example shown in Fig. 3, the user wanted to watch 'movie' programs when seeing a whole search hierarchy whose search keys were 'duration: over 120 min', 'genre: sports' and 'start time: late evening', and tapped 'movie' which is one of the 'genre' attributes. The search keys in the screenshot at the bottom of Fig. 3 were 'duration: over 120 min', 'genre: movie' and 'start time: late evening'.

In the step (7) of the usability test described in Sect. 3.2.1, she flicked the frame of 'genre' many times to specify 'music' as search keys. Using our improved interface, she could browse for short music programs by just tapping the screen with two fingers and selecting 'music.' This function enabled users to change the attribute values independently of the order specified by the system.

4 Video Retrieval Application

4.1 Database

We constructed a video retrieval application implementing the interface described in Sect. 3 for use on an iPad. The data was for 13,565 Japanese TV programs for 45 days, and each program had five attributes: channel, start time, cast, genre, and duration. We selected all attributes obtained by TV program listing except for the program information. Channel was a discrete attribute with seven possible values. Start time and duration were continuous attributes divided by inherent hierarchical values, e.g., "early morning" was defined between 4 and 7 a.m. for the start time attribute. Cast and genre were discrete attributes with inherent hierarchical values, e.g., 'sports' comprises baseball, football, golf, etc.

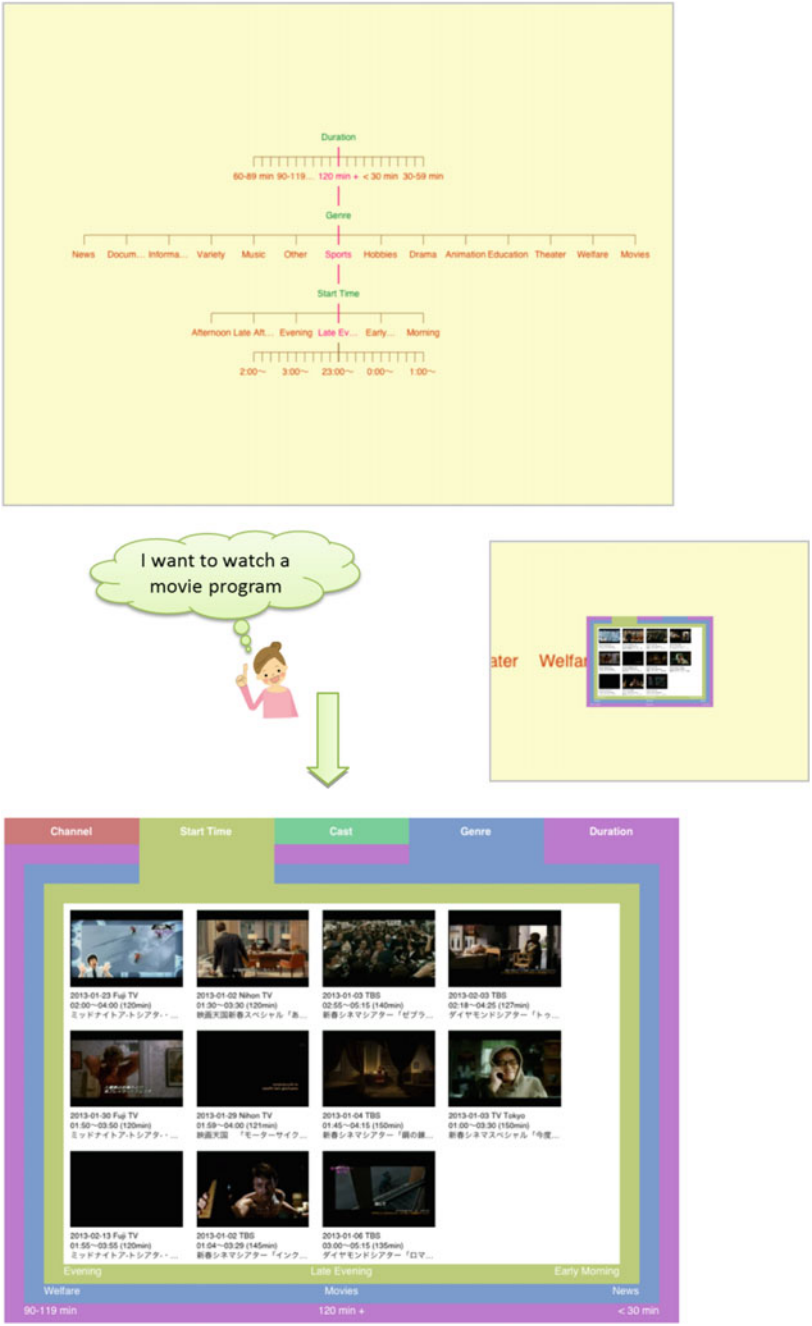


Fig. 3 Function for moving to a specific position

4.2 Features of Implementation

With *Cube*, users can continue to search a given facet by flicking the right or left side of the frame because the first attribute values of the facet appear after the last ones. For natural interaction, we defined the shift for a hierarchy attribute based on discrete data differently than that for one based on continuous data. That is, if the facet being searched is a hierarchy attribute based on discrete data such as performer and genre, one cycle of flicking the right or left side of the frame would result in the display of all retrieved results for only the upper level of the attribute. For example, if the search hierarchy was 'genre: sports—golf,' the attribute values in the upper level, 'sports,' would be fixed, and the user could browse the retrieved results for the attribute values on the same level as 'golf.'

In contrast, if the facet being searched is a hierarchy attribute based on continuous data such as start time and duration, the first attribute values for the next attribute in the upper level would appear after the last attribute values of the present attribute in the upper level. For example, if the search hierarchy was 'start time: early morning—7 am,' 'start time: morning—8 am' would appear after the user flicked the right side of the frame to the left. Although several faceted navigation approaches have been reported [1, 13], they do not take into account the features of each attribute value.

5 Conclusion

We improved 'Revolving Cube Show' graphical search interface on the basis of user testing. We added two functions, one for displaying a search hierarchy as a guide tree and one for moving to a specific position in the search hierarchy. We created a video retrieval application for the iPad that takes into account attribute features and tested it using data for 10,352 Japanese TV programs. The improved interface enabled users to easily understand the current position in the overall hierarchy and to quickly change attribute values.

References

1. Burke, R.D., Hammond, K.J., Young, B.C.: Knowledge-based navigation of complex information spaces. In: Proceedings of the 13th National Conference on Artificial Intelligence, pp. 462–468 (1996)
2. Christel, M.G.: Establishing the utility of non-text search for news video retrieval with real world users. In: Proceedings of ACMMM '07, pp. 707–716 (2007)
3. Debowski, S.: Wrong way: go back! An exploration of novice search behaviors while conducting an information search. Electron. Libr. **19**(60), 371–382 (2001)
4. Gordon, A.S.: Using annotated video as an information retrieval interface. In: Proceedings of the 5th International Conference on Intelligent User Interfaces (IUI '00), pp. 133–140 (2008)

5. Hearst, M.A., Karadi, C.: Cat-a-cone: an interactive interface for specifying searches and viewing retrieval results using a large category hierarchy. In: Proceedings of SIGIR'97, pp. 246−255 (1997)
6. Hearst, M.A.: clustering versus faceted categories for information exploration. Commun. ACM **49**(4), 59–61 (2006)
7. Hearst, M.A.: UIs for faceted navigation: recent advances and remaining open problems. In: Proceedings of the 2nd Workshop on Human-Computer Interaction and Information Retrieval, HCIR 2008, pp. 13−27 (2008)
8. Hürst, W., Darzentas, D.: Quantity versus quality: the role of layout and interaction complexity in thumbnail-based video retrieval interfaces. In: Proceedings of ICMR '12, Article 45, 8 pp (2012)
9. Hutchinson, H.B., Bederson, B.B., Druin, A.: The evolution of the international children's digital library searching and browsing interface. In: Proceedings of the 2006 Conference on Interaction Design and Children, pp. 105–112 (2006)
10. Kajiyama, T., Satoh, S.: A video navigation interface using multi-faceted search hierarchies. In: Proceedings of ACM Multimedia Systems 2013, MMSys'13, 5 pp (2013)
11. Kajiyama, T., Satoh, S.: Selflocalization and navigation in dynamic search hierarchy for video retrieval interface. In: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems (KICSS2013), pp. 291−297 (2013)
12. Li, J., Wang, Z., Wang, D., Zhang, B.: Interactive video retrieval with rich features and friendly interface. In: Proceedings of CIVR '08, pp. 567−568 (2008)
13. Villa, R., Gildea, N., Jose, J.M.: A faceted interface for multimedia search. In.: Proceedings of SIGIR '08, pp. 775−776 (2008)
14. Zhang, J., Marchionini, G.: Evaluation and evolution of a browse and search interface: relation browser ++. In: Proceedings of the 2005 National Conference on Digital Government Research, pp. 179–188 (2005)

On the Application of Orthogonal Series Density Estimation for Image Classification Based on Feature Description

Piotr Duda, Maciej Jaworski, Lena Pietruczuk, Marcin Korytkowski, Marcin Gabryel and Rafał Scherer

Abstract This paper presents an image classification algorithm called density-based classifier. The proposed method puts together the image representation based on keypoints and the estimation of the probability density of descriptors with the application of orthonormal series. For each class of images a separate classifier is constructed. The presented procedure ensures that different descriptors affect the final decision in a different degree. The trained classifier determines whether the query image is assigned to the class or not. The obtained experimental results show that proposed method provides good results. The algorithm can be applied to many tasks in the field of image processing.

Keywords Content-based image retrieval · Image classification

P. Duda · M. Jaworski · L. Pietruczuk · M. Korytkowski · M. Gabryel · R. Scherer (✉)
Institute of Computational Intelligence, Czestochowa University of Technology, Czestochowa, Poland
e-mail: rafal.scherer@iisi.pcz.pl

P. Duda
e-mail: piotr.duda@iisi.pcz.pl

M. Jaworski
e-mail: maciej.jaworski@iisi.pcz.pl

L. Pietruczuk
e-mail: lena.pietruczuk@iisi.pcz.pl

M. Korytkowski
e-mail: marcin.korytkowski@iisi.pcz.pl

M. Gabryel
e-mail: marcin.gabryel@iisi.pcz.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_40

1 Introduction and Motivation

Image processing is a very fast expanding field of knowledge; however, there is still a lot of open problems to be solved [5]. Image classification seems to be one of the most important aspects, which still needs to be improved and other approaches should be searched. The most important steps in every image classification system are

- selection of training samples
- image description
- selection of suitable classification approaches
- accuracy assessment

Choice of a method to execute above steps depends on a few conditions, first: user's needs. User should determine type of query before selecting type of classifier. Different approach should be applied to find similar image, other to point an object in an image, and another to indicate a class of object, etc. Second: scale and properties of the image. The images comparability is especially important in the step of image description. For example, higher resolution of the image can cause generating higher number of descriptors, what according to a problem can be advantage or drawback. The last but not least important problems are cost and time constraints. The system should try to ensure the best satisfaction of all conditions.

Novelty presented in this paper is construction of the classifier based on probability density estimation. The proposed algorithm is dedicated for image description implemented by feature descriptors. It is commonly accepted that a good descriptor should possess the ability to deal with many image transformations like rotation, intensity, scale and affine variations to some extent. Well known methods to generate feature descriptors are the Scale-Invariant Feature Transform (SIFT) [17] and the Speeded Up Robust Features (SURF) [1]. The descriptor in both causes is the d-dimensional point which is characteristic for part of the image.

In the proposed method, for each of the dimensions of the descriptor the density estimator of all instances from the training set ($\hat{p}(\cdot)$) is computed and the density estimator of the selected class instances ($\hat{p}_c(\cdot)$) is calculated

$$\hat{R}(x) = k\frac{\hat{p}_c(x)}{\hat{p}(x)}, \tag{1}$$

where x is a value of descriptor from one of the dimensions, and some $k \in R$. The ratio $R(\cdot)$ is then used to determine whether the image should be assigned to the particular class.

The proposed classifier solves only the two-class problem. It allows saying whether a given class can be considered or not. This approach appears to be justified through the construction of the training set. For example, image with a class "lion" assigned to it should preferably demonstrate the membership to classes: "lions," "felidae" (cats), and "animals." Considering multiclass problem it is sufficient to create a separate classifier for each of the classes.

The rest of the paper is organized as follows. Section 2 is divided into two sub-sections and describes well-known works in the fields of image classification and feature descriptors. Section 3 presents the mathematical basis used for density estimation. In Sect. 4 the proposed algorithm is described. Section 5 shows the experimental results and Sect. 6 contains the conclusions and future work. An earlier version of this paper was presented at the International Conference KICSS'2013 [6].

2 Related Works

2.1 Image Classification

The goal of image classification is to find and identify objects in the image and to assign a class or classes based on these objects. It is a problem well-known and considered in the literature. Developed methods were applied to many real world problems. The most popular tasks are processing of remotely sensed satellite images [18, 40, 42] or medical images [13, 14, 21].

Researchers try to solve the problem using a variety of tools such as decision trees [12], neural networks [16, 22], SVM [3, 44], fuzzy sets [20], boosting [15] etc. The investigation of the whole topic is out of the scope of this paper. Only a few important papers will be recalled.

In [7], a method for learning classes from small number of training examples using Bayesian framework is presented. In this framework first a probability density function is obtained and object category is assigned by updating this function based on new image or images. Bayesian networks were also investigated in [41]. The performance of these networks was improved by merging naive Bayesian networks and creating a complete network. As the result computational cost was reduced and the accuracy of classification increased.

In [39], authors investigated the problem of classifying the images with different depicted styles. In their method a structure is transformed into a feature vector based on spectral graph analysis of a hierarchical description of an image. In [8] authors investigated the challenges of reduction of required number of training examples and analyzing data that are potentially erroneous. They also raised the matter of supervised versus unsupervised training methods. They proposed a novel approach of combining those methods. In their algorithm first the unsupervised visual pruning is obtained by using unified and scale-invariant object representation and next a cross-modal learning scheme is applied.

2.2 Feature Selection

In order to facilitate the task of image classification the images are described in terms of some characteristic features. For example, in the method called the Histograms of Oriented Gradients (HOG) [4] the images are partitioned into small segments (cells). In each cell the distribution of gradient directions or edge orientations is determined. It is expressed in the form of histograms. The combination of histograms from all cells is the feature vector (descriptor) of the image.

Another approach is presented in [2]. The authors propose a shape context descriptor. In this method each shape in the image is described by a set of discrete points. Shapes have to be extracted from the images beforehand, e.g. using segmentation techniques.

In literature, there are several methods of describing the images using local descriptors. Such approach is also applied in the presented paper. The most popular methods are SIFT [17] and SURF [1], which were mentioned already in the first section. In brief, they consist of two main components. First one is the keypoint detector. It is based on a blob detection method. In SIFT the Laplacian of Gaussian is used (approximated by the Difference of Gaussian operator). In SURF the Determinant of Hessian operator is applied. In both considered methods the keypoint can be defined as a local extremum of blob detector function. It should be noted that the local extrema are sought in the tree-dimensional space: two dimensions of the image and the third dimension denoting the scale of the operator. Once the keypoints are found in the image, a descriptor is composed for each of them. The neighborhood of the keypoint is partitioned into 16 cells (4×4). According to the SIFT method in each cell a histogram of gradients for 8 directions is computed. Thus the SIFT descriptor is a 128-dimensional vector, which is further normalized. In line with the SURF method in each cell the coefficients of 2-dimensional Haar wavelets are calculated.

There are a few algorithms in literature which are the upgrades of the SIFT method. In [43] the authors propose a method called PCA-SIFT. Instead of using the SIFT histograms the principal components analysis is applied to the normalized gradient patch. The PCA-SIFT descriptors are more compact and, as authors claim, more distinctive and more robust to various image transformations. In [19], the Gradient Location and Orientation Histogram (GLOH) descriptor is proposed. It is also based on SIFT descriptor, however, the number of spatial regions for the histograms is extended. The number of dimensions in descriptor is then lowered to 64 with the application of the PCA algorithm.

3 Density Approximation

In this paper, we apply nonparametric techniques for image classification. They work both in stationary [9, 11, 23–25, 30, 33, 36] and non-stationary environment [10, 27–29, 31, 32, 34, 35, 37, 38].

Consider the interval (a, b) and a basis (A) of linear space $L^2((a, b))$. An infinite sequence of functions $\{g_j(\cdot)\}_{j \in N}$ from A, such that $\|g_j(\cdot)\|_2 = 1$ are called orthonormal functions. Then, every function $\phi \in L^2((a, b))$ has representation

$$\phi(x) = \sum_{j=0}^{\infty} b_j g_j(x), \tag{2}$$

where $b_j = \int_a^b \phi(x) g_j(x) dx$.

There are many orthonormal series discussed in the literature. The most known are the Hermite orthonormal series in $L^2(-\infty, \infty)$ and the Fourier orthonormal series in $L^2(-\pi, \pi)$. The Hermite orthonormal system is defined as follows:

$$g_0(x) = \pi^{\frac{1}{4}} \exp \frac{-x^2}{2} \tag{3}$$

$$g_1(x) = -2^{\frac{1}{2}} x g_0(x) \tag{4}$$

$$g_j(x) = -(\frac{2}{j})^{\frac{1}{2}} x g_{j-1}(x) - (\frac{j-1}{j})^{\frac{1}{2}} x g_{j-2}(x), \tag{5}$$

for $j = 2, 3, \ldots$.

Let $X_1, \ldots, X_n$ be a sequence of i.i.d. random variables with unknown density $f(x)$. If function $f \in L^2((a, b))$ then $b_j = E[g_j(X)]$ for $j \in N$ and can be estimated by

$$\hat{b}_j = \frac{1}{n} \sum_{i=1}^{n} g_j(X_i). \tag{6}$$

Moreover, under some condition [26] density f can by estimated as

$$\hat{f}(x) = \frac{1}{n} \sum_{j=0}^{M} \sum_{i=1}^{n} g_j(X_i) g_j(x). \tag{7}$$

Application of this method permits the calculation of the density of the random variable at each point of the interval, while keeping in memory only $M + 1$ points $\hat{b}_j$, for $j = 0, \ldots, M$.

4 Density-Based Classification

Training dataset is a set of n descriptors. Each descriptor is the d-dimensional vector generated from image which belongs to one of K classes. Let $x_{c,l_c} = (x^1_{c,l_c}, \ldots, x^d_{c,l_c})$ denote the l_cth descriptor from the cth class, where $l_c = 1, \ldots, n_c$ and $c = 1, \ldots, K$.

Figure 1 presents the image from class "airplanes," with marked descriptors. One can see that descriptors were detected not only in the airplane but also in the background. The motivation of proposed algorithm is to increase the importance of the descriptors found in the object, reducing simultaneously the influence of descriptors found in the background.

4.1 Learning of the Classifier

At the very beginning the training dataset is divided into two subsets. The smaller one is called the validation set and it needs descriptors only from a few images from every class. The bigger one contains the rest of descriptors.

Let us introduce the function $\psi_c : R \to \{0, 1\}$, for any $\delta = 1, \ldots, d$

$$\psi_c(x^\delta_{m,l_m}) = \begin{cases} 1 & m = c, \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

Let x denotes one of the components of the vector x_{c,l_c}. For clarity index denoting the dimension is omitted. Let us define the following function

$$\hat{R}_c(x) = \frac{\sum_{j=0}^{M} a^c_j g_j(x)}{\sum_{j=0}^{M} \hat{b}_j g_j(x)}, \tag{9}$$

Fig. 1 An example image from class "airplanes" with descriptors

where $\{g_j(\cdot)\}_{n\in N}$ and $\hat{b}_j$ are as in Sect. 3 and a_j^c are computed as

$$a_j^c = \frac{1}{n}\sum_{i=1}^{n} g_j(X_i)\psi_c(X_i), \tag{10}$$

where X_i are descriptors from training set. One can see that Eq. (9) can be expressed in an equivalent form as

$$\hat{R}_c(x) = \frac{n_c}{n}\frac{\hat{p}_c(x)}{\hat{p}(x)}, \tag{11}$$

where n_c is the number of descriptors from class c in the training dataset. More about density estimation based on orthonormal series can be found in [26].

It is worth noting that if a descriptor is typical of a class and it appears only in this class, then the value of function $\hat{R}_c$ is large. If the point is rarely present in the class then the value of R is small.

This model must be created for each class separately. However, in memory the user has to store only $d((K+1)*(M+1)+K)$ parameters. For every class and every dimension $M+1$ parameters are needed for density estimation. Additional $M+1$ parameters are stored for density estimation of all data. The remaining dK parameters is required in the last stage of learning process. On a basis of the validation set the optimal value of parameter $\Omega \in [0, 1]$ is calculated. It is further discussed in the next subsection.

4.2 Classes Assignment

When a new image needs to be classified at first, the set of descriptors is generated from it. Next for every dimension of descriptor, value of function (9) is computed for the first class ($c = 1$). As a result values r_i, $i = 1, \ldots, d$, are obtained for each dimension separately. However, it is taken into account that in very special cases the function $\hat{R}_c$ can take unexpectedly high values. For convenience it is useful to keep the values of function $\hat{R}_c$ in interval $[0, 1]$. For this purpose it is checked if the value of $\hat{p}(x)$ is larger than some threshold ω. Estimator of density $\hat{p}(x)$ can take values arbitrarily close to zero if no point from training dataset was noticed in a neighborhood of x. Parameter ω can be the same for all classes and can be set by the user. Values which satisfy the ω-test are used to calculate the average $\overline{r}_l$, where l is the index of the descriptor, $l = 1, \ldots, N$. (N is a number of descriptors generated from query image).

Each descriptor from the considered image is processed by the classifier and receives its $\overline{r}_l$ value. The average of these values for all descriptors $\widetilde{r}$ is computed by formula (12) and compared with parameter Ω.

$$\widetilde{r} = \frac{1}{N}\sum_{l=0}^{N}\overline{r}_l \tag{12}$$

If this average value is larger than Ω then the considered image belongs to the first class, else it does not. We store the result and go on to the rest of the classes ($c = 2, \ldots, K$). One image can be assigned to one, many or none of the classes. Parameter Ω is set on a basis of the validation set. The aim is to provide the accuracy of classification as high as possible. It tries to ensure that all images from correct class will be assigned correctly and as few as possible images from the rest (wrong) classes will be assigned incorrectly.

5 Experimental Results

The experimental results were obtained on a basis of The Caltech-101 Object Categories dataset [7]. Authors wrote about this dataset: "Each object category contains about 40–800 images. Size of each image is roughly 300×200 pixels. All images are annotated with the following information: a bounding box of the object and a carefully traced silhouette of the objects by a human subject."

From the set of 101 categories, 10 pairs of classes were randomly selected. One class from every pair was chosen as selected to classify and the second as a collection of items for which the class is not distinguished. From every category 10 images were randomly selected as a test set and were excluded from the training set. The number of images for every class which were used in learning process is displayed in Table 1. About 10 % of images from training set were carried out as a validation set. For each image the key points were generated using SIFT or SURF algorithms, implemented in EmguCV library.

The SIFT algorithm was applied in 3 different settings. The default settings of the SIFT algorithm in EmguCV library (contrast th. = 0.04 and edge th. = 10) is denoted by $SIFT - d$. The SIFT algorithm with contrast th. set to 0.02 and edge th. set to 20 was labeled by $SIFT-$. As a result of such parameters settings a smaller

Table 1 Number of images

Class name	Distinguished class	Background class
Airplanes–sea-horse	790	47
Leopards–mandolin	190	33
Metronome–tick	66	39
Brontosaurus–okapi	32	29
Binocular–rooster	23	39
Emu–revolver	72	43
Ibis–pyramid	70	47
Pagoda–wild-cat	38	24
Sunflower–wrench	75	29
Ferry–pizza	57	44

number of points is generated than in the case of $SIFT - d$. Similarly by $SIFT+$ the SIFT algorithm with contrast th. set to 0.08 and edge th. set to 5 was labeled. Such parameters setting decreases number of generated keypoints. In every case a descriptor is a 128-dimensional vector.

The SURF algorithm was applied in 2 different settings. In both cases descriptor is a 64-dimensional vector. First, the contrast parameter was set to 300 (denoted by $SURF300$) and secondly it was set to 500 ($SURF500$).

The Hermite orthonormal series is applied to perform experiments. In the evaluation process, parameter ω was set to 0.1 and obtained accuracies for SIFT descriptors are shown in Table 2. Table 3 presents the obtained accuracies for SURF descriptors. As can be seen from the presented tables, the choice of the keypoint detection and description method influences significantly the final accuracy of the image classification. In most cases, the SURF algorithm outperforms the SIFT method.

Table 2 Accuracies for SIFT descriptors

Classes name	$SIFT - d$	SIFT+	SIFT−
Airplanes–sea-horse	0.75	0.65	0.65
Leopards–mandolin	0.75	0.75	0.6
Metronome–tick	0.8	0.85	0.7
Brontosaurus–okapi	0.7	0.75	0.7
Binocular–rooster	0.55	0.55	0.55
Emu–revolver	0.85	0.8	0.8
Ibis–pyramid	0.65	0.55	0.7
Pagoda–wild-cat	0.85	0.85	0.85
Sunflower–wrench	1	1	0.8
Ferry–pizza	0.75	0.55	0.85
Mean	0.765	0.73	0.72

Table 3 Accuracies for SURF descriptors

Classes name	$SURF500$	$SURF300$
Airplanes–sea-horse	0.75	0.7
Leopards–mandolin	0.85	0.95
Metronome–tick	0.85	0.7
Brontosaurus–okapi	0.5	0.9
Binocular–rooster	0.65	0.65
Emu–revolver	0.75	0.9
Ibis–pyramid	0.7	0.8
Pagoda–wild-cat	0.85	0.85
Sunflower–wrench	1	1
Ferry–pizza	0.85	0.6
Mean	0.775	0.805

6 Conclusions and Future Work

In this paper, the density-based image classification algorithm was introduced. It is used for verifying the membership of the image to the considered class. The obtained experimental results showed the usefulness of the presented solution. The big advantage of the proposed method is the low memory requirement.

The algorithm, however, leaves a lot of space for future consideration. The algorithm should be tested on a larger number of classes. A promising appears to be the extension of the algorithm functionality to other image processing operations. For example, the detection of images in a query image can be accomplished by the application of procedures like the pyramid representation. Also, it seems to be justified to search methods of keypoints detection which are dedicated for image classifier.

Acknowledgments The project was funded by the National Center for Science under decision number DEC-2011/01/D/ST6/06957.

References

1. Bay, H., Tuytelaars, T., Van Gool, L.: SURF: Speeded Up Robust Features. In: ECCV, pp. 404–417 (2006)
2. Belongie, S., Malik, J.: Matching with shape contexts. In: IEEE Workshop on Contentbased Access of Image and Video Libraries (CBAIVL-2000) (2000)
3. Chapelle, O., Haffner, P., Vapnik, V.N.: Support vector machines for histogram-based image classification. Trans. Neural Netw. **10**(5), 1055–1064 (1999)
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005 (2005)
5. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: ideas, influences, and trends of the new age. ACM Comput. Surv. **40**(2), 5:1–5:60 (2008)
6. Duda, P., Jaworski, M., Pietruczuk, L., Scherer, R., Korytkowski, M., Gabryel, M.: On the application of fourier series density estimation for image classification based on feature description. In: Skulimowski, A. (ed.) Looking into the Future of Creativity and Decision Support Systems: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems, Kraków, Poland, 7–9 Nov 2013. Advances in Decision Sciences and Future Studies, vol. 2. Progress and Business Publishers, Kraków (2013)
7. Fei-Fei, L., Fergus, R., Perona, P.: A Bayesian approach to unsupervised one-shot learning of object categories. In: Proceedings of the Ninth IEEE International Conference on Computer Vision, ICCV'03, vol. 2, pp. 1134. IEEE Computer Society, Washington, DC, USA (2003)
8. Fritz, M., Kruijff, G.J.M., Schiele, B.: Tutor-based learning of visual categories using different levels of supervision. Comput. Vis. Image Underst. **114**, 564–573 (2010)
9. Gałkowski, T., Rutkowski, L.: Nonparametric fitting of multivariable functions. IEEE Trans. Autom. Control **AC-31**, 785–787 (1986)
10. Greblicki, W., Rutkowska, D., Rutkowski, L.: An orthogonal series estimate of time-varying regression. Ann. Inst. Stat. Math. **35**(Part A), 215–228 (1983)
11. Greblicki, W., Rutkowski, L.: Density-free Bayes risk consistency of nonparametric pattern recognition procedures. Proc. IEEE **69**, 482–483 (1981)
12. Jiang, L., Wang, W., Yang, X., Xie, N., Cheng, Y.: Classification methods of remote sensing image based on decision tree technologies. In: Li, D., Liu, Y., Chen, Y. (eds.) CCTA (1). IFIP Advances in Information and Communication Technology, vol. 344, pp. 353–358. Springer (2010)

13. Kelly, P.M., Cannon, T.M.: Candid: Comparison algorithm for navigating digital image databases. In: Proceedings Seventh International Working Conference on Scientific and Statistical Database Management, pp. 252–258 (1994)
14. Keysers, D., Dahmen, J., Ney, H., Wein, B., Lehmann, T.: A statistical framework for model-based image retrieval in medical applications (2002)
15. Korytkowski, M., Rutkowski, L., Scherer, R.: On combining backpropagation with boosting. In: 2006 International Joint Conference on Neural Networks, IEEE World Congress on Computational Intelligence, Vancouver, BC, Canada, pp. 1274–1277. IEEE (2006)
16. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Bartlett, P.L., Pereira, F.C.N., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) NIPS, pp. 1106–1114 (2012)
17. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proceedings of the International Conference on Computer Vision, ICCV'99, vol. 2, pp. 1150. IEEE Computer Society, Washington, DC, USA (1999)
18. Maheshwary, P., Srivastava, N.: Prototype system for retrieval of remote sensing images based on color moment and gray level co-occurrence matrix (2009)
19. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Trans. Pattern Anal. Mach. Intell. **10**(27), 1615–1630 (2005)
20. Nowicki, R.: Rough-neuro-fuzzy system with MICOG defuzzification. In: 2006 IEEE International Conference on Fuzzy Systems, IEEE World Congress on Computational Intelligence, 16–21 July 2006, Vancouver, BC, Canada, pp. 1958–1965 (2006)
21. Orphanoudakis, S.C., Chronaki, C., Kostomanolakis, S.: I2C: a system for the indexing, storage, and retrieval of medical images by content. Med. Inf. (Lond.) **19**(2), 109–122 (1994)
22. Patan, K., Patan, M.: Optimal training strategies for locally recurrent neural networks. J. Artif. Intell. Soft Comput. Res. **1**(2), 103–114 (2011)
23. Rutkowski, L., Rafajłowicz, E.: On optimal global rate of convergence of some nonparametric identification procedures. IEEE Trans. Autom. Control **AC-34**(10), 1089–1091 (1989)
24. Rutkowski, L.: A general approach for nonparametric fitting of functions and their derivatives with applications to linear circuits identification. IEEE Trans. Circ. Syst. **CAS-33**, 812–818 (1986)
25. Rutkowski, L.: Identification of MISO nonlinear regressions in the presence of a wide class of disturbances. IEEE Trans. Inf. Theory **IT-37**, 214–216 (1991)
26. Rutkowski, L.: New Soft Computing Techniques for System Modeling, Pattern Classification and Image Processing. Springer (2004)
27. Rutkowski, L.: On Bayes risk consistent pattern recognition procedures in a quasi-stationary environment. IEEE Trans. Pattern Anal. Mach. Intell. **PAMI-4**(1), 84–87 (1982)
28. Rutkowski, L.: On nonparametric identification with prediction of time-varying systems. IEEE Trans. Autom. Control **AC-29**, 58–60 (1984)
29. Rutkowski, L.: On-line identification of time-varying systems by nonparametric techniques. IEEE Trans. Autom. Control **AC-27**, 228–230 (1982)
30. Rutkowski, L.: Sequential estimates of probability densities by orthogonal series and their application in pattern classification. IEEE Trans. Syst. Man Cybern. **SMC-10**(12), 918–920 (1980)
31. Rutkowski, L.: Nonparametric identification of quasi-stationary systems. Syst. Control Lett. **6**, 33–35 (1985)
32. Rutkowski, L.: The real-time identification of time-varying systems by nonparametric algorithms based on the parzen kernels. Int. J. Syst. Sci. **16**, 1123–1130 (1985)
33. Rutkowski, L.: Sequential pattern recognition procedures derived from multiple Fourier series. Pattern Recogn. Lett. **8**, 213–216 (1988)
34. Rutkowski, L.: An application of multiple Fourier series to identification of multivariable non-stationary systems. Int. J. Syst. Sci. **20**(10), 1993–2002 (1989)
35. Rutkowski, L.: Nonparametric learning algorithms in the time-varying environments. Signal Process. **18**, 129–137 (1989)

36. Rutkowski, L.: Multiple Fourier series procedures for extraction of nonlinear regressions from noisy data. IEEE Trans. Signal Process. **41**(10), 3062–3065 (1993)
37. Rutkowski, L.: Adaptive probabilistic neural-networks for pattern classification in time-varying environment. IEEE Trans. Neural Netw. **15**, 811–827 (2004)
38. Rutkowski, L.: Generalized regression neural networks in time-varying environment. IEEE Trans. Neural Netw. **15**, 576–596 (2004)
39. Shariat, S., Rabiee, H.R., Khansari, M.: Inferring a Bayesian network for content-based image classification. In: 13th International CSI Computer Conference, 09 March 2008. Kish Island, Iran (2008)
40. Soliman, O.S., Mahmoud, A.S., Hassan, S.M.: Remote sensing satellite images classification using support vector machine and particle swarm optimization. In: Proceedings of the 2012 Third International Conference on Innovations in Bio-Inspired Computing and Applications, IBICA'12, pp. 280–285. IEEE Computer Society (2012)
41. Xiao, B., Song, Y.Z., Hall, P.M.: Learning invariant structure for object identification by using graph methods. Comput. Vis. Image Underst. **115**(7), 1023–1031 (2011)
42. Xu, M., Wei, C.: Remotely sensed image classification by complex network eigenvalue and connected degree. In: Computational and Mathematical Methods in Medicine 2012 (2012)
43. Yan, K., Sukhtankar, R.: PCA-SIFT: a more distinctive representation for local image descriptors. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004, vol. 2, pp. 506–513 (2004)
44. Yaw, C., Yishi, W., Cuixain, C., Ricanek, K.: Optimal training strategies for locally recurrent neural networks. J. Artif. Intell. Soft Comput. Res. **1**(3), 241–253 (2011)

Fast Image Search by Trees of Keypoint Descriptors

Patryk Najgebauer and Rafał Scherer

Abstract In this paper, we present a method for fast image searching tree-based representation of local interest point descriptors. Thanks to decreased number of steps needed to perform the search, such a representation of image keypoints is more efficient than the standard, frequently used list representation where images are compared in all-to-all manner. The proposed method generates a tree structure from a set of image descriptors, e.g., generated by the SURF algorithm. The descriptors are stored as leaves in the tree structure and other parent tree nodes are used to group similar descriptors. Each next parent node of the tree forms a wider, more general, group of descriptors. We store average values of the descriptor components in the nodes making it possible to quickly compare sets of descriptors by traversing the tree from the root to a leaf by choosing the smallest deviation between searched descriptor and values of nodes. With each next step of tree traversing we reduce the final number of descriptors that will be needed to compare. The proposed structure also allows to compare whole trees of descriptors what can speed up the process of images comparison, as it involves generating trees of descriptors for single images or for groups of related images accelerating the process of searching for similarities among others.

Keywords Image processing · Keypoints comparing · Comparing trees

P. Najgebauer · R. Scherer (✉)
Institute of Computational Intelligence, Częstochowa University
of Technology, al. Armii Krajowej 36, 42-200 Częstochowa, Poland
e-mail: rafal.scherer@iisi.pcz.pl
URL: http://iisi.pcz.pl

P. Najgebauer
e-mail: patryk.najgebauer@iisi.pcz.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_41

1 Introduction

Recognizing and comparing images by their content is increasingly applied in many aspects of human life. These methods allow improving manifold human activities that tend to be monotonous and grudgingly performed. Growing popularity of these methods is mainly due to development and increasing efficiency of electronic devices. It is closely related because image processing methods are very demanding in terms of the required number of operations and the amount of data that need to be processed.

Most of the methods are specialized in narrow areas such as methods of face, fingerprint, or text recognition. They naively simulate a tight group of activities that they usually perform better and faster than humans. Most of the specialized methods are not suitable for use in a general-purpose image comparison.

A major problem of pattern recognition methods in the case of general-purpose applications, is the lack of knowledge and imagination that have every human. This is mainly due of the large differences in the way of image perception by humans and computers. Humans focus on the information that is carried by the image, they recognize and remember the names of objects and pay less attention to the details, trying to simplify and memorize generalized information. Humans in many cases are unable to recreate the memorized image from their memory. Computers perceive an image as a set of values that form individual pixels in the image and it is very difficult to extract important features of the image from such a set.

Several general-purpose algorithms determine image keypoints, which are areas that contrast to the rest of the image. Thanks to their application, we can omit most of the picture and focus only on the specific areas. However, they do not allow for semantic image recognition but they create a so-called point maps that describe image local gradients and allow for easy comparison of images to identify common areas.

Unfortunately, in most cases, these methods generate large amount of keypoints that contains only partial information about the image. Their large and changing amount in case of different images creates problems during the search for common parts in a larger group of images.

A problem also arises from the fact that some keypoints are not reproduced over identical images, for example with small differences of rotation. This problem makes impossible to create pre-determined relationship between keypoints for later image comparison.

For illustrative purposes, the proposed method utilizes the SURF algorithm [1] for detecting keypoints in images. It is a popular and one of the fastest general-purpose algorithms for keypoints generating. The main features of each generated keypoints are: position, orientation, size, and descriptor.

Algorithms for detecting keypoints use masks that define image local extreme typical for blob detection algorithm (Fig. 1) [6, 8]. Image is analyzed several times and in each step the size of the mask is increased. That creates some kind of a pyramid of masks (scales). This allows determining the keypoints regardless of their scale. Finally, the algorithm for each keypoint in reference to designated orientation generates a descriptor. Determination of keypoint orientation allows to generate the same descriptors regardless of the global orientation of the image. Descriptor is a representation of the local gradient over the keypoint, and is something of unique value. Basic descriptor generated by SURF algorithm contains 64 values. Each of subsequent group of four descriptor values forms a $\mathbf{V_{sub}}$ vector (1), where $\sum dx$ is sum of Haar mask in the x axis direction, $\sum dy$ is sum of Haar mask in the y axis direction, $\sum |dx|$ is sum of absolute values of Haar mask in the x axis direction and $\sum |dy|$ is sum of absolute values of Haar mask in the y axis direction. All of descriptors $\mathbf{V_{sub}}$ vectors create a matrix of 4×4 size that extends over the keypoint (Fig. 1). This solution allows to describe accurately the local gradient (Fig. 2).

$$\mathbf{V_{sub}} = \left[\sum \mathbf{dx}, \sum \mathbf{dy}, \sum |\mathbf{dx}|, \sum |\mathbf{dy}|\right] \tag{1}$$

An advantage of the algorithm is substantial speed-up relative the prior solutions, due to several improvements. In the previous solutions Gaussian mask was used that required repeated aggregation of all pixels with a given ratio. In the SURF algorithm, it is improved by using the integral image concept that allows determining quickly the sum of pixels in a given area. It also has some effect on the accuracy of designated

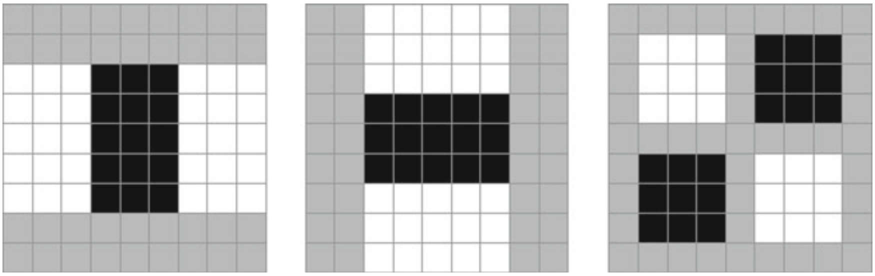


Fig. 1 SURF smallest 9×9 pixels blob detector

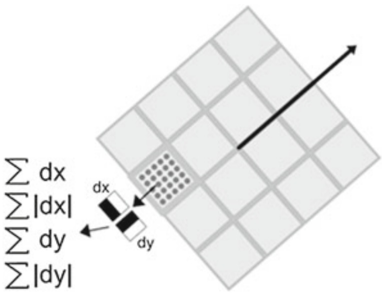


Fig. 2 Descriptor structure, 4×4 matrix of four values from a sub region

descriptors because the aggregation pool of pixels has squared shape, and any change in the rotation and the position of the keypoints causes a small deviation.

The paper is an extended version of [7] and in the next section we describe the problem, present the solution and numerical simulations.

2 Problem Description

Deviations that occur between similar keypoints force to use a variation tolerance between compared descriptors. For example, Table 1 shows a 4×4 matrix of deviations between $\mathbf{V_{sub}}$ values of descriptors of two similar keypoints. To illustrate the similarity of keypoints which descriptors were compared they are presented in Fig. 3. For humans, these points seem to be identical but the standard deviation between these keypoints descriptors is equal 0.4743. For this example it can be assumed that standard deviation does not exceed 0.5 for similar descriptors. Increasing value of tolerance also results in possibility to create groups of similar descriptors. First adopted and simplest solution for representation and indexing keypoints is to generate hash codes based on the descriptors values. This solution would be efficient to use together with indexing tools of databases. Unfortunately, the nature of construction and previously mentioned deviation between the descriptors of keypoints cause many problems. Most often problems in searching similar keypoints result from:

- descriptors on the boundary of two hash code values,
- random pick deviations of single values of descriptors,

Table 1 Deviations ($\Delta V_{sub}(x, y)$) between $\mathbf{V_{sub}}$ values of two similar keypoints

$(x\backslash y)$	1	2	3	4
1	0.0078	0.0042	0.0619	0.0120
2	0.0293	0.0176	0.0282	0.0103
3	0.0210	0.0306	0.0473	0.0027
4	0.0003	0.0298	0.0181	0.0678

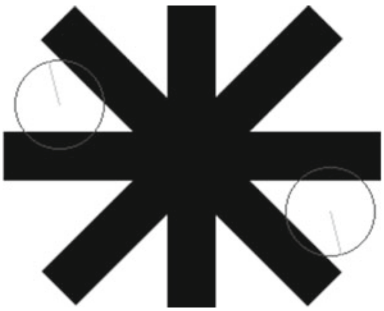


Fig. 3 SURF similar keypoints example with 0.47 value of difference between descriptor components

Fig. 4 Example of keypoints detection by the SURF algorithm

- deviations resulting from changes in the values of position, orientation and scale,
- distortion created from image noise.

During keypoints generation, a set of various amount of keypoints is created. It depends in most cases on the image size and the amount of details presented by the image. For example, the number of keypoints generated from a photo of the size about 1280×800 pixel in most cases exceeds 1000. Such a large number of keypoints creates a large number of descriptors combinations that should be considered in the process of images comparing. Figure 4 illustrate how many keypoints is generated by SURF algorithm from image of 374×369 pixels size. The simplest way to compare keypoints between images is all-to-all comparison, but with such a large number of points, this may require enormous computational complexity of the process. For example, two sets of the one thousand keypoints need one million comparisons. To reduce the number of required descriptors comparisons they can be ordered by sorting thus we can skip certain part of the descriptors in comparison.

3 Method Description

To speed up the process of comparing descriptors, the proposed method assumes that descriptors are represented by a structure based on a tree. Descriptors with similar values are grouped under shared nodes of the tree. Each node of the tree contains a simplified version of the descriptors which value is a sum of descriptors of its child elements. This arrangement allows for faster searching similar descriptors by

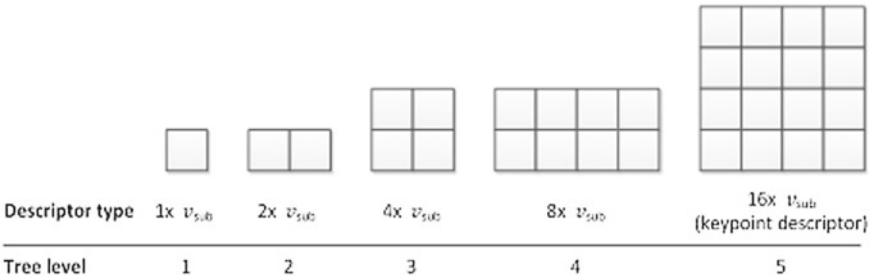


Fig. 5 Types of node descriptors in relations to node tree level

starting from the root node and by searching for the most similar descriptor traversing to the next level node. With each new level descriptors nodes provide more precise descriptor value and indicates a more narrow group of similar keypoints. Generated tree is a balanced tree that is characterized by

- Constant height of the tree that equals 5. All paths of generated tree from root to leaves have equal length that is equal to the height of the tree.
- Each non-leaf node has at least one child node. This is due to the fact that the descriptors of the keypoints are stored in the leaves.

Each node stores a descriptor of different length depending on its tree level. Types of descriptors for each level of nodes are illustrated in Fig. 5. For better illustration of descriptors types, descriptors are presented by using blocks where each block represents a single $\mathbf{V_{sub}}$ vector. Thus, descriptors of the first level nodes contains one $\mathbf{V_{sub}}$ vector of four floating point values as presented in (1). Descriptors of leaf nodes contains 16 $\mathbf{V_{sub}}$ vectors that is 64 floating point values which is equal to the length of keypoints descriptor. Leaf nodes in presented solution are the only nodes that store the keypoints data.

3.1 Creation of Descriptors Tree

In the process of creating the tree, for each descriptor an individual subtree is created that consists of a single path from the root to the leaf. The process of patch designation starts from the bottom where size of the nodes descriptor is largest. This lowest node of tree takes the exact copy of the descriptor and other data of keypoint. The next parent node we create contains a descriptor of reduced by half $\mathbf{V_{sub}}$ number of his previous created child node. Process of descriptor reduction is according to formula (2), where $\mathbf{v}_1(n)$ is value of previous nodes descriptor and $\mathbf{v}_2(n)$ are the values of the descriptor of the currently designed node

$$\mathbf{V}_2(n) = \mathbf{v}_1(2n) + \mathbf{v}_1(2n+1)\ . \tag{2}$$

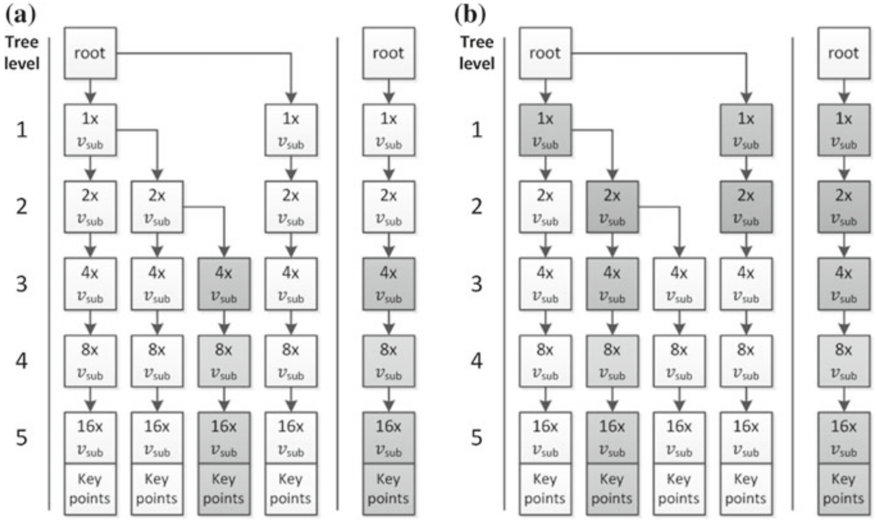


Fig. 6 Visualization of process of single descriptor adding to the tree **a** and descriptor comparison with the tree **b**

This process is repeated until the nodes descriptor does not contain a single $\mathbf{V_{sub}}$ element. Examples of subtree generated for a single keypoint are presented in Table 2. After creating a subtree from a keypoint, we insert this subtree to the main descriptors tree. This process is presented in Fig. 6a. During insertion both trees are compared. Only the nodes at the same tree level are compared beginning from the root. Comparison is performed by determining a standard deviation between descriptors of the compared nodes. Proposed method implies a margin of deviation between descriptors equal to 0.5. If the deviation is in the range of the threshold, then compared nodes are considered as identical and we proceed to compare their child nodes. If appropriate node from the main tree is not found, then the subtree node and its child nodes will be copied to the main tree (right side of figure presents subtree components marked by grey background that are copied to main tree on left side of figure) by creating a new branch. In a case when all subtree nodes are related with main tree then keypoints data from subtree is copied to the appropriate leaf in the main tree.

This approach allows to simultaneously store and group similar keypoints. Finally, the larger group of keypoints that descriptor values are similar can be stored in a common leaf of tree what even further accelerates the comparison process.

3.2 *Comparing Descriptors Tree*

Descriptors tree search can be done in two ways. First, by searching for individual descriptors. Second, by comparing with the other trees. In the case of searching single keypoints, the process is similar to the process of keypoint insertion as described in Sect. 3.1. Thus, in the first place a subtree for the searched descriptor is designated and then the subtree is compared with the main tree. This process is presented in Fig. 6b. Figure shows visualization of a descriptor tree on the left side and a subtree designed from the compared descriptor on the right. Only elements on same level are compared started from the root. Matched components are marked by grey background. Do not matched components are skipped with their branch. If all nodes of the subtree are consistent with the main tree, then the descriptor matches. The result will be a set of at least one keypoint stored in the tree.

To compare descriptor trees determined for individual images we have to determine a common part of both trees. In this case, a large amount of comparisons is omitted. Whole branches of the trees are omitted if they are not compatible between the compared trees. Some comparisons are also omitted because of the grouping keypoints to share in the tree nodes.Some comparisons are also omitted due to the keypoints grouping that share the same nodes in the tree structure. Comparison of pre-created descriptors tree is efficient for multiple images comparison such as in the case of similarity searching in a group of pictures. This is caused by omitting process of subtree creating for each keypoint.

The final effect is a list of common descriptors between images which can be easily analysed to check relationship between their orientation and location. It is very important because some of the keypoints are not repeated even between identical images. For this reason it cannot be determined a constant relationship between the keypoints of a single image. Relationships can be examined only after keypoints selection that correspond to both compared images.

4 Experimental Results

Experiments were performed on the test pictures presented in Fig. 7. To better illustrate the algorithm, test pictures are similar in order to obtain positive results of the search. Similarity of test pictures are changed in varying degrees in order to better present the differences of the algorithm performance. The essence of the experiment was to generate a tree from a list of keypoints of each image and then compare it with keypoints of other images. Results of the experiment are presented in tables. Table 2 shows the effects of individual descriptors trees generated for each image. The first column defines the name of the picture which keypoints created the descriptor tree. The second row shows the number of descriptors that were generated by the SURF algorithm. Images are of similar size, however, there are large divergences between the amount of generated keypoints. Major impact on the amount of generated keypoints is the number of details presented in pictures as can be seen in images Fig. 7d, e. The third row of the table shows the total number of tree nodes. This number will be usually greater than the number of generated keypoints. This is caused by the additional intermediate nodes, which group leaf nodes and link them with the root of tree. The last row shows the number of leaf nodes of the tree. These nodes store the descriptor and other keypoints data. They have the longest descriptor, even

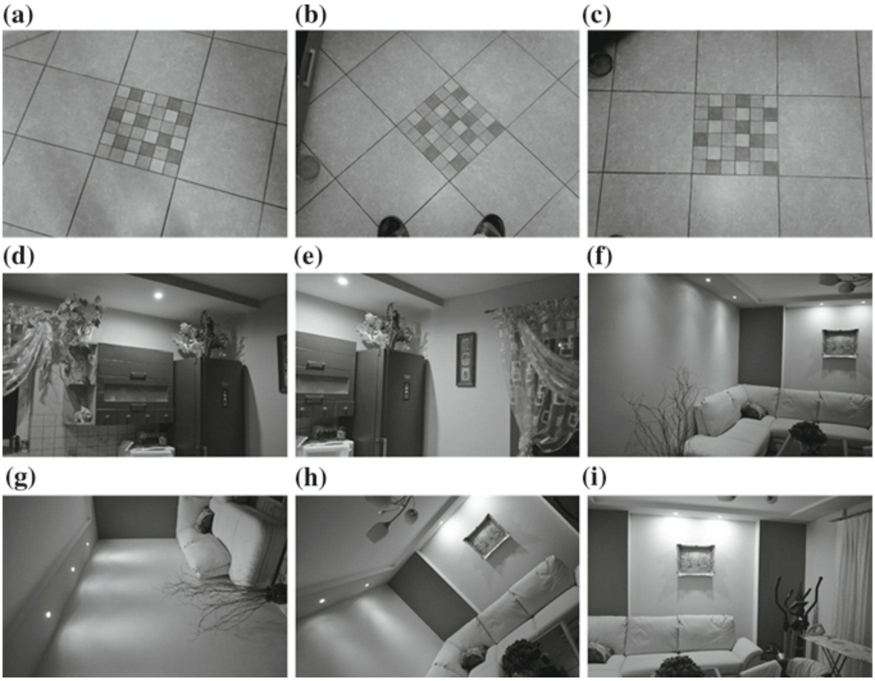


Fig. 7 Images used in experimental results

Table 2 Descriptors values of a subtree fragment created from a single keypoint

Tree level	$\mathbf{V}_{\mathbf{sub}}$	$\sum dx$	$\sum dy$	$\sum \lvert dx \rvert$	$\sum \lvert dy \rvert$
1	1	−0.4409	0.7248	0.5188	1.5143
2	1	0.0000	0.0000	0.0000	0.0000
	2	−0.4409	0.7248	0.5188	1.5143
3	1	0.0000	0.0000	0.0000	0.0000
	2	0.0000	0.0000	0.0000	0.0000
	3	−0.1519	0.9252	0.2248	1.1725
	4	−0.2890	−0.2004	0.2939	0.3418
4	1	0.0000	0.0000	0.0000	0.0000
	2	0.0000	0.0000	0.0000	0.0000
	3	0.0000	0.0000	0.0000	0.0000
	4	0.0000	0.0000	0.0000	0.0000
	5	−0.0043	0.7343	0.0528	0.7868
	6	−0.1476	0.1908	0.1720	0.3857
	7	−0.2081	−0.1448	0.2119	0.2534
	8	−0.0809	−0.0556	0.0820	0.0884

Table 3 Experimental results of tree creation from images presented in Fig. 7

Image	No. of tree leaves	No. of tree nodes	No. of image keypoints
7a	245	604	359
7b	256	710	318
7c	187	534	213
7d	858	2561	891
7e	754	2312	785
7f	394	1231	435
7g	274	797	295
7h	402	1322	489
7i	592	1940	650

greater than the combined descriptors of its parent nodes. As we can see in Table 3, the number of leaf nodes of the tree is always less than the number of keypoints which proves the grouping of similar descriptors. The average value of the grouped keypoints is 13 % of all generated keypoints. If an image is more homogeneous then the grouping ratio is higher as for image 7a where the rate of the grouping keypoints is 32 % and for various images as 7d grouping ratio is much smaller and equals only 4 % of all the keypoints. This implies that the algorithm will have greater efficiency in the case of homogeneous and simple images even large sizes.

During process of tree creation, the algorithm generates a certain number of intermediate nodes which the average number is 66 % of all tree nodes. This value is constant for each image and is between 59 and 70 %. It is invariant from image details or amount of generated keypoints. Their number is more affected by construction of the proposed tree structure. Even though that the number of intermediate nodes is greater than leaf nodes the amount of needed data is less because they use simplified descriptors.

The effects of image comparison between the descriptors tree and descriptors of other images is presented in Table 4. The results are presented in a matrix form. The matrix shows the comparison of the images in all-to-all combinations. The table shows three values for each comparison. The first value represents the number of compatible keypoints descriptors between the images. The next value is the number of examined tree nodes in the process of image comparison. Each examination was based on a descriptor comparison of tree nodes. The last value presents maximum possible number of combinations of keypoints comparisons which would have to be done in the case of comparing between two sets of image descriptors without any improvement. From the table it can be concluded that the average number of compared tree nodes is only 6 % of number of all possible keypoints combinations. This value in our experiment varies from 4 % to 8 %. Variations of this value is a result of the internal structure of the created tree. General result is better than traditional methods of comparing [2–5] even with the fact that the number of tree nodes is larger than the number of keypoints.

Table 4 Experimental images comparison results (see Fig. 7), values: final match descriptors, number of compared nodes, number of compared descriptor by list (all to all) method

Image	7a	7b	7c	7d	7e	7f	7g	7h
	36							
	6492							
7b	114162	–	–	–	–	–	–	–
	27	29						
	4735	4452						
7c	76467	67734	–	–	–	–	–	–
	0	1	0					
	15731	17367	1504					
7d	319869	283338	189783	–	–	–	–	–
	0	0	0	40				
	13014	14977	13429	28617				
7e	281815	249630	167205	699435	–	–	–	–
	1	0	0	0	1			
	6433	8306	6852	14542	15829			
7f	156165	138330	92655	387585	341475	–	–	–
	0	0	0	1	1	39		
	4171	5521	4615	10273	10591	9262		
7g	105905	93810	62835	262845	231575	128325	–	–
	2	1	0	4	7	71	56	
	7399	8526	7658	15407	17745	14341	11546	
7h	175551	155502	104157	435699	383865	212715	144255	–
	0	0	0	0	0	18	7	28
	9999	11996	10615	21908	23793	19100	16136	24746
7i	233350	206700	138450	579150	510250	282750	191750	317850

5 Conclusions

The proposed method can speed up image comparison thanks to building descriptors tree. It requires only approximately 6 % comparison operations of all possible keypoint combinations. Generated tree requires more memory than a list of all descriptors but its size is reduced by grouping similar descriptors. Number of similar descriptors depends on the amount of image details. The method obtained the worst results on images representing nature. In such a case, background contains a lot of unimportant and difficult to group details. These keypoints expand the descriptor tree by insignificant details of the compared images.

The method obtained the best results in processing artificial images created by humans, such as logos, diagrams, schematic. This is caused by a greater number of similar keypoints which mostly are related to the characteristic elements of the image that allows for better grouping of keypoints.

Acknowledgments The project was funded by the National Center for Science under decision number DEC-2011/01/D/ST6/06957. Patryk Najgebauer received a scholarship from the project DoktoRIS—Scholarship program for innovative Silesia co-financed by the European Union under the European Social Fund.

References

1. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: SURF: speeded up robust features. Comput. Vis. Image Underst. **110**, 346–359 (2008)
2. Bazarganigilani, M.: Optimized image feature selection using pairwise classifiers. J. Artif. Intell. Soft Comput. Res. **1**, 147–154 (2008)
3. Chang, Y., Wang, Y., Chen, C., Ricanek, K.: Improved image-based automatic gender classification by feature selection. J. Artif. Intell. Soft Comput. Res. **1**, 241–253 (2011)
4. Górecki, P., Artiemjew, P., Drozda, P., Sopyla, K.: Categorization of similar objects using bag of visual words and support vector machines. In: Proceedings of 4th International Conference on Agents and Artificial Intelligence, ICAART'12, pp. 231–236. Vilamoura, Algarve, Portugal (2012)
5. Górecki, P., Sopyla, K., Drozda, P.: Ranking by K-Means voting algorithm for similar image retrieval. In: Artificial Intelligence and Soft Computing 2012. LNCS, vol. 7267, pp. 509–517. Springer, Heidelberg (2012)
6. Guerrero, M.: A Comparative Study of Three Image Matching Algorithms: Sift, Surf, and Fast. Master Thesis, Utah State University (2011)
7. Najgebauer, P., Gabryel, M., Korytkowski, M., Scherer, R.: Tree representation of image key point descriptors. In: Proceedings of the 8th International Conference on Knowledge, Information and Creativity Support Systems, Krakow, Poland, pp. 402–412, Progress & Business Publishers, Krakow (2013)
8. Wang, L., Ju, H.: A robust blob detection and delineation method. In: ETT and GRS 2008 International Workshop on Geoscience and Remote Sensing, pp. 827–830, IEEE Press, New York (2008)

Creative Intersections of Computing and Music Composition

Ewa Łukasik

Abstract The goal of this paper is to present various aspects of music composition supported by computer algorithms and methods. Amateur pop composers search databases for music similar to their composition in order to be sure that certain aspects of their music are stylish, or on the contrary, to be sure that they did not commit plagiarism. Others are interested in surprising effects of automated composition and put some demands on the produced music, e.g., to convey a desired emotional character or to change musical parameters of the composition, preferably in real time, by *live coding*. Reflections on creativity in the domain of computer supported music composition is illustrated by some projects accomplished in the Institute of Computing Science, Poznan University of Technology, imaging current trends in this domain.

Keywords Creativity · Music composition · Music retrieval · Music genre classification · Music emotions recognition · Generative music

1 Introduction

Modern computer technologies have entered the stage of advanced audio creation, recording, processing, and disseminating. They have also influenced the way music is created, gave birth to new genres of music, and led to a rise of new music tastes in the audience. With the development of mobile technologies offering multitasking smart phones delivering *any media anytime* and *anywhere,* music can be listened to constantly. With powerful music production computer tools, the expansion of Web

E. Łukasik (✉)
Institute of Computing Science, Poznań University of Technology,
ul. Piotrowo 2, 60-965 Poznań, Poland
e-mail: Ewa.Lukasik@cs.put.poznan.pl

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7_42

2.0, and the dawn of Web 3.0, cheap music production possibilities and the *anybody-can-do it* paradigm enable many artists to create, perform, and noninstitutionally disseminate their music. Computer music has become an element of emerging intelligent technologies and should be considered within the Foresight program [32].

Creativity is one of the qualities unique to humans and is a key issue for artificial intelligence (AI) and cognitive sciences [5]. Its definition depends on the domain in which it is characterized, but it is always related to some kind of novelty.

People may be trained to improve creative skills (compare Idea Marathon introduced by [8]), since creativity may be regarded as a search for a solution to a problem, or for something surprising but suitable for a developing work or design [4]. In the past, music composition was regarded as an exclusive ability that required years of training. Nowadays, with the development of multimedia technology, it can be supported by a variety of computer tools that enable people to set up advanced music production studios at home and create music even without instruments or playing skills. The creative process in music composition may be related to automated music production, where a variety of music elements are modified or defined from scratch by a machine; thus musical creativity is supported by a computer [10].

A computer may be a multitasking tool in music creation. It may be used at the conceptual stage for generating score, creating basic structures of music-rhythm, melody, key, harmony, etc. The computer may be used live, playing like a musical instrument. It may enrich the compositional techniques or even automatically generate music. It can also interactively modify or enrich the timbre of the sound. Composers have to be computer literate to follow new trends in music. Computing specialists may introduce new means of musical expression and music construction. Both groups may profit from the synergistic effects of merging these domains.

The goal of this contribution, extended version of the paper [18], is threefold:

- to discuss various aspects of composing music in the computer era with the use of computers and computer methods,
- to present aspects of mining music data by music creators,
- to illustrate the above facets by the exemplary creative music projects accomplished in the Institute of Computing Science Poznan University of Technology, applying music information retrieval, machine learning, and other computing methods to produce music.

The paper is structured as follows: Sect. 2 discusses music composition as a collaborative act of creativity throughout history, highlighting computer methods supporting music composition. Section 3 briefly describes projects accomplished in the Institute of Computing Science, Poznan University of Technology, related to the use of novel computer methods for music creation support. Section 4 concludes the paper.

2 Music, Its Composition, and Artificial Intelligence

Music, its composition, and perception by humans is a complicated subject that has been analyzed by many theorists in the past and at present. Mazzola et al. noted in [19], that to create a composition, *it is certainly not about creating new combinations of score signs, but about the meaning, which is added to the given understanding of music*. In this sense, music composition is an advanced cognitive process not within the reach of everybody.

Functions of music differ with the flow of time. Music conveys and affects emotions. Music may activate or calm people down, or it may lead them into a state of contemplation. It may also influence people [30].

Fundamental questions concerning the nature of music and human experience of music are the subject of the philosophy of music studies [26]. Music is also of interest to semioticians, semiologists, anthropologists, linguists, sociologists, and neurologists [29]. Nattiez [21] proposed a global theory for the interpretation of specific music pieces and the phenomenon of music. He continued the theory of his master, Jean Molino, who in his paper, originally written in French in 1975 and translated into English [20] claimed: *Music, often an art/entertainment, is a total social fact whose definitions vary according to era and culture*.

2.1 The Concept of Collaborative Composition

Music composition, if simplified, may be regarded as putting sounds together [13]. The search for diversity is one of the aspects of music composition, but on the other hand, there is a human demand for repetition of pleasant stimuli. *Two impulses struggle with each other within man: the demand for repetition of pleasant stimuli, and the opposing desire for variety, for change, for a new stimulus*—said Arnold Schoenberg, an Austrian composer and painter, one of the most important twentieth century music theorist and composer (quotation after Zanette [40]). Zanette continues (ibid.): *Repetition of perceptual elements – melodic motifs, rhythmic patterns, harmonic progressions – brings coherence to the musical structure, which is the basis of its comprehensibility. Variation, in turn, is necessary to avoid monotony and dullness*. Humans can achieve it and the same would be expected from AI algorithms for automated music composition. It seems that the most difficult constraint for these algorithms is to generate music reflecting desired emotions.

Music composing is an act of creativity. But, on the other hand, composers, consciously or unconsciously, copy some patterns that already exist in music. Unconsciously, as humans naturally acquire knowledge and comprehend music from listening. Consciously, because some artists go with the musical trends, or, in the case of beginners, would like to *sound like the others*. In this sense, music composition may be treated as a transformed product of collaborative intelligence of other artists from the same or precedent eras.

A new paradigm in the digital music composition has appeared recently *do-it-together* (DIT), instead of *do-it-yourself* (DIY). Users distribute work collaboratively on a single or several musical pieces sharing a common interface all over the Internet [13, 14]. Another example is huge groups of music creators that use samples and loops provided by other artists as a basis for their own compositions. Such practitioners of music have got the name of *plunderphonics* [23].

2.2 Music Information Retrieval

The rapid progress in digital music production and distribution has led to the creation of large collections of music. They demand content-based music management in order to organize these collections automatically and to retrieve music upon a variety of different needs including query by example of melody, genre, mood, or music emotions. New algorithms concern both symbolic and audio digital recordings. In symbolic music information retrieval (MIR), structured signals are used, as MIDI files or OMR [12], and in audio–arbitrary unstructured audio signals. Still newer and newer feature spaces are considered for MIR tasks, e.g., those presented in [9].

The milieu of independent pop artists, often amateurs, either search music databases forthe inspiration, or try to check if their music, especially in melodic layer, does not constitute an unconscious plagiarism.

2.3 Music Emotions, Their Recognition, and Retrieval

There is a great interest in research concerning the emotions that music expresses or elicits in the listener. Music Emotion Recognition (MER) systems have a substantial research bibliography. They enable mining music according to mood and emotions [16]. The emotions are represented in various ways, most often as:

- a finite set of emotions (e.g., joy, sadness, anger, boredom, and aggression [7],
- arousal–valence space [39],
- 3D emotional space (arousal, valence, and pleasure [35]).

Classification of music emotions may be performed as a one-label [39], multi-label [36], binary, or ranking [39].

The process of retrieving music by emotions requires a carefully planned music signal features extraction and selection. The most prominent features are: timbral, spectral, tonal, and rhythmic [17, 35]. There are already repositories with the music labeled by emotions. Although the content-based recognition of music emotions is popular nowadays, it is still difficult to automate the generation of the music with desired emotional character. One of such attempts is presented in Sect. 3.3.

2.4 Computer Music Generation

Automated composition has been in mind from the beginning of music history. It is claimed that establishing rules of composition was a kind of automating the process of composing music. Today, automatic computer composition is one of the topics of Artificial Intelligence. David Cope's system called Experiments in Musical Intelligence (EMI [3]) analyzes the score structure of a MIDI sequence, finds frequent patterns (a signature), creates a database of the important segments, and mimics the style of a composer while creating new sequences of sound. His system can generate compositions with stylistic similarities to the originals [13]. An automated system by François Pachet, called 'Continuator' [24] learns the improvisation style of a musician playing a polyphonic MIDI instrument in real time. A computer can play on the fly, maintaining the initial style. These are only two well-known examples of systems for automated music composition, but nowadays, many more of them exist. In general, automated computer compositions are based on [25]: mathematical models, knowledge based systems, grammars, evolutionary methods, learning systems, and hybrid systems.

With the growing literacy in computing, composers frequently use computers in the process of creating music. Creators explore sound and its gradual change evoked by the system they have programmed. They follow ideas of Brian Eno, pioneer of *ambient music* organized nonlinearly, around a timbre rather than melody. He introduced the idea, willingly taken over by other electronic music authors, that a composer is first a planner and a programmer, and then becomes an audience to the results. The exploration of the sound itself was introduced by the Italian composer Giacinto Scelsi in the 1950s of the twentieth century: listeners experience a single pitch sound rich in timbre with microtonal oscillations, harmonic content, and dynamic change.

Computer systems today enable the development of the above initial ideas. Electronic music composers create systems that generate or modify sound unpredictably, with minimal involvement of an artist. Parameters for varying sound or music are not explicitly artist's decisions, but are related to other events. The reference for such systems is in Steve Reich's article *Music as a gradual process* [27], where the author stated: *I do not mean the process of composition, but rather pieces of music that are, literally, processes,* hence the name *process music*. Musical artwork is a process of its generation, so another term is *generative music* [38]. Recently, the generative music is produced interactively with other modalities, like haptics, other sounds, or visual textures in real time.

Modifying sound characteristics in real time is called *interactive electronic music* or *Live Coding* and has become a current practice for contemporary music creators. According to Blackwell et al. [2]: *Live coding is improvised interactive* *programming, typically to create electronic music and other digital media, done live with an audience.*

Contemporary sound systems, e.g., Max/MSP give enormous possibilities to form sound and its various timbral characteristics. This kind of music can be created exclusively by computer literate people; the music composition domain, therefore, opens its doors for computing.

3 Computers and Music Composition—Sample Projects

In this section we briefly describe four projects realized over the past 10 years in the Institute of Computing Science Poznan University of Technology by graduate students within their diploma thesis under the supervision of the author. They show how the new methods in computing may influence the creativity in music composition. The projects span themes from the anti-plagiarism support for an amateur pop music composer through the automated composition of music with the desired emotional character to the image analysis-driven modification of generative music. These projects show how creative the intersections of computing and music composing can be.

3.1 Query by Melody System for Preventing Plagiarism in Pop Music Composition

The project that we recall in this section was accomplished almost 10 years ago ([31], 2005). We take it back to show the creative manner in which computer literate amateur composer used computer to support his creative process. The system was created to prevent plagiarism in the melodic layer of a musical piece.

In popular music, melodies often consist of one or more musical phrases or motifs which are usually repeated throughout a song or piece in various forms. In multi-instrumental, symphonic compositions the psychoacoustic notion of a melody may be too complicated to be computationally discovered, as it may sweep between various instruments or voices, as in polyphonic music. In [28] the following characteristics of a melody may be found: finite succession of notes, cantabile pitch range, monophonic, lead part, identifies/characterizes the piece, song, unity, diversity, contains repeating patterns, often linked to text, done by humans, understandable, and easy to remember by humans. With such constraints, the problem of discovering the melody of a musical piece constitutes a challenge for Artificial Intelligence.

Fortunately, popular music is very often transcribed to MIDI format. MIDI files are essentially similar to musical scores and typically describe the start, duration, volume, and instrument type for every note of a musical piece, so extracting a melody line seems to be easy. However, it is usually a multi-track representation, where each track is devoted to an instrument or a group of instruments. Then, a new objective occurs, to select one track containing the melody from a list of input tracks. This problem has been already solved in the literature using statistical properties of the musical content and pattern recognition techniques.

The system called *Melody Finder* [31] is a database of MIDI files where *query-by-example* has been applied, with the melody as an example. Such implementations of melody search are generally used by music consumers to find out the desired song, but in this case, the system was designed from the point of view of an amateur composer who wanted to check if the melody of the piece of music just composed was not the one that had already been created.

In the *Melody Finder* system, a heuristic strategy to find the melody track has been implemented. Observations of MIDI files in MIDI editor led to the conclusion that a great deal of melody tracks do not have multiple simultaneous notes. In this way, it has been determined that almost every piece of music, which has the original vocal line, has a record of this line in the form of scores on a separate track. Exceptions are in classical music, with no vocal line. On an average, only 4 out of 11 tracks can potentially be melodic lines in the popular music. Since further reduction of track-candidates would be time-consuming, it was decided to analyze all of them to create metadata related to the melody.

The melody metadata is represented by the intervals between the consecutive notes, similar to the MPEG-7 standard. In this way, the query may be independent from the music key.

The musical theme defined by a user may be inserted to a system either by a classical MIDI keyboard or by using a virtual keyboard built into the system and displayed on a computer screen.

3.2 Genre Classifier for Tagging Music Compositions

The project presented in this section was accomplished in 2010, when automatic recognition of music genre was (and still is) on the top of interest of potential pop music creators.

The music genre is a set of characteristics of a group of musical works which distinguishes a given piece from others. End users are accustomed to browsing music on the web by genre. The most often used taxonomy refers to the popular Western music and also dance and ethnic music, e.g., Latin or Irish. There is a lot of digital music that is not tagged at all. That is why a lot of effort is put to develop efficient algorithms for automatic genre classification. The most notable is the work of Tzanetakis and Cook [34]. The system quoted here, described in detail in [6], was tested on the music from Jamendo Music Social Network. The genres are tagged by artists themselves textually, which is often imprecise.

A simple system would help to augment the consistency of music description, therefore, such a system has been proposed in [6]. The search for mp3 files was conducted using the parameters related to the music timbre (MFCC), tempo, histogram, and other typical features characterizing music content. A two-level hierarchical taxonomy has been proposed for 6 genres and 13 subgenres. Classification results reached the level of *state-of-the-art* systems which on an average are nearly 75 % accurate, varying for different genres. The system proposed could assist musicians in preparing a more correct description of their music published in social networks and thus it might improve users' satisfaction.

3.3 Automated Composition of Music with Predefined Emotional Character

The system presented in this subsection (fully described in [37]) is an example of an automatic generation of melodies. Its novelty consists in the fact that the music automatically composed carries a desired emotional character.

The system generates an initial motif, and then applies the genetic algorithm to modify it in the iterative way. The music is constantly examined by the module that is able to recognize the emotions. It was earlier trained by the set of musical pieces labeled by the emotions they carry. In the prototype version of the system, four emotions might have been chosen: joy, sadness, aggression, and calm. The module was trained utilizing low level features describing music emotions, mainly based on the spectral analysis. To ensure the musical correctness of the automatically generated music from the very beginning, several musical rules had to be applied at the initial stage of the generation of musical motifs.

The architecture of the system has been inspired by the modular system proposed in 1995 by Jacob [11] with the additional module responsible for the recognition of emotions. Even if the system automatically generates music, it still needs social network to mine the music according to emotions, which is needed to train the assessing module. Therefore, the emotional character of the music will be built on existing pieces and their emotional labels.

3.4 Image Contours Driven Modification of Live Generative Music

The project presented in this section is an example of the most recent trend of using computers in the real-time creation of ambient music, namely *live coding*. It was accomplished by Szymon Kaliski as his MSc project in the Institute of Computing Science Poznan University of Technology in 2013 [15].

The aim of the project was to build an application that allows to modify desired parameters of musical piece interactively. Unlike in other similar applications music parameters are controlled neither by apparent decisions of the system user (composer) nor by any rules but by using digital image processing. Detected contours of an image are used as the input data controlling music generation. Such an approach has been inspired by the nonclassical music notation—image contours are treated as a specific graphical music notation. If the image is divided into several parts, contours detected in each of the parts may control different sound parameters or generate different MIDI notes.

The attractiveness of this solution for the contemporary users lies in the fact that the task of an image acquisition and processing is performed in the mobile device in the time of performing music. Using *OSC* (*Open Sound Control*) protocol, enabling efficient communication between computer and multimedia devices, the mobile device controls the client application—digital audio station, in this case built using *max4live* that enables creating complex sound effects. Or it is linked with MIDI controller and then it enables to generate the sequence of MIDI commands to the built-in synthesizer to produce the sequence of sounds. Another function of the system is to personalize parameter values preferred by the user to be applied in other creations.

4 Conclusions

The development of computer, mobile, and Internet technologies has brought new opportunities for creative people to compose music. They have also changed the way of creating, distributing, and listening to the music. All these changes have already influenced and will stimulate the creativity—music creation in particular. The effects of synergy of music and computing surprise and will even more surprise the audience. A future composer will have to hold computing skills to follow the new trends.

This paper has highlighted some problems related to music composition that are practically undertaken by computer literate music creators. It has also presented some projects undertaken in the Institute of Computing Science, Poznań University of Technology by MSc students under the supervision of the author [15, 31, 37] and described in [6]. They have used computer technology to solve some nonstandard music composition problems. These projects were accomplished between 2005 and 2013 and show the evolution of digitally literate composers' interests and technical means of audio creation over time.

Computer music is one of the emerging technologies—very important in the contemporary and future Information Society that should be considered while planning its development. Let us, therefore, finish this paper with a quote from Nilson [22] that promises new effects of creative intersections of computing and music composition arising from the synergy of these two domains: *Rather than the composer-pianists of the 19th century*, *the 21st century is seeing a rise of composer-programmers, who can explore electronic music through its natural tool, the computer, even in live performance situations*.

References

1. Andersson, Ch.: The Long Tail: Why the Future of Business is Selling Less of More. Hyperion, New York (2006)
2. Blackwell, A., McLean, A., Noble, A., Rohrhuber, J. (eds.): Collaboration and learning through live coding. Report from Dagstuhl Seminar 13382. Dagstuhl Reports **3**(9), 130–168 (2013)
3. Cope, D.: Experiments in Music Intelligence. A-R Editions, Madison, Wisconsin (1996)
4. Dahlstedt, P.: Evolution in creative sound design. In: Miranda, E.R., Biles, J.A. (eds.) Evolutionary Computer Music, pp. 79–99. Springer, London (2007)
5. Dartnall, T.: Artificial Intelligence and Creativity: An Interdisciplinary Approach. Kluwer, Dordrecht (1994)
6. Dropik, L., Łukasik, E.: Two-level hierarchical classification of music genre for music social netwoks. Found. Comput. Decis. Sci. **35**(4), 261–282 (2010)
7. Hevner, K.: Experimental Studies of the Elements of Expression in Music. The American Journal of Psychology **48**(2), 246–268 (1936)
8. Higuchi, T., Yuizono, T., Miyata, K.: Creativity effects of Idea-Marathon System (IMS): Torrance Tests of Creative Thinking (TTCT) figural tests for college students. In: 8th Knowledge, Information and Creativity Support Systems, KICSS 2013, Kraków, pp. 138–148 (2013)
9. Homenda, W., Jastrzebska, A.: Complementarity and similarity of complementary structures in spaces of features and concepts. In: 8th Knowledge, Information and Creativity Support Systems, KICSS 2013, Kraków, pp. 149–160 (2013)
10. Indurkhya, B.: On the role of computers in creativity-support systems. In: 8th Knowledge, Information and Creativity Support Systems, KICSS 2013, Kraków, pp. 233–244 (2013)
11. Jacob, B.L.: Composing with genetic algorithms. In: 1995 International Computer Music Conference, pp. 452–455 (1995)
12. Jastrzębska, A., Lesiński, W.: Optical music recognition as the case of imbalanced pattern recognition: a study of single classifiers. In: 8th Knowledge, Information and Creativity Support Systems, KICSS 2013, Kraków, pp. 267–278 (2013)
13. Jehan, T.: Creating music by listening. Ph.D. Dissertation, MIT (2005)
14. Jordà, S., Wüst, O.: FMOL—a system for collaborative music composition over the web. In: 12th International Workshop on Database and Expert Systems Applications, DEXA, Munich, pp. 537–542 (2001)
15. Kaliski, Sz.: System for the creative modification of musical composition based on digital image analysis. Master Thesis, E. Łukasik supervisor, Poznań University of Technology (2013) (in Polish)
16. Kostek, B., Plewa, M.: Parametrisation and correlation analysis applied to music mood classification. Int. J. Comput. Intell. Stud. **2**(1), 4–25 (2013)
17. Laurier, C., Lartillot, O., Eerola, T., Toiviainen, P.: Exploring relationships between audio features and emotion in music. In: Triennial Conference of European Society for the Cognitive Sciences of Music, Jyväskylä, pp. 260–264 (2009)
18. Łukasik, E.: Mining music social networks from an independent artist perspective. In: 8th Knowledge, Information and Creativity Support Systems, KICSS 2013, Kraków, pp. 350–359 (2013)
19. Mazzola, G., Park, J., Thalmann, F.: Musical Creativity. Strategies and Tools in Composition and Improvisation. Springer, Berlin, Heidelberg (2011)
20. Molino, J., Underwood, J.A., Craig, A.: Musical fact and the semiology of music. Music Anal. **9**(2), 105–111 + 113–156 (1990)
21. Nattiez, J.-J., Abbate, C.: Music and Discourse: Toward a Semiology of Music. Princeton University Press (1990)
22. Nilson, C.: Live coding practice. In: The 7th International Conference on New Interfaces for Musical Expression NIME'07, pp. 112–117, New York (2007)

23. Oswald, J.: Plunderphonics, or Audio piracy as a compositional prerogative. In: Wired Society Electro-Acoustic Conference, Toronto (1985) http://www.plunderphonics.com/xhtml/xplunder.html. Accessed 7 Feb 2014
24. Pachet, F.: The continuator: musical interaction with style. J. New Music Res. **32**(3), 333–341 (2003)
25. Papadopoulos, G., Wiggins, G.: AI methods for algorithmic composition: a survey, a critical view and future prospects. In: The Society for the Study of Artificial Intelligence and the Simulation of Behaviour Symposium on Musical Creativity AISB'99, pp. 110–117 (1999)
26. Philosophy of Music, Stanford Encyclopedia of Philosophy. http://plato.stanford.edu/entries/music/#2. Accessed 7 Feb 2014
27. Reich, S.: Music as a Gradual Process, Writings on Music 1965–2000. Oxford University Press (2002)
28. Rizo, D., León, P, Pertusa, A., Iñesta, J.: Melodic track identification in MIDI files. In: 19th International Florida Artificial Intelligence Research Society Conference, pp. 254–259 (2006)
29. Rose, F.C.: The Neurology of Music. Imperial College Press (2010)
30. Schaeffer, B.: The History of Music. WSiP, Warszawa (1983). (in Polish)
31. Skraburski, P.: Algorithm for musical motifs retrieval from the MIDI database based on MIDI pattern example. Master Thesis, E. Łukasik supervisor, Poznań University of Technology (2005) (in Polish)
32. Skulimowski, A.M.J.: Towards a new insight into technological evolution: the foresight of information society and emerging intelligent technologies. In: 8th Knowledge, Information and Creativity Support Systems, KICSS 2013, Kraków, pp. 599–610 (2013)
33. Tanaka, A., Tokui, N., Momeni, A.: Facilitating collective musical creativity. In: 13th Annual ACM International Conference on Multimedia MM'05, Singapore, pp. 191–198 (2005)
34. Tzanetakis, G., Cook, P.: Musical genre classification of audio signals. IEEE Trans. Speech Audio Process. **10**(5), 293–302 (2002)
35. Wieczorkowska, A., Synak, P., Lewis, R.A., Raś, Z.W.: Creating reliable database for experiments on extracting emotions. In: Kłopotek, M.A., Wierzchoń, S.T., Trojanowski, K. (eds.) Intelligent Information Processing and Web Mining, pp. 395–404. Springer, Berlin, Heidelberg (2005)
36. Wieczorkowska, A., Synak, P.: Quality assessment of k-NN multi-label classification for music data. In: Esposito, F., Raś, Z.W., Malerba, D., Semeraro, G.: Foundations of Intelligent Systems, LNCS 4203, pp. 389–398, Springer, Berlin, Heidelberg (2006)
37. Winiarz, Sz.: Automated composition of music with the determined emotional character. Master Thesis, E. Łukasik supervisor, Poznan Univ. of Technology (2013) (in Polish)
38. Wooller, R., Brown, A.R., Miranda, E., Berry, R., Diederich, J.: A framework for comparison of processes in algorithmic music systems. In: Generative Arts Practice: Creativity and Cognition Symposium, Sydney, pp. 109–124 (2005) http://eprints.qut.edu.au/6544/1/6544.pdf. Accessed 7 Feb 2014
39. Yang, Y., Chen, H.: Music emotion ranking. In: International Conference Acoustics, Speech and Signal Processing ICASSP'09, Taipei, Taiwan, pp. 1657–1660 (2009)
40. Zanette, D.H.: Music, complexity, information. Comput. Res. Repos. arXiv:0807.0565 (2008)

Author Index

A
Adrian, Weronika T., 13
Alves, João, 53
Assimakopoulos, D., 329

C
Cetnarowicz, Krzysztof, 151
Chmiel, Wojciech, 315
Concilio, Grazia, 269

D
de Amorim, Renato Cordeiro, 465
De Pietro, Giuseppe, 345, 359
de Silva Garza, Andrés Gómez, 259
Duda, Piotr, 529
Dutkiewicz, Lidia, 123
Dżega, Dorota, 137

E
Echizen, Isao, 201
Esposito, Massimo, 345, 359, 373

F
Filipowicz, Bogusław, 315

G
Gabryel, Marcin, 111, 529
Garofalakis, J., 329
Gore, Shounak, 401
Govindaraju, Venu, 401
Grobler-Dębska, Katarzyna, 123

H
Higuchi, Takeo, 185
Homenda, Wladyslaw, 413, 427
Homola, Petr, 455
Horzyk, Adrian, 39

I
Iannaccone, Marco, 373
Iida, Takuya, 201
Indurkhya, Bipin, 213

J
Jastrzebska, Agnieszka, 427, 441, 493
Jaszuk, Marek, 479
Jaworski, Maciej, 529

K
Kacprzyk, Janusz, 1
Kadłuczka, Piotr, 315
Kajiyama, Tomoko, 201, 519
Kawaji, Takahiro, 185
Köhler, Thomas, 171
Korytkowski, Marcin, 529
Kucharska, Edyta, 123
Kunifuji, Susumu, 165
Kwiecień, Joanna, 315

L
Lange, Sascha, 53
Lesinski, Wojciech, 441, 493
Li, Zhenpeng, 27
Ligęza, Antoni, 13
Luckner, Marcin, 413
Łukasik, Ewa, 553

M
Marsh, Jesse, 269
Marszałek, Zbigniew, 111
Minutolo, Aniello, 345
Mirkin, Boris, 465
Miyata, Kazunori, 185
Molinari, Francesco, 269
Moolwat, Onuma, 67

A.M.J. Skulimowski and J. Kacprzyk (eds.), *Knowledge, Information and Creativity Support Systems: Recent Trends, Advances and Solutions*, Advances in Intelligent Systems and Computing 364, DOI 10.1007/978-3-319-19090-7

N
Naghsh, Fatemeh Azadi, 229
Najgebauer, Patryk, 541
Nalepa, Grzegorz J., 13, 99
Neves, José, 53
Niedúwiecki, Michał, 151
Nowak, Maciej, 301
Nowicki, Robert K., 111

O
Ouchi, Noritomo, 201

P
Pacharawongsakda, Eakasit, 389
Pascalau, Emilian, 99
Pechsiri, Chaveevan, 67
Pedrycz, Witold, 413
Pietruczuk, Lena, 529
Pietruszkiewicz, Wiesław, 137
Piriyakul, Rapepun, 67
Pota, Marco, 359
Pukocz, Przemysław, 315

R
Rączka, Krzysztof, 123
Riedmiller, Martin, 53
Rizzo, Francesca, 269
Rzecki, Krzysztof, 151

S
Sakurai, Keizo, 185
Satoh, Shin'ichi, 519
Scherer, Rafał, 529, 541
Schmitt, Ulrich, 243
Skulimowski, Andrzej M.J., 1, 279
Starzyk, Janusz A., 479
Szwed, Piotr, 507

T
Tang, Xijin, 27, 85
Tezuka, Taro, 229
Theeramunkong, Thanaruk, 389
Tzagarakis, M., 329

W
Wei, Cuiping, 85
Woźniak, Marcin, 111

Y
Yasumasa, Shun, 229
Yuizono, Takaya, 185