Valérie Monfort
Karl-Heinz Krempels
Tim A. Majchrzak
Žiga Turk (Eds.)

LNBIP 246

Web Information Systems and Technologies

11th International Conference, WEBIST 2015
Lisbon, Portugal, May 20–22, 2015
Revised Selected Papers

Springer

Lecture Notes in Business Information Processing 246

More information about this series at http://www.springer.com/series/7911

Valérie Monfort · Karl-Heinz Krempels
Tim A. Majchrzak · Žiga Turk (Eds.)

Web Information Systems and Technologies

11th International Conference, WEBIST 2015
Lisbon, Portugal, May 20–22, 2015
Revised Selected Papers

Springer

Editors

Valérie Monfort
University of Paris
Paris
France

Karl-Heinz Krempels
RWTH Aachen University
Aachen
Germany

Tim A. Majchrzak
Department of Information Systems
University of Agder
Kristiansand, Vest-Agder Fylke
Norway

Žiga Turk
Faculty of Civil and Geodetic Engineering
University of Ljubljana
Ljubljana
Slovenia

ISSN 1865-1348 ISSN 1865-1356 (electronic)
Lecture Notes in Business Information Processing
ISBN 978-3-319-30995-8 ISBN 978-3-319-30996-5 (eBook)
DOI 10.1007/978-3-319-30996-5

Library of Congress Control Number: 2016934962

Printed on acid-free paper

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

Preface

This book includes extended and revised versions of a set of selected papers from WEBIST 2015 (the 11th International Conference on Web Information Systems and Technologies), held in Lisbon, Portugal, in 2015, organized and sponsored by the Institute for Systems and Technologies of Information, Control and Communication (INSTICC). The conference was held in cooperation with the Association Tunisienne pour l'Intelligence Artificielle (ATIA) and the Web Intelligence Consortium (WIC). This conference was also technically sponsored by the European Research Center for Information System (ERCIS).

The purpose of the WEBIST series of conferences is to bring together researchers and practitioners interested in technological advances and business applications of Web-based information systems. WEBIST 2015 had five main topic areas, covering different aspects of Web information systems, namely, Internet technology, Web interfaces and applications, society, e-business, e-government, Web intelligence, and mobile information systems, We strongly believe the proceedings demonstrate new and innovative solutions, and highlight technical problems in each field that are challenging and significant.

The papers included in this book were selected from those with the best reviews taking into account the quality of their presentation at the conference, assessed by the session chairs. Therefore, we hope that you find these papers interesting, and we trust they may represent a helpful reference for all those who need to address any of the aforementioned research areas.

We wish to thank all those who supported and helped to organize the conference. On behalf of the conference Organizing Committee, we would like to thank the authors, whose work contributed to a very successful conference, and the members of the Program Committee, whose expertise and diligence were instrumental in ensuring the quality of the final contributions. We also wish to thank all the members of the Organizing Committee whose work and commitment were invaluable. Last but not least, we would like to thank Springer for their collaboration in getting this book to print.

May 2015

Valérie Monfort
Karl-Heinz Krempels
Tim A. Majchrzak
Žiga Turk

Organization

Conference Co-chairs

Valérie Monfort	LAMIH Valenciennes UMR CNRS 8201, France
Karl-Heinz Krempels	RWTH Aachen University, Germany

Program Co-chairs

Tim A. Majchrzak	University of Agder, Kristiansand, Norway
Žiga Turk	University of Ljubljana, Slovenia/Reflection Group, Slovenia

Program Committee

Silvia Abrahão	Universitat Politecnica de Valencia, Spain
Isaac Agudo	University of Malaga, Spain
Jose Luis Herrero Agustin	University of Extremadura, Spain
Mugurel Ionut Andreica	Polytechnic University of Bucharest, Romania
Guglielmo de Angelis	CNR - IASI, Italy
Margherita Antona	Foundation for Research and Technology - Hellas (FORTH), Greece
Valeria De Antonellis	University of Brescia, Italy
Ismailcem Budak Arpinar	University of Georgia, USA
Elarbi Badidi	United Arab Emirates University, United Arab Emirates
Andrea Ballatore	University College Dublin, Ireland
Werner Beuschel	University of Applied Sciences, Brandenburg, FRG, Germany
Eva Blomqvist	Linköping University, Sweden
James Blustein	Dalhousie University, Canada
Christoph Bussler	Oracle Corporation, USA
Maria Claudia Buzzi	CNR, Italy
Elena Calude	Massey University, Institute of Natural and Mathematical Sciences, New Zealand
Pasquina Campanella	University of Bari Aldo Moro, Italy
Cinzia Cappiello	Politecnico di Milano, Italy
Nunzio Casalino	Università degli Studi Guglielmo Marconi, Italy
Sven Casteleyn	Universitat Jaume I, Spain
Ana R. Cavalli	Institute Telecom SudParis, France
Christophe Claramunt	Naval Academy Research Institute, France
Mihaela Cocea	University of Portsmouth, UK
Martine De Cock	Ghent University, Belgium

Christine Collet	Grenoble Institute of Technology, France
Marco Comuzzi	City University London, UK
Isabelle Comyn-Wattiau	Cnam and Essec, France
Emmanuel Coquery	Université Claude Bernard Lyon 1, France
Alfredo Cuzzocrea	ICAR-CNR and University of Calabria, Italy
Antonina Dattolo	University of Udine, Italy
Steven Demurjian	University of Connecticut, USA
Enrico Denti	Alma Mater Studiorum - Università di Bologna, Italy
Stefan Dessloch	Kaiserslautern University of Technology, Germany
Schahram Dustdar	Vienna University of Technology, Austria
Atilla Elci	Aksaray University, Turkey
Larbi Esmahi	Athabasca University, Canada
Davide Eynard	University of Lugano, Switzerland
Anna Fensel	STI Innsbruck, University of Innsbruck, Austria
Miriam Fernandez	The Open University, UK
Joao Carlos Amaro Ferreira	ISEL, Portugal
Josep-Lluis Ferrer-Gomila	Balearic Islands University, Spain
Filomena Ferrucci	Università di Salerno, Italy
Karla Donato Fook	IFMA - Maranhão Federal Institute for Education, Science and Technology, Brazil
Howard Foster	City University London, UK
Geoffrey Charles Fox	Indiana University, USA
Pasi Fränti	Speech and Image Processing Unit, University of Eastern Finland, Finland
Britta Fuchs	FH Aachen, Germany
Ombretta Gaggi	Università di Padova, Italy
Panagiotis Germanakos	University of Cyprus, Cyprus
Massimiliano Giacomin	Università degli Studi di Brescia, Italy
Henrique Gil	Escola Superior de Educação do Instituto Politécnico de Castelo Branco, Portugal
Nuno Pina Gonçalves	Superior School of Technology, Polithecnical Institute of Setúbal, Portugal
Anna Goy	University of Turin, Italy
Ratvinder Grewal	Laurentian University, Canada
Angela Guercio	Kent State University, USA
Francesco Guerra	University of Modena and Reggio Emilia, Italy
Miguel Guinalíu	Universidad de Zaragoza, Spain
Aaron Gulliver	University of Victoria, Canada
Shanmugasundaram Hariharan	TRP Engineering College, India
A. Henten	Aalborg University, Denmark
Jane Hunter	University of Queensland, Australia
Emilio Insfran	Universitat Politècnica de València, Spain
Ivan Ivanov	SUNY Empire State College, USA
Kai Jakobs	RWTH Aachen University, Germany

Monique Janneck	Luebeck University of Applied Sciences, Germany
Ivan Jelinek	Czech Technical University in Prague, Czech Republic
Yuh-Jzer Joung	National Taiwan University, Taiwan
Carlos Juiz	Universitat de les Illes Balears, Spain
Katerina Kabassi	TEI of the Ionian Islands, Greece
Georgia Kapitsaki	University of Cyprus, Cyprus
George Karabatis	UMBC, USA
Sokratis Katsikas	University of Piraeus, Greece
Matthias Klusch	German Research Center for Artificial Intelligence (DFKI) GmbH, Germany
Hiroshi Koide	Kyushu Institute of Technology, Japan
Fotis Kokkoras	TEI of Thessaly, Greece
Karl-Heinz Krempels	RWTH Aachen University, Germany
Tsvi Kuflik	The University of Haifa, Israel
Weigang Li	University of Brasilia, Brazil
Dongxi Liu	CSIRO, Australia
Xumin Liu	Rochester Institute of Technology, USA
Leszek Maciaszek	Wroclaw University of Economics, Poland and Macquarie University, Sydney, Australia
Michael Mackay	Liverpool John Moores University, UK
Tim A. Majchrzak	University of Agder, Kristiansand, Norway
Dwight Makaroff	University of Saskatchewan, Canada
Massimo Marchiori	University of Padua, Italy
Kazutaka Maruyama	Meisei University, Japan
Maristella Matera	Politecnico di Milano, Italy
Dennis McLeod	University of Southern California, USA
Tarek Melliti	University of Evry, France
Ingo Melzer	Daimler AG, Germany
Emilia Mendes	Blekinge Institute of Technology, Sweden
Abdelkrim Meziane	CERIST Alger, Algeria
Tommi Mikkonen	Institute of Software Systems, Tampere University of Technology, Finland
Valérie Monfort	LAMIH Valenciennes UMR CNRS 8201, France
Tomasz Muldner	Acadia University, Canada
Stavros Nikolopoulos	University of Ioannina, Greece
Alex Norta	Independent, Estonia
Vit Novacek	Digital Enterprise Research Institute, Nuig, Ireland
Paulo Novais	Universidade do Minho, Portugal
Dusica Novakovic	London Metropolitan University, UK
Laura Papaleo	Université Paris Sud, France
Eric Pardede	La Trobe University, Australia
Kalpdrum Passi	Laurentian University, Canada
David Paul	The University of New England, Australia, Australia
José António Sena Pereira	IPS - ESTSetúbal, Portugal

Toon De Pessemier	Ghent University - iMinds, Belgium
Josef Pieprzyk	Queensland University of Technology, Australia
Luis Ferreira Pires	University of Twente, The Netherlands
Pierluigi Plebani	Politecnico di Milano, Italy
Jim Prentzas	Democritus University of Thrace, Greece
Birgit Pröll	Johannes Kepler University Linz, Austria
Dana Al Qudah	University of Warwick, UK
Werner Retschitzegger	Johannes Kepler University, Austria
Thomas Risse	L3S Research Center, Germany
Thomas Ritz	FH Aachen, Germany
Davide Rossi	University of Bologna, Italy
Gustavo Rossi	Lifia, Argentina
Davide Di Ruscio	University of L’Aquila, Italy
Maytham Safar	Kuwait University, Kuwait
Aziz Salah	Université du Québec à Montréal, Canada
Yacine Sam	University Tours, France
Comai Sara	Politecnico di Milano, Italy
Anthony Savidis	Institute of Computer Science, FORTH, Greece
Claudio Schifanella	Università degli Studi di Torino, Italy
Wieland Schwinger	Johannes Kepler University, Austria
Jochen Seitz	Technische Universität Ilmenau, Germany
Mohamed Sellami	RDI Group, LISITE LAB, ISEP Paris, France
Weiming Shen	NRC Canada, Canada
Marianna Sigala	University of the Aegean, Greece
Marten van Sinderen	University of Twente, The Netherlands
Richard Soley	Object Management Group, Inc., USA
Chris Staff	University of Malta, Malta
Anna Stavrianou	Laboratoire LIG Grenoble, France
Hussein Suleman	University of Cape Town, South Africa
Samir Tata	Institut Mines Télécom, France
Dirk Thissen	RWTH Aachen University, Germany
Giovanni Toffetti	IBM Research Lab Haifa, Israel
Raquel Trillo	University of Zaragoza, Spain
Ziga Turk	University of Ljubljana, Slovenia/Reflection Group, Slovenia
Juergen Umbrich	Vienna University of Economics and Business, Austria
Geert Vanderhulst	Alcatel-Lucent Bell Labs, Belgium
Jari Veijalainen	University of Jyvaskyla, Finland
Petri Vuorimaa	Aalto University, Finland
Olga Vybornova	Université Catholique de Louvain, Belgium
Mohd Helmy Abd Wahab	Universiti Tun Hussein Onn Malaysia, Malaysia
Fan Wang	Microsoft, USA
Jason Whalley	Northumbria University, UK
Maarten Wijnants	Hasselt University, Belgium
Manuel Wimmer	Technische Universität Wien, Austria
Viacheslav Wolfengagen	Institute JurInfoR, Russian Federation

Guandong Xu	University of Technology Sydney, Australia
Kostas Zafiropoulos	University of Macedonia, Greece
Amal Zouaq	Royal Military College of Canada, Canada

Additional Reviewers

Golnoosh Farnadi	Ghent University, The Netherlands
Isaac Lera	Universitat de les Illes Balears, Spain
Lin Li	Wuhan University of Technology, China
Diego Rivera	Institut Telecom SudParis, France
Raul Armando Fuentes Samaniego	Telecom SudParis, France
Alexander Semenov	University of Jyvaskyla, NRU ITMO, Finland
Shuaiqiang Wang	University of Jyväskylä, Finland

Invited Speakers

Victor Chang	Leeds Beckett University, UK
Paolo Traverso	Center for Information Technology - IRST (FBK-ICT), Italy
Bernd Amann	LIP6 - Pierre and Marie Curie University, France
Nishanth Ramakrishna Sastry	King’s College London, UK
Alberto Broggi	VisLab - Università di Parma, Italy
Cornel Klein	Siemens AG, Germany

Contents

Papers

Web Interfaces and Applications

Internet Technology

Society, e-Business and e-Government

Web Intelligence

Mobile Information Systems

Web Interfaces and Applications

Entity Grouping for Accessing Social Streams via Word Clouds

Martin Leginus[1](✉), Leon Derczynski[2], and Peter Dolog[1]

[1] Department of Computer Science, Aalborg University,
Selma Lagerlofs Vej 300, 9200 Aalborg, Denmark
{mleginus,dolog}@cs.aau.dk

[2] Department of Computer Science, University of Sheffield, S1 4DP, Sheffield, UK
leon@dcs.shef.ac.uk

Abstract. Word clouds have been proven as an effective tool for information access in different domains. As social media is a main driver of large increase in available user generated content, means for accessing information in such content are needed. We study word clouds as a means for information access in social media. Currently-used clouds that are generated from social media data include redundant and misranked entries, harming their utility. We propose a method for generating improved word clouds over social streams. In this method, named entities are detected, disambiguated and aggregated into clusters, which in turn inform cloud construction. We show that word clouds using named entity clusters attain broader coverage and decreased content duplication. Further, an extrinsic evaluation shows improved access to data, with word clouds having grouped named entities being rated more relevant and diverse. Additionally we find word clouds with higher Mean Average Precision (MAP) tend to be more relevant to underlying concepts. Critically, this supports MAP as a tool for predicting cloud quality without needing a human.

Keywords: Word clouds · Recognized named entities · User evaluation · Social media · Social stream access

1 Introduction

A word cloud is a visual information retrieval interface which presents prominent and interesting terms from the underlying data collection. Word clouds allow quick access and exploration over document collections [1] and reduce information overload [2]. There are various studies about tag cloud generation from folksonomy data [3,4], but few studies available about word clouds generated from user generated content on social media [5].

To investigate information access over social media, we investigate the "model organism" of this data type, Twitter [6], a worldwide popular online social network where users publish daily an enormous amount of content (upwards of 600 million pieces of content per day). Therefore, Twitter users often face information

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 3–24, 2016.
DOI: 10.1007/978-3-319-30996-5_1

overload while searching, browsing and exploring tweets [7]. To enhance information access to relevant tweets, one might leverage word-cloud based retrieval interfaces. Word clouds can be intuitively employed for browsing of underlying collection of tweets and at the same time enabling access only to relevant content. For instance, the interactive browsing interface Eddi, where a word cloud is a core component of the interface [7], helps to decrease information overload. According to users, Eddi gives a more efficient and enjoyable mode of browsing the enormous amount of user stream tweets. To further improve usefulness of word clouds, a personalized cloud generation is proposed [5]. The suggested framework combines different user past actions (user past tweets and retweets) with negative user past preferences (tweets read but not indicated as relevant) to generate personalized word clouds. Such personalized clouds enhance access to relevant tweets when compared with state-of-the-art non-personalized approaches. Authors find user past retweets as more useful for personalization of clouds than user past published tweets. In addition, negative past preferences when combined with user past positive actions further improve the quality of word clouds. Similarly, word clouds ease e-health monitoring when browsing large collections of tweets [8].

Despite the benefits of word clouds for accessing and browsing social stream data, it remains a difficult type of text to handle. As a result of the diversity of language choice and spelling present in social media, end users are often presented with several different terms that refer to the same entity or concept, each term using different syntax and form; this leads to an increase lexical sparsity for the same degree of conceptual sparsity [9].

Hence, conventional methods of generating word clouds lead to undesirable results when applied to social stream text. In particular, variations of proper nouns create duplicated clusters, each of reduced prominence. Compounding the issue, variety in expression is increased by tight space constraints in some formats (like Twitter's 140-character limit) and by social media's generally informal, uncurated setting, as well as the inclusion of quasi-word hashtags [10,11]. For example, the football club *"Manchester United"* may also be referred to as *"MUFC"* and *"Man U"*. Adding entries for each of these leads to a decrease in the prominence of this key concept, while also taking up space in the cloud and thus reducing its eventual diversity.

Redundancies in the word cloud might lead to user confusion and an inability to effectively browse, explore and retrieve other relevant content. Therefore, two aspects should be considered when designing word cloud generation algorithms. Primarily, one aims to condense divergent terms describing the same concept into a single term. Also, a cloud's high-level diversity must be maximised, so the cloud gives a broad account of topics in the collection. These two conflicting requirements must be balanced to achieve an optimal word cloud.

The aim of this study is to improve word cloud generation by grouping coreferent entity expressions across multiple documents, applying existing named entity recognition systems in a novel fashion and grounding terms in this difficult genre to linked data resources. We systematically study the benefits of grouped named entities on word cloud generation, and investigate the role of unsupervised hierarchical clustering in finding candidate entity synonyms. This work builds on a previous conference paper version, [12]. We use three established synthetic metrics – Coverage, Overlap and Mean Average Precision (MAP) – for word clouds generated from social media data [4,5]. Further, to verify the findings of the synthetic evaluation, we perform a user study com-paring clouds with grouped and un-grouped named entities. The main contributions and findings of this paper are:

- The best performing DivRankTermsEntities method significantly increases Coverage with respect to the baseline method ($p = 0.0363$) and significantly decreases Overlap ($p = 0.000094$) (decreased redundancies). In addition, access to relevant documents is improved.
- Word clouds with grouped named entities are significantly more relevant ($p = 0.00062$, one sample t-test) and diverse ($p = 0.003$, one sample t-test).
- Users report that word clouds with grouped named entities that attain higher levels of MAP are more relevant than the word clouds with decreased levels of MAP. Hence, the MAP metric should be considered and measured when designing word cloud generation methods.

The structure of the paper is as follows. Section 2 provides a brief description of relevant related work. Section 3 describes a general process of word cloud generation and points out the focus of our work, and presents a method for word cloud generation with grouped entities, experimenting with both commerical and unsupervised entity alias extraction. In Sect. 4, we describe graph-based word cloud generation which is the underlying framework for the later evaluation. Section 5 presents the findings from an offline evaluation of generated word clouds from TREC2011 microblog collection as well as results of the performed user study. Finally, we discuss the paper's contributions as well as possible limitations of this work in Sects. 6 and 7.

2 Related Work

2.1 Word Cloud Generation

Tag cloud generation from folksonomy data has been thoroughly researched. Several tag cloud generation methods are proposed [3,4] and even synthetic metrics expressing tag cloud quality designed [4]. There are few studies that explore the benefits of word clouds for browsing social stream data. For instance, the browsing tool Eddi, where word cloud is a core component of the interface [7], helps to decrease information overwhelm. Similarly, word clouds are useful for the detection of epidemics when browsing thousands of tweets is needed [8].

Crowdsourcing has been used to recognize named entities in tweets [13]. This study reports that word clouds with named entities recognized by human workers are considered better. This supports our motive to promote and improve the handling of named entities in word clouds recognized in tweets.

Our contribution beyond the study from [13] is threefold. First, we perform grouping of recognized named entities, which has a positive impact on the generated word clouds. Second, we systematically study how to generate word clouds with named entities and measure performance using multiple metrics. Third, we compare these measured performances with user ratings, to discover relations between metrics and the user's perspective.

2.2 Social Stream Text Repair

Social stream text is noisy, and difficult to process with typical language processing tools [10]. Consolidation of the varying expressions used to mention entities is possible, over large well-formed corpora [14]. Achieving this over social streams presents new challenges, in terms of the reduced context and heightened diversity of expression [15–17]. The field continues to be active, with the ACL 2015 W-NUT shared task being dedicated to text normalisation and attracting many submissions [18] – the other task at this venue being named entity recognition in social streams, which we also focus on in this work.

We propose a simple consolidation technique and explore its positive impact on the word cloud generation. Other potential methods we could employ to improve cloud quality are normalization and co-reference. Normalization [19] applies to many low-frequency terms, and as a result has a low impact with named entities. Also, while normalisation can compare minor spelling mistakes, it typically does not condense highly orthographically different expressions of the same entity. Co-reference requires context to operate – something that is absent in short social media stream messages. Mapping keywords to unambiguous entity references is difficult, but understood [20].

3 Word Clouds Generation with Grouped Named Entities

The process of generating word clouds from social media data is comprised of several subsequent steps:

1. *Data Collection* where underlying documents are aggregated with respect to a user query, profile or trending topic. Often, the whole document collection might be used for a word cloud generation. In this work, we aggregate tweets for word cloud generation with respect to a user query.
2. *Data Preprocessing* where extracted terms or phrases can be clustered, lemmatised or normalised. Documents can be further enriched or annotated with recognized named entities. The aim of this work is to investigate how recognized named entities detected during this phase impact the following word cloud generation.
3. *Word Cloud Generation* where the most relevant and important terms from the underlying collection are selected and consequently a word cloud is generated. Different word selection methods can be applied [3–5].

The goal is to explore how recognized and grouped named entities from the *Data preprocessing* phase affect consequent word cloud generation. Do grouped named entities improve the quality of word clouds in terms of Coverage, Overlap and enhanced access to relevant tweets? Which word cloud generation method gives best results when using named entities? We transform these research questions into the following two hypotheses:

- *H1: Word clouds with grouped recognized named entities improve Coverage, Overlap and Mean Average Precision of generated word clouds.*
- *H2: Word clouds with grouped recognized named entities are more relevant and more diverse with respect to a provided query from the user perspective.*

In the following, we describe a method for grouping recognized named entities from tweets.

3.1 Grouping Named Entities

Conventional named entity recognition is not sufficient due to the nature of Twitter data [21]. Standard named entity recognition approaches do not perform well on tweets because of the error prone structure (misspellings, missing capitalization or grammar mistakes) and their short length. We propose a method that aims to recognize named entities, to link the possible aliases and consequently to generate a word cloud with the recognized and linked named entities. This method can be thought of as a *Data preprocessing* step when generating word clouds over data from social streams.

We combine standard named entity recognition tools with linked data. Alternative names for recognized entities are exploited for term cluster creation for each named entity. A canonical term from an entity term cluster is selected and, if relevant and prominent enough, it is presented in the final word cloud. The method is summarized as follows:

1. Gather a tweet collection – a set of tweets corresponding to a certain trending topic or a query on Twitter.
2. Recognise named entities (**NER**) and disambiguate them (**entity linking**) using the TextRazor service, which performs this task relatively well [21].[1]
3. Using linked data, find alternative names for the recognised entity. We used Freebase's [22] `aliases` field for this. For instance, for the entity *Manchester United FC* the following aliases might be retrieved *Man United*, *Manchester United*, *Man Utd*, *MUFC*, *Red Devils*, *The Reds* or *United*.
4. Perform lemmatisation to group together all the inflicted forms of a word to exploit only the base form of the term.
5. Using the aliases, build a term cluster for each entity, containing e.g. *Manchester United*, *Man U*, *MUFC*.
6. Find canonical names, such as *Manchester United FC*.

[1] See www.textrazor.com.

7. Generate the "condensed" cloud with aggregated counts of entity mentioned frequencies with some word cloud generation technique.

This may be performed as a general-purpose technique, and also to "targeted" streams, e.g. where tweets are filtered based on user-defined criteria such as keywords or spatial regions.

3.2 Distributional Named Entity Grouping

To examine the viability of alternatives to the commercial TextRazor, we also investigate whether unsupervised clusterings can provide entity groups. We run a large corpus of Twitter data sampled from the twitter "garden house" [23] through Brown clustering [24]. This technique captures distributionality through mutual information and uses this as a metric for agglomerative bottom-up hard clusterings of terms found in a corpus. The end result is a binary tree, with terms as leaves, and subtrees holding semantically similar properties (Fig. 1). Paths from the root are described using a bitstring, which has the virtue of being able to estimate semantic similarity of two paths be the length of their common prefix. The transition from distributional to semantic similarity comes from the feature of language that the meaning words can be determined from the words around them [25,26].

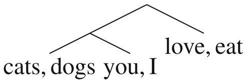


Fig. 1. A binary, hierarchical clustering of semantically similar entries. Each leaf corresponds to a cluster of words (i.e., a "class") and leaves near to their common ancestors correspond to clusters that are similar to each other.

We built a Brown clustering with 2500 classes, an optimal value in many situations [27], from approximately 10^9 English language tweets sampled within 2009–2014. We used langid.py [28] for language filtering. From this hierarchical clustering, we selected candidate entity terms and qualitatively explored how well entity surface forms were grouped by this technique.

Note that increased precision in clustering comes at potentially large computational cost, and so the accuracy of our data is constrained by the amount of time available; Twitter is a challenging environment due to proliferation of word types (i.e., surface forms) in relation to newswire. Additionally, the clustering is for single word types, and so multi-word expressions are out (e.g. *red devils*).

We noted that different expressions of the same entity were represented closely in the resulting tree.[2]

[2] In these output excerpts, the columns are: bitstring; word; frequency in dataset.

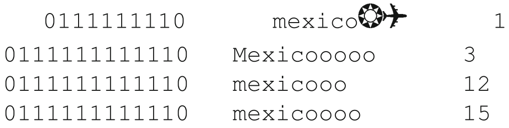
```
11111110110     MUFC    2147
11111110010     #mufc   3131
11111110110     #MUFC   5470

100010010100    mufc    114
```

In this case, *MUFC*, *#mufc* and *#MUFC* are all close to each other, while *mufc* belongs to another cluster. Interestingly, the *mufc* cluster is rich in football team-related terms; it contained 4800 items, the most frequent of which looked like this:

```
100010010100    Milan     5148
100010010100    Rangers   5839
100010010100    Barcelona 12869
100010010100    Chelsea   17589
100010010100    Arsenal   18603
100010010100    United    34614
100010010100    Liverpool 23231
```

The technique is consistently effective at grouping alternative spellings of the same terms, which is very useful in the context of social media. this opens up further opportunities for consolidating the noise intrinsic to social media – frequently used words will grow into clusters containing the majority of their spelling variants. Thus, not only do Brown clusters offer means of disambiguating entity co-references for consolidating terms in tag clouds; it is also possible to consolidate non-named-entity terms, both with regard to their spelling and also their semantics.

```
0111100000  tm         2414
0111100000  tomorow    2420
0111100000  tommorow   7526
0111100000  tmr        8387
0111100000  tmrw       12075
0111100000  tomorrow   571411
```



```
   0111111110       mexico❂✈        1
0111111111110   Mexicoooooo      3
0111111111110   mexicooo         12
0111111111110   mexicoooo        15
```

While – de facto – Textrazor provides useful groundings and Freebase provides useful aliases, we propose use of Brown clustering as a potential source of alternative term generation and capture.

4 Graph-Based Word Cloud Generation

In this section, we describe a graph-based method for generating word clouds with and without entity grouping. The benefits of graph-based word cloud generation are following. First, the method identifies relevant and important keywords in underlying text collections. Several studies have empirically demonstrated the benefits of graph based methods over standard popularity or TF-IDF word cloud generation methods [3,5,29]. Second, graph-based methods allow biasing of word cloud generation toward user preferences or search queries. Our graph-based selection methods firstly transforms terms space into a graph. Then, the stochastic ranking of vertices in the graph is performed. In this work, we consider only global ranking but the proposed methods can be easily applied to biased graph-based ranking e.g., biased towards a user query or a user profile.

4.1 Graph-Based Creation

Extracted terms from underlying tweets are used to build a graph where each term is a graph vertex. If two terms (vertices) co-occur at least α times, we consider these two terms as similar. Eventually, for each similar term pair, two directed edges are generated $t_1 \rightarrow t_2$ and $t_2 \rightarrow t_1$. Hence, edges capture co-occurrence relations between individual terms.

4.2 Graph-Based Ranking

Graph-based ranking of terms simulates a stochastic process i.e., random traversal of the terms in the graph. We use a PageRank-style algorithm [3], but any other algorithm based on random traversal of the graph could be employed. The aim is to estimate the global importance of a term t. If needed, it is possible to bias ranking towards user preferences through a vector of prior probabilities $\boldsymbol{p_p}$. For global graph-based ranking, i.e. without introduced bias, we set each entry in $\boldsymbol{p_p} = \{p_1 \ldots p_{|V|}\}$ to $\frac{1}{|V|}$ where V is the set of all graph vertices. The sum of prior probabilities in $\boldsymbol{p_p}$ is 1. A random restart of stochastic traversal of the graph is assured with a back probability β which determines how often a random traversal restarts and jumps back to a randomly selected (following $\boldsymbol{p_p}$ probability distribution) vertex in the graph. So, the β parameter allows adjustment of bias toward user preferences or to vertices that are globally relevant in the underlying graph. To simulate random traversal of the graph, iterative stationary probability is defined as:

$$\pi(v)^{(i+1)} = (1-\beta)\left(\sum_{u=1}^{d_{in}(v)} p(v|u)\pi^{(i)}(u)\right) + \beta\boldsymbol{p_p} \tag{1}$$

where $\pi(v)^{(i+1)}$ is a probability of visiting node v at time $i+1$, $d_{in}(v)$ is the set of all incoming edges to node v and $p(v|u)$ is a transition probability of jumping from node u to node v. In this work, a transition probability is set to $p(v|u) = \frac{1}{d_{out(u)}}$ for nodes v that have an ingoing edge from node u, otherwise $p(v|u)$ equals 0. The resulting global rank of a term t after convergence is considered as relevance of t i.e.;

$$I(t) = \pi(t) \tag{2}$$

Top-k ranked terms are then used for word cloud generation where the ranking score indicates the prominence of the term in a word cloud.

5 Evaluation

We retrieved available tweets with relevance judgments from TREC2011 microblog collection [30] during August 2014. Although some tweets were not available during retrieval, we compare results over the same corpus. We do not consider the missing tweets as a limitation of our evaluation – see [31]. The relevance judgments for TREC2011 microblog collection were built using a standard pooling technique. For TREC the relevance of a tweet with respect to a query was assessed with a three-point scale; 0: irrelevant tweet, 1: relevant tweet and 2: highly relevant tweet. In this work, we consider both relevant and highly relevant tweets as equally relevant.

5.1 Metrics

We evaluate individual aspects of generated word clouds using the synthetic metrics introduced in [4,5]. The generated word cloud with k terms is denoted as WC_k. A term t links to a set of tweets Tw_t. Tw_{t_q} is the set of all tweets that are associated with a query phrase t_q.

Similarly, $TwREL_{t_q}$ is the set of all relevant tweets for the query t_q.

The first metric is *Coverage*, defined as:

$$\text{Coverage}(WC_k) = \frac{|\cup_{t \in WC_k} Tw_t|}{|Tw_{t_q}|}, \tag{3}$$

where the numerator of the fraction is the size of the union set. The union set consists of tweets associated with each term t from the word cloud WC_k. $|Tw_{t_q}|$ is the number of all tweets that are associated with a query phrase t_q. The metric ranges between 0 and 1. When a Coverage for a particular word cloud WC_k is close to 1, the majority of tweets are "covered" i.e., linked from the word cloud WC_k.

Overlap of WC_k: Different words in WC_k may be linking to the same tweets. The Overlap metric captures the extent of such redundancy. Thus, given $t_i \in WC_k$ and $t_j \in WC_k$, we define the *Overlap*(WC_k) of WC_k as:

$$\text{Overlap}(WC_k) = avg_{t_i \neq t_j} \frac{|Tw_{t_i} \cap Tw_{t_j}|}{\min\{|Tw_{t_i}|, |Tw_{t_j}|\}}, \tag{4}$$

If *Overlap*(WC_k) is close to 0, then the intersections of tweets annotated by depicted words are small and such word clouds are more diverse.

Relevance of WC_k: Expresses how relevant the words in WC_k are to the query phrase t_q. We compute a relevance of a word cloud WC_k in the following fashion:

$$Relevance(WC_k) = avg_{t \in WC_k} \frac{|Tw_t \cap TwREL_{t_q}|}{|Tw_t|}, \tag{5}$$

The more Tw_t and $TwREL_{t_q}$ overlap, the more related t is to t_q. When $Tw_t \subseteq TwREL_{t_q}$, then t can be perceived as more specific sub-category of the original query t_q.

However, the Relevance measure does not capture ordering differences of words within the cloud and considers each term as a single query. The assumption that terms depicted in the cloud are of equal importance is often invalid. We believe that word weights and their order is an important aspect of word clouds where better ranked terms might be more visible i.e., larger font size or better position.

Further, we measure Mean Average Precision metric [5] for the evaluation of word clouds as follows:

We consider a generated word cloud as a query which should retrieve relevant tweets with respect to the query. Therefore, a better word cloud should link to more relevant tweets with respect to the query. We measure this as follows:

1. For given terms and corresponding weights of a word cloud WC_k, create a query vector Q_{WC_k} with normalized weights. Each entry of the query vector Q_{WC_k} represents the importance of a term from the word cloud WC_k with the normalized weight i.e., more important terms from the word cloud are represented with higher weights.
2. Rank and retrieve top-k tweets matching a given query Q_{WC_k}
3. Measure mean average precision(MAP) where each relevant tweet from TREC2011 microblog collection is considered a positive.

Ranking of relevant tweets with respect to a given query Q_{WC_k} is computed with standard information retrieval function OKAPI BM25 which can be defined as:

$$S(tw, Q_{WC_k}) = \sum_{q_i \in Q_{WC_k} \cap tw} c(q_i, Q_{WC_k}) \cdot TF(q_i, tw) \cdot IDF(q_i) \tag{6}$$

where

$$TF(q_i, tw) = \frac{f(q_i, tw) \cdot (k_1 + 1)}{f(q_i, tw) + k_1 \cdot (1 - b + b \cdot \frac{|tw|}{avgtwl})}$$

$$IDF(q_i) = \log \frac{N - n(q_i) + 0.5}{n(q_i) + 0.5}$$

and $f(q_i, tw)$ is a q_i term frequency within a tweet tw, $|tw|$ is the length of a given tweet tw, $avgtwl$ is average length of tweet within the corpus, N is a total number of tweets in the corpus and $n(q_i)$ is the number of tweets that contain the term q_i. To capture the importance of a word from the generated word cloud, we multiply the whole relevance score for a given term with the word cloud weight $c(q_i, Q_{WC_k})$ for the given term q_i. The function $c(q_i, Q_{WC_k})$ returns a weight of the term q_i from the query vector Q_{WC_k} which corresponds to the term weight from the word cloud WC_k. We set the same values for parameters $k_1 = 1.2$ and $b = 0.75$ as in [32].

We measured the average precision at K for the retrieved top K list of ranked tweets with respect to the given word cloud. Further, we measured the MAP for all generated word clouds. The average precision of top K ranked tweets with respect to the word cloud is calculated as follows:

$$AP@K(Q_{WC_k}) = \frac{\sum_k^K (P(k) \cdot rel(k))}{\#relevanttweets}$$

where $P(k)$ is the precision at k-th position in the ranked top K list and $rel(k)$ is 1 if the tweet at rank k is relevant, otherwise $rel(k)$ is 0 and $\#relevanttweets$ is the number of relevant tweets within the top K list. MAP is defined as:

$$MAP@K = \frac{\sum_{Q_{WC_k} \in AWC_k} AP@K(Q_{WC_k})}{|AWC_k|}$$

where AWC_k is the set of all generated word clouds and $AP@KQ_{WC_k}$ is average precision for the given word cloud Q_{WC_k}. In this work, we measure MAP at 30 under the assumption that it represents a reasonable cutoff for the number of relevant tweets similar to the approach in [30].

5.2 Baseline Method

PageRank Exploiting Only Extracted Terms (PgRankTerms). This method was originally proposed in [3] to estimate tag relevance wrt. a certain query, and it outperformed several tag selection approaches in terms of relevance. In this work, the method estimates global terms importance within the graph created from the pooled tweets for the individual query from TREC2011 microblog collection. The β parameter is set to 0.85 (recommended value for a Pagerank algorithm). Due to the short nature of tweets, threshold α for edge creation between individual terms is set to 0. Shorter texts lead to small numbers of co-occurring terms, which consequently leads to a sparse graph (Fig. 2).

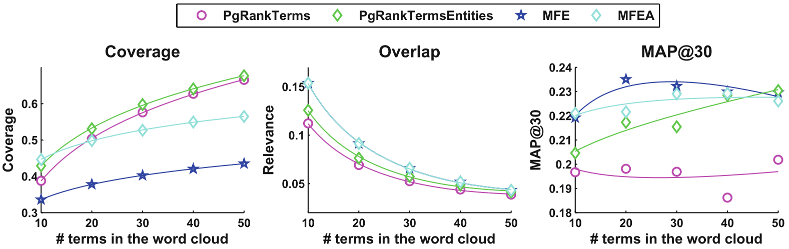


Fig. 2. Coverage, overlap, and mean average precision for word clouds of various sizes generated for queries from TREC2011 microblog collection.

5.3 Entity Based Methods

Most Frequent Entities (MFE). This method selects only recognized entities as defined in Sect. 3.1. The method provides a list of entities sorted by frequency in descending order, selecting top-k most popular entities.

Most Frequent Entities with Grouped Aliases (MFEA). This method selects only recognized entities and associated Freebase aliases as defined in Sect. 3.1. The method provides a list of entities sorted by frequency in descending order.

PageRank Exploiting Extracted Terms, Entities and Grouped Aliases (PgRankTermsEntities). This method estimates the global importance of terms and recognized named entities within the graph created from the extracted terms, recognized named entities and grouped Freebase aliases from pooled tweets for the individual query from TREC2011 microblog collection. The parameters are set to the same values as in the baseline method.

5.4 Results

We performed the evaluation on queries from TREC2011 microblog collection [30]. The MFE method has the worst Coverage ranging from 35 % for word clouds with 10 terms to 45 % for word clouds with 50 terms. MFEA has better Coverage with approximately 10 % absolute improvement over the MFE method. The baseline method PgRankTerms attains greater Coverage than MFE and MFEA methods. The reason for higher Coverage of PgRankTerms is that entity mentions do not occur enough in tweets to outperform other extracted words.

However, when extracted words are combined with grouped named entities like in PgRankTermsEntities, the improvements in Coverage are highest. The PgRankTermsEntities method outperforms all other word cloud generation methods. PgRankTermsEntities improves Coverage with respect to PgRankTerms and MFEA because it groups entity synonyms e.g. USA, US and America and represent them with the canonical entity name United States of America. In addition, it selects the most important terms which are not referring to named entities e.g., *#service*, *#jobs* for the query *BBC World Service staff cuts*. The relative improvements in comparison to PgRankTerms are 11 % for 10 terms, 6 % for 20 terms, 4 % for 30 terms and 2 % for 40 and 50 terms word clouds. Coverage improvements decrease as word clouds increase in size because the number of relevant/prominent recognized named entities in the underlying graph is lower. These results support the hypothesis H1: that grouping named entities improves the Coverage of word clouds.

Word cloud generation methods which exploit named recognized entities improve MAP. PgRankTermsEntities, MFE and MFEA outperform PgRankTerms in terms of MAP. The relative improvements of PgRankTermsEntities in comparison to PgRankTerms are 4 % for 10 terms, 10 % for 20 terms, 9 % for 30 terms, 23 % for 40 and 14 % for 50 terms word clouds. Thus, word clouds with named recognized entities improve access to the relevant tweets of the corpus which validates the H1 hypothesis. The main reason for the attained improvements is that almost 89 % of all relevant tweets from TREC2011 microblog collection contain at least one recognized entity.

Similarly, 31 % of all relevant tweets contain at least one Freebase alias (with minimal length of 4 characters). Comparing all pooled tweets from the

TREC2011 microblog collection 77 % contain recognized named entities and 28 % of tweets contain at least one Freebase alias. Further, linking entity synonyms increases both Coverage and also the prominence of the named entity in the word cloud. Thus, it is more likely that the named entity will be represented in the word cloud and, if relevant for the query, it will improve access to the relevant tweets.

Improved access to relevant tweets and enhanced Coverage of word clouds can be attained through a combined selection of terms and recognized named entities. Thus, for enhanced word cloud generation it is important to combine recognized and grouped named entities with relevant and prominent terms from the underlying dataset.

The methods exploiting recognized named entities do have higher Overlap than the PgRankTerms method. We consider this finding interesting and unanticipated. The increased redundancies in the generated word clouds are caused by imperfect NER tools. In particular, tweets with an ambiguous name entity such as *BBC News Service* link to several semantically similar entities such as *BBC*, *BBC News*, *BBC NEWS Service*, which might lead to higher Overlap scores. Further, detected Freebase aliases might often increase Overlap for the similar reason e.g., alias *us* for *United States* covers many irrelevant tweets. To minimize the impact of ambiguous aliases we restrict the alias detection to a minimum length of 4 characters and the alias may not be a stop word.

Lemmatisation also had a positive effect on word cloud generation. Lemmatising terms to group them improves Coverage 1.75 % above the baseline, and 3 % for the PgRankTermsEntities. Similarly, MAP improves with an increase of 11 % for PgRankTermsEntities and 7 % for the baseline technique. The negative impact of lemmatisation on word cloud generation is higher Overlap (decreased diversity of word clouds), with an increase of 3 % using the baseline technique. As the result is overall positive, we included lemmatisation as a preprocessing step for all cloud generation methods.

5.5 Diversification

To overcome the problems introduced by higher redundancy in word clouds, we investigate how to maximize global relevance as well as diversity of selected terms. Instead of following greedy diversification approaches, we take a unified approach of ranking global relevance together with the diversification objective. We use the DivRank algorithm [33] which assumes that transition probabilities change over time following the "rich gets richer" principle. The transition probability of visiting a node (term) A from other nodes is reinforced by the number of times a node A has been already visited. This reinforcement aspect is defined as $N_T(v)$ and it captures the number of visits to a node v until time T. Let us assume that at time T a random walk is at node u, then at time $T+1$ the walk proceeds to a node v with a transition probability $p_T(v|u)$ which is proportional to: $p(v|u) \cdot N_T(v)$. Hence, the general form of the DivRank algorithm can be declared as follows where a transition probability from a node u to node v at time T is:

$$p_T(v|u) = (1-\beta)\left(\frac{p_0(v|u)\cdot N_T(v)}{\sum_{v\in V} p_0(v|u)\cdot N_T(v)}\right) + \beta p_P$$

where p_P is the vector of prior probabilities and $p_0(v|u)$ is an initial estimation of transition probability which is equivalent to definition in Pagerank algorithm (see Sect. 4.2).

It is challenging to approximate $N_T(v)$. A simple approximation might use $p_T(v)$ to estimate $N_T(v)$. Authors of DivRank algorithm [33] denote such an approximation as pointwise DivRank and is defined as follows:

$$p_T(v|u) = (1-\beta)\left(\frac{p_0(v|u)\cdot p_T(v)}{\sum_{v\in V} p_0(v|u)\cdot p_T(v)}\right) + \beta p_R$$

In the following, we present an impact of DivRank algorithm on the word cloud generation with grouped named entities. Figure 3 shows that with diversified word cloud generation, Overlap decreases. The relative improvements of DivRankTermsEntities outperforms the PgRankTerms baseline are 14 % for 10 terms, 14 % for 20 terms, 12 % for 30 terms, 11 % for 40 and 12 % for 50 terms word clouds. The DivRankTermsEntities method significantly decreases Overlap in comparison to the PgRankTerms baseline (Wilcoxon signed-rank test, $p = 0.000094$) The improvements are even more significant with respect to PgRankTermsEntities method with 24 % for 10 terms, 22 % for 20 terms, 20 % for 30 terms, 19 % for 40 and 18 % for 50 terms word clouds. In contrast, diversified word cloud generation significantly improves Coverage of word cloud generation. The improvement is statistically significant with respect to the baseline method PgRankTerms (Wilcoxon signed-rank test, $p = 0.0363$). The mean of relative improvements DivRankTermsEntities with respect to PgRankTermsEntities (the best performing method when measuring Coverage) is 2.35 %.

Diversified word cloud generation from grouped and recognized named entities combined with extracted words decreases significantly Overlap, improves significantly Coverage and improves access to relevant tweets. This validates hypothesis $H1$.

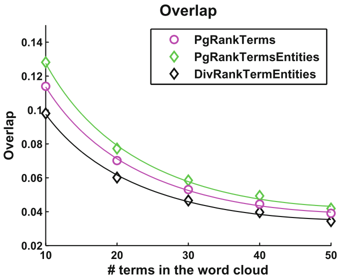


Fig. 3. Overlap for diversified word clouds with the method DivRankTermEntities of various sizes generated for queries from TREC2011 microblog collection.

5.6 Crowdsourced Evaluation

In order to verify the findings from empirical evaluation of word clouds with different synthetic metrics, we designed a crowdsourced user evaluation of generated word clouds. We selected 8 queries from TREC2011 microblog collection for which we generated word clouds with DivRankTermEntities and PgRankTerms methods. We included 4 queries where the enhancement of MAP for word clouds with named entities with respect to the baseline was the greatest (denoted as Impr. MAP). Similarly, we added 4 word clouds for queries where the Overlap has been decreased the most with respect to the baseline (denoted as Impr. diversity ($\downarrow$ *Overlap*)). The answers sought by the user evaluation are twofold. First, are word clouds with named entities perceived as more relevant and diverse by the end users? Second, do measured synthetic metrics correlate with the ratings of relevance and diversity by users?

Participants were asked to view a pair of word clouds, a set of tweets related to a certain query, and a related Wikipedia article. Their task was to determine which word cloud was more relevant and which was more diverse. The user was asked to rate the relevance and diversity of an individual word cloud with respect to the query on a Likert scale of 1 to 5 (Rating 1: word cloud A is very relevant/diverse to the pertaining query; Rating 3 - both word clouds are equally relevant/diverse to the pertaining query; and Rating 5 - word cloud B is very relevant/diverse to the pertaining query). We altered assignment of word clouds with named entities to either word cloud A or B for each query to prevent user bias that "word cloud A (with named entities) is always more relevant and diverse".

Non-grouped vs. Entity-Grouped Clouds. Each word cloud pair was compared using 20 ratings from distinct users. For 7 out of 8 word clouds, the average ratings of relevance and diversity favoured word clouds generated with automatically grouped named entities. For simplicity's sake, in the following we refer to word clouds generated with grouped entities as word cloud B; positive ratings are those over 3.0.

From 160 distinct relevance ratings, 89 were positive towards word clouds with named entities, 27 were neutral ratings and 44 were more towards the baseline generated word clouds (see Fig. 4). Similarly for diversity ratings, 73 were positive towards word clouds with named entities, 51 were neutral ratings and 36 were more towards the baseline generated word clouds.

To further compare differences between word clouds generated by the baseline and clouds with grouped named entities, we performed a statistical significance test. The null hypothesis is that user ratings are normally distributed with mean 3.0, i.e., word clouds generated by the DivRankTermEntities and PgRankTerms methods are rated as equally relevant and equally diverse. For the relevance judgments, we found that word clouds generated by the DivRankTermEntities method are significantly better rated than the baseline word clouds ($p = 0.00062$, one sample t-test). Similarly, we determined that word clouds generated by the DivRankTermEntities are significantly better rated for diversity with respect

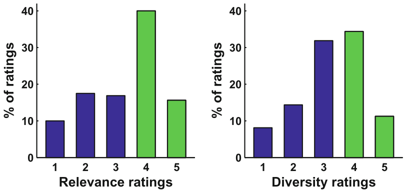


Fig. 4. Green bins (ratings 4 and 5) in the histograms indicate positive rating towards word clouds with grouped named entities. Ratings 1 and 2 indicate user preference towards the baseline word clouds and rating 3 represents that the baseline and the word cloud with grouped entities are equally relevant or diverse.

to the baseline method ($p = 0.003$, one sample t-test). These findings support hypothesis $H2$: users find word clouds with grouped entities more relevant and diverse than those with no entity grouping.

Table 1. Three distinct groups statistics which were created according to the measured levels of synthetic metrics.

Group	# clouds	min δ	mean δ
Impr. MAP	4	0.14	0.26
Impr. diversity ($\downarrow$ *Overlap*)	4	-0.02	-0.023
Decr. MAP &Overlap	2	-0.02	-0.133

Synthetic Metrics vs User Perception. The second goal of the user evaluation is to determine whether word clouds with higher levels of measured synthetic metrics are rated by users as more relevant and diverse or vice versa. We focused on the MAP and Overlap metrics.[3]

To determine the correlation between user judgments and synthetic metrics, we have created 3 different groups (see Table 1). We exploit the same two groups of word clouds Impr. MAP and Impr. diversity ($\downarrow$ *Overlap*) as in Sect. 5.6. In addition, we added a group Decr. MAP &Overlap with two clouds where levels of MAP and Overlap were lower than the baseline word clouds. For each group, we report a minimum δ value which is a minimal difference between measured levels of the particular metric for word clouds with grouped entities and the baseline. Hence a minimum δ is a threshold of measured synthetic metric whether to include a word cloud into the particular group. For instance, the threshold $\delta = 0.14$ for the Impr. MAP group indicates that only those word cloud pairs

[3] Validation by users of the third metric introduced in [4], Coverage, is only possible with an interactive user evaluation. Hence, we do not include "coverage assessment" of word clouds in this study.

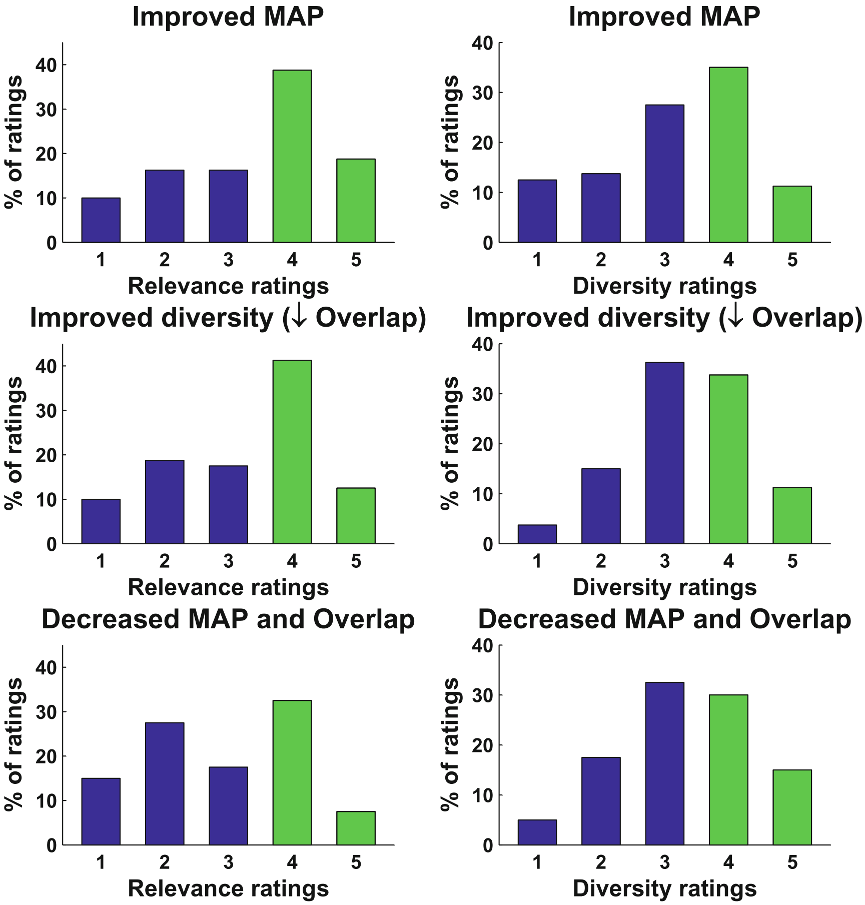


Fig. 5. Aggregated user ratings for three distinct groups of word clouds categorized according to the measured levels of synthetic metrics.

where the improvements of MAP are at least 0.14 (comparing the baseline and DivRankTermEntities methods) are included. The mean of δ expresses the average value of differences in metric values for each word cloud pair in the group, e.g., the average improvements of MAP in the group Impr. MAP is 0.26. Note that negative values of δ reflect cases where the metric is lower than baseline. For Decr. MAP &Overlap group, we only report levels of MAP due to substantial differences in comparison to the Overlap levels which have very slight differences between the baseline and the word clouds with grouped entities.

When all the ratings aggregated altogether from three groups, word clouds with grouped entities are still rated significantly more relevant ($p = 0.0046$, one sample t-test) and diverse ($p = 0.00047$, one sample t-test) than the baseline.

The relation between created groups and user judgments is presented in Fig. 5. Users rated word clouds with higher MAP as more relevant. Of 80 ratings, 46 (57.5 %) indicated that word clouds with grouped named entities are more relevant than the baseline. Conversely, for the word clouds with the decreased MAP and Overlap, only 40 % of the ratings indicatie preference towards word clouds with grouped named entities. Hence, word clouds with higher MAP get 17.5 % more positive ratings (4 or 5 ratings) than the baseline. The difference is even more pronounced for "rating 5 - much more relevant than the baseline word cloud", where Decr. MAP &Overlap group attained only 7.5 % from all ratings, whereas the Improved MAP group attained 18.75 %. Therefore, we can conclude that word clouds with grouped named entities which attain higher levels of MAP are more likely to be better rated in terms of relevance by users.

When measuring diversity, word clouds from the Impr. diversity(↓ *Overlap*) group were slightly more rated as "equally or more diverse than word clouds generated by the baseline" than other groups. In particular, with Impr. diversity(↓ *Overlap*), we observed a decreased number of ratings, expressing that the baseline word cloud is much more diverse (3.75 % for Impr. diversity (↓ *Overlap*) group and 12.5 for Impr. MAP). However, when looking at the decreased Map and Overlap group, the distribution of the ratings is fairly even. Hence, the Overlap metric is not a suitable predictor of user diversity ratings. This might be because the relative improvements of Overlap are too subtle to produce observable differences in user judgements of diversity. In order to attain more significant differences of Overlap, we believe that larger collections of tweets (retrieved w.r.t more general information needs) should be employed.

On the other hand, 46.3 % of word clouds with improved MAP and 45 % of word clouds from Decr. MAP & Overlap group were rated as more diverse than the baseline. Therefore, users rating word clouds with grouped entities have tend to find them more diverse than word clouds with no grouping.

6 Discussions, Limitations and Future Work

False positives during entity recognition may have reduced relevant ratings. For instance, a word cloud generated for the query "*Super Bowl, seats*" contained "*Super (2010 American film)*" which is irrelevant for this query. Similarly, for "*Kubica crash*", the entity "*crash bandicoot*" ended up in the word cloud. Unsupervised semantic clusterings (Sect. 3.2) may serve to better differentiate such cases.

Some word clouds generated with the PgRankTermsEntities suffered from increased Overlap. This was partially caused by imprecise named entity disambiguation where ambiguous named entities were not grounded correctly. Therefore, the quality of word clouds with grouped named entities is bounded by the precision of named entity annotation tools.

Evaluating word clouds with crowdsourced user evaluation is a challenging task due to uncertainty of reliability and quality of user ratings. In our pilot study, we aimed to ensure the quality of user ratings with pre-filtering quiz questions. However, we have observed that for test questions where users were asked to rate word cloud diversity (one cloud was supposed to be more diverse) many participants disagreed. Due to the subjective nature of the task, we disregarded a user "qualifying" phase (as is often best practice in crowdsourcing [34]) and instead aimed to collect more user ratings and observe aggregated ratings. To further ensure the quality of the ratings, we accepted ratings only from participants in English-speaking countries, as word clouds were generated from tweets written in English.

As future work, we would like to design a hybrid clustering method which would combine knowledge from linked data repositories (e.g., Freebase) with probabilistic context of terms (the similar intuition as in Brown clustering). The hybrid approach could further improve accuracy of entity grouping as context of terms could minimize incorrect grouping of aliases. Further, the future work would explore how grouped named entities could improve a personalized word cloud generation [5], mainly whether grouped named entities could alleviate a sparsity problem when expressing user preferences.

7 Conclusion

Generating word clouds from social streams is a difficult task. Because users often discuss the same entity using multiple aliases, the utility of word clouds becomes degrades when this complex and high-variety data is used directly. Consequently, methods of unifying these variations become necessary, in order to get accurate counts. Accordingly, we propose techniques for grouping aliases that refer the same entity, and for representing these groups using a canonical term. The method improves the coverage of word clouds and access to the relevant content.

This variety also leads to redundant terms, that must be clustered together, in order to improve the precision of the cloud. Due to imperfect state-of-the-art named entity recognition on social media, redundancy of terms in word clouds often remains. This makes it necessary diversify terms. We found that unsupervised term extraction and clustering techniques (such as Brown clustering) can be used to automatically identify similar and co-referent terms, beyond the lists available through commercial and third-party ontology services. It was then demonstrated that our technique not only significantly decreases redundancy, but also leads to significantly higher coverage than baseline word cloud generation, leading to better word clouds and therefore improved information access. Combined, these factors alleviate problems with in clouds from social media.

Naturally, this leads to questions about how evaluation can be tested. Earlier, we hypothesised that word clouds with grouped named entities are significantly more relevant and diverse than word clouds with no entity grouping. This was evaluated extrinsically against the crowd, with reported user experiences supporting the hypothesis. Further, word clouds with grouped named entities that score higher MAP are more likely to be rated as relevant by users.

Finally, we compared these gold-standard human judgments to a proposed synthetic cloud evaluation metric. It was shown that this previously-proposed MAP metric for automatic cloud evaluation predicts extrinsic human evaluations of cloud quality. Thus, when designing word clouds, the MAP metric should be used as a quality predictor of the cloud generation technique, enabling automatic assessment of word cloud quality without a human in the loop.

Acknowledgments. This work was partially supported by the European Union under grant agreement No. 611233 Pheme.

References

1. Kuo, B.Y., Hentrich, T., Good, B.M., Wilkinson, M.D.: Tag clouds for summarizing web search results. In: Proceedings of the Conference on the World Wide Web (WWW), pp. 1203–1204. ACM (2007)
2. Miotto, R., Jiang, S., Weng, C.: eTACTS: a method for dynamically filtering clinical trial search results. J. Biomed. Inf. **46**, 1060–1067 (2013)
3. Leginus, M., Dolog, P., Lage, R.: Graph based techniques for tag cloud generation. In: Proceedings of the ACM Conference on Hypertext and Social Media, pp. 148–157, ACM (2013)
4. Venetis, P., Koutrika, G., Garcia-Molina, H.: On the selection of tags for tag clouds. In: Proceedings of the Conference on Web Search and Data Mining (WSDM), pp. 835–844. ACM (2011)
5. Leginus, M., Zhai, C., Dolog, P.: Personalized generation of word clouds from tweets. J. Assoc. Inf. Sci. Technol. (2015)
6. Tufekci, Z.: Big questions for social media big data: representativeness, validity and other methodological pitfalls. In: Proceedings of the International Conference on Weblogs and Social Media (ICWSM), AAAI, pp. 505–514 (2014)
7. Bernstein, M.S., Suh, B., Hong, L., Chen, J., Kairam, S., Chi, E.H.: Eddi: interactive topic-based browsing of social status streams. In: Proceedings of the Annual Symposium on User Interface Software and Technology (UIST), pp. 303–312. ACM (2010)
8. Lage, R., Dolog, P., Leginus, M.: The role of adaptive elements in web-based surveillance system user interfaces. In: Dimitrova, V., Kuflik, T., Chin, D., Ricci, F., Dolog, P., Houben, G.-J. (eds.) UMAP 2014. LNCS, vol. 8538, pp. 350–362. Springer, Heidelberg (2014)
9. Rout, D., Bontcheva, K., Hepple, M.: Reliably evaluating summaries of Twitter timelines. In: Proceedings of the AAAI Workshop on Analyzing Microtext, AAAI, pp. 64–71 (2013)
10. Derczynski, L., Maynard, D., Aswani, N., Bontcheva, K.: Microblog-genre noise and impact on semantic annotation accuracy. In: Proceedings of the ACM Conference on Hypertext and Social Media, pp. 21–30. ACM (2013)
11. Maynard, D., Greenwood, M.A.: Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In: Proceedings of the Conference on Language Resources and Evaluation (LREC), Reykjavik, Iceland, ELRA (2014)
12. Leginus, M., Derczynski, L., Dolog, P.: Enhanced information access to social streams through word clouds with entity grouping. In: Proceedings of the conference on Web Information Systems and Technologies (WEBIST) (2015)
13. Finin, T., Murnane, W., Karandikar, A., Keller, N., Martineau, J., Dredze, M.: Annotating named entities in twitter data with crowdsourcing. In: Proceedings of the Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, ACL, pp. 80–88 (2010)

14. Hogan, A., Zimmermann, A., Umbrich, J., Polleres, A., Decker, S.: Scalable and distributed methods for entity matching, consolidation and disambiguation over linked data corpora. Web Semant. Sci. Serv. Agents World Wide Web **10**, 76–110 (2012)
15. Hu, Y., Talamadupula, K., Kambhampati, S., et al.: Dude, srsly?: The surprisingly formal nature of Twitter's language. In: Proceedings of the International Conference on Weblogs and Social Media (ICWSM), AAAI (2013)
16. Baldwin, T., Cook, P., Lui, M., MacKinlay, A., Wang, L.: How noisy social media text, how diffrnt social media sources. In: Proceedings of the International Joint Conference on Natural Language Processing (IJCNLP), pp. 356–364 (2013)
17. Bontcheva, K., Derczynski, L., Funk, A., Greenwood, M.A., Maynard, D., Aswani, N.: TwitIE: an open-source information extraction pipeline for microblog text. In: Proceedings of the conference on Recent Advances in Natural Language Processing (RANLP), pp. 83–90 (2013)
18. Baldwin, T., Kim, Y.B., de Marneffe, M.C., Ritter, A., Han, B., Xu, W.: Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition. ACL-IJCNLP **2015**, 126 (2015)
19. Han, B., Baldwin, T.: Lexical normalisation of short text messages: makn sens a #twitter. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL), ACL, pp. 368–378 (2011)
20. Augenstein, I., Gentile, A.L., Norton, B., Zhang, Z., Ciravegna, F.: Mapping keywords to linked data resources for automatic query expansion. In: Proceedings of the Second International Workshop on Knowledge Discovery and Data Mining Meets Linked Open Data, pp. 9–20 (2013)
21. Derczynski, L., Maynard, D., Rizzo, G., van Erp, M., Gorrell, G., Troncy, R., Petrak, J., Bontcheva, K.: Analysis of named entity recognition and linking for tweets. Inf. Process. Manag. **51**, 32–49 (2015)
22. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: a collaboratively created graph database for structuring human knowledge. In: Proceedings of the Meeting of the Special Interest Group on Management of Data (SIGMOD), pp. 1247–1250. ACM (2008)
23. Kergl, D., Roedler, R., Seeber, S.: On the endogenesis of Twitter's Spritzer and Gardenhose sample streams. In: Proceedings of the Conference on Advances in Social Networks Analysis and Mining (ASONAM), pp. 357–364. IEEE (2014)
24. Brown, P.F., Desouza, P.V., Mercer, R.L., Pietra, V.J.D., Lai, J.C.: Class-based n-gram models of natural language. Comput. Linguist. **18**, 467–479 (1992)
25. Wittgenstein, L.: Philosophical Investigations. Basic Blackwell, London (1953)
26. Lesk, M.: Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In: Proceedings of the Annual International Conference on Systems Documentation (SIGDOC), pp. 24–26. ACM (1986)
27. Derczynski, L., Chester, S., Bøgh, K.S.: Tune your brown clustering, please. In: Proceedings of the Conference on Recent Advances in Natural Lang Processing (RANLP) (2015)
28. Lui, M., Baldwin, T.: langid.py: an off-the-shelf language identification tool. In: Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL), vol. 3, pp. 25–30. ACL (2012)
29. Wu, W., Zhang, B., Ostendorf, M.: Automatic generation of personalized annotation tags for Twitter users. In: Proceedings of the annual meeting of the Association for Computational Linguistics (ACL), ACL, pp. 689–692 (2010)

30. Ounis, I., Macdonald, C., Lin, J., Soboroff, I.: Overview of the TREC-2011 microblog track. In: Proceedings of the Text REtrieval Conference (TREC) (2011)
31. McCreadie, R., Soboroff, I., Lin, J., Macdonald, C., Ounis, I., McCullough, D.: On building a reusable Twitter corpus. In: Proceedings of the meeting of the Special Interest Group in Information Retrieval (SIGIR), pp. 1113–1114. ACM (2012)
32. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval, vol. 1. Cambridge University Press, Cambridge (2008)
33. Mei, Q., Guo, J., Radev, D.: Divrank: the interplay of prestige and diversity in information networks. In: Proceedings of the meeting of the Special Interest Group on Knowledge Discovery and Data Mining (SIGKDD), pp. 1009–1018. ACM (2010)
34. Sabou, M., Bontcheva, K., Derczynski, L., Scharl, A.: Corpus annotation through crowdsourcing: towards best practice guidelines. In: Proceedings of the conference on Language Resources and Evaluation (LREC), ELRA (2014)

Compatibility Checking of REST API Based on Coloured Petri Net

Li Li(✉) and Wu Chou

Huawei Shannon IT Lab, Huawei, Bridgewater, NJ, USA
{li.nj.li,wu.chou}@huawei.com

Abstract. One of the underutilized advantages of REST API is extensibility. Extensibility allows a REST API to make certain changes to its resource representation, identification, interaction, and connection without breaking its clients. A client can cope with these changes in the REST API through hypertext-driven interactions - an iterative process in which the client can determine the resource identification based on its representation, utilize its identification to determine the interactions, and follow the interactions to determine its connections. However, our analysis reveals that there are limitations to the flexibility of the hypertext-driven navigation due to the dependency between these interaction layers in the REST API, and there is a critical need to determine if two REST APIs are compatible for the client. To address this issue, we describe a structured approach to REST client modelling that decomposes a REST client into two functional components: client oracle and client agent. From this client model, we derive a formal definition of compatibility based on the REST Chart representation of the REST API, and an efficient algorithm is developed to verify the compatibility between two REST Charts. A prototype system has been implemented, and the preliminary experimental results show that the approach is feasible and promising.

Keywords: REST API · Coloured petri net · REST chart · Compatibility checking · Client oracle · Client agent

1 Introduction

In recent years, the REST architectural style [1] has been widely applied in API design for multiple areas, including Real-Time Communications [2], Cloud Computing [3], and Software-Defined Networking (SDN) [4]. It is an efficient and flexible way to access and integrate large-scale complex systems which may have many interacting REST APIs to provide their resources as service for applications. However, in large scale distributed systems, these interacting REST APIs are evolving rapidly and under frequent updates. An acute problem in REST based system is how to efficiently migrate REST clients to keep up with the rapid updates and service variations that are frequently made to the numerous REST APIs - a situation may cause the backward compatibilities to break.

For example, OpenStack is an open source IaaS platform that currently supports 14 REST APIs [5], implemented by over 30 components - managing compute, storage,

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 25–43, 2016.
DOI: 10.1007/978-3-319-30996-5_2

network, VM image, and identity services. To maintain backward compatibility, OpenStack simultaneously supports different versions of the same API - for example, there are 3 versions of Compute API, 2 versions of Block Storage API, 2 versions of Identity API, and 2 versions of Image API. In fact, the actual number of REST APIs in an OpenStack installation can be even much higher if we count the third party REST APIs.

On the other hand, OpenStack development follows a rapid release cycle, and the development cycle of different versions of OpenStack often overlap in time [6]. For example, the Grizzly and Havana versions of OpenStack overlap each other by 6 months, and each has 6 releases within 11 months respectively. The Havana and IceHouse versions of OpenStack overlap each other by 6 months, and the IceHouse version of OpenStack has 4 releases within 6 months. Each new release can introduce changes to its REST APIs that can break the REST clients originally programmed for the previous release.

For example, version 2.0 of Floodlight REST API in OpenStack made significant changes to version 1.0. Version 2.0 allows a client to traverse to a port resource in one of the two paths: (1) *initial* → *networks* → *ports* → *port* or (2) *initial* → *ports* → *port*. The first path is in version 1.0 while the second path is not. A version 1.0 client looking for a port resource in the version 2.0 REST API can follow the first path without change, but it cannot take advantage of the second path, unless it is pre-programmed to take alternative paths. Version 2.0 also introduces changes that a version 1.0 client does not recognize, such as renaming the attachment resource in version 1.0 to the device resource. A version 1.0 client looking for an attachment resource will not find it in version 2.0, unless it is reprogrammed to look for device resource.

To find such incompatibilities between versions of a REST API is therefore important to design and port clients in face of the frequent changes and updates. This task is difficult because a REST API permits 4 types of changes on its resources, and it can happen at the layers of its resource representation, identification, interaction, and connection. These changes can occur in any combination and each of them may require a special method to deal with. Even if a well-designed REST API can navigate its clients through some of the changes with hypertext-driven interactions, e.g. content negotiation and URI redirection, a client is still at risk of not being able to reach its targeted resource if the changes are beyond its programmed flexibility. Although it might be possible to program more flexibility in the client to reduce such risks with a new REST API, it is difficult to know what flexibility is necessary ahead of changes, and adding unnecessary flexibility can impact the performance. For these reasons, we propose a formal and efficient way to check and verify the compatibility for the client between two REST APIs.

While compatibility checking can be based on either the *implementations (behaviours)* or the *descriptions (structures)* of the REST APIs after they are implemented, this paper takes an approach to allow the compatibility checking and verification for the REST APIs at their design time before they are implemented. This approach requires that the REST APIs be described and represented in a more formal machine-readable way, and it is justified for several reasons: (1) a formal description can be read by both users and machines; (2) a formal description is more accurate than an informal one (such as English); and (3) a formal description can be used to automatically generate

REST API services and clients. The particular formal description adopted by this paper is REST Chart [14], a REST service description language and modelling framework based on Coloured Petri Net. Petri Net is a mathematical model with a graphical representation that is easy to visualize for users. With REST Chart, compatibility checking of two REST APIs is transformed into the compatibility checking of two REST Charts that describe the REST APIs. The main contributions of this paper are summarized below:

1. A layered model is described to analyze how changes to the resource representation, identification, interaction, and connection can impact each other, and it is applied to characterize the capabilities and limitations of hypertext-driven navigation in coping with REST API changes.
2. A REST client model that decomposes a REST client into two structural components: client oracle and client agent, to localize and classify the impact of REST API changes on the client implementation.
3. From this client model, a formal definition of compatibility between REST Charts is derived for compatibility checking and verification.
4. We describe an efficient algorithm to find and identify the compatible paths between two REST Charts in hypertext-driven navigation.

The rest of this paper is organized as follows. Section 2 surveys the related work. Section 3 analyzes the 4 types of changes that can be made to the REST API. Section 4 introduces the framework of REST Chart modelling for REST APIs. Section 5 presents the proposed REST client operational model which provides a theoretical basis for the REST Chart based compatibility testing and verification. Section 6 derives the compatibility conditions and the REST Chart comparison algorithm based on the proposed operational model. Section 7 discusses the implementation and experimental results, and the findings of this paper are summarized in Sect. 8.

2 Related Work

Several REST service description languages have been developed since 2009. WADL [7] is an early effort to describe REST services, followed by RAML [8], Swagger [9], RSDL [10], API-Blueprint [11], SA-REST [12], ReLL [13], REST Chart [14], RADL [15], and RDF-REST [16]. All these description languages are encoded in some machine-readable languages, such as XML, and most of them are standalone documents, but a few of them, such as SA-REST, are intended to be embedded within a host language, such as HTML. However, they lack a formal method to efficiently compare different versions of a REST API based on these description languages.

There are several open source Java packages and Web tools [17–19] that compare two WSDL files for WS-* based web services – they include using XML Schema [20] files to identify changes (addition, deletion, modification, and reorder) made to the WSDL elements, such as port types and operations, and the XML elements and attributes. Some tools distinguish changes that will break interface compatibility from those that will not. For example, adding an optional XML element to a XML Schema of an input element for an operation will not invalidate the interface, but adding a required XML element, or changing the type, name, or position of an existing required XML element will.

Moreover, these methods of WSDL comparison for WS-* based web services cannot be applied to compare REST Charts for REST APIs, because a REST Chart is not structured as a WSDL file. Despite that REST Chart is represented as a XML dialect, we cannot use generic XML diff tools to compare REST Charts, because REST Chart has special semantics not understood by these tools.

In addition, there are a few open source tools [21] that compare two XML Schema definitions and identify their differences as changes (addition, deletion, modification, and reorder). These tools can assist the comparison of REST Chart, as well as other service description languages that use XML Schema to define the input and output messages of a service. However, they fall short to provide a generic framework for comparing the compatibility of REST service APIs, which is critically needed for large scale distributed software systems.

Petri Net is a mathematical model that has been used to model and analyze concurrent and distributed communication and computation systems [22–24]. A Petri Net consists of places and transitions connected by arcs. A place can store tokens, which can represent data, messages or conditions, and a transition can move tokens between places to model computation, processing or inference. Coloured Petri Net assigns colours (data types) to places and tokens, such that it can realistically model systems in which tokens can have complex structures. However, it lacks a structured way to check and verify the compatibility of Petri Net in RESTful interactions.

3 A Layered Model of REST API Changes

Hypertext-driven navigation, also called “hypermedia as the engine of application state” in [1], is an iterative process in which a client uses representations (e.g. XML) to select identifications (e.g. URI), utilizes identifications to determine interactions with resources (e.g. HTTP), and follows the interactions to obtain representations and connections. This process is illustrated in Fig. 1, where a client moves from resource *a* to resource *c* through three layers of objects: representations, identifications, and connections, using two layers of operations: selections and interactions. At first, the client selects the entry point URI_a. It then uses the appropriate protocol to interact with resource *a* identified by URI_a. After the interaction returns $\mathrm{Hypertext}_\mathrm{a}$, the client selects URI_b from the $\mathrm{Hypertext}_\mathrm{a}$ and the cycle continues until the client reaches the target resource *c*. Because hypertext-driven navigation encourages a client to make its decisions based on current resource states, a REST API can make certain changes at these layers where a client can cope with these changes at runtime through hypertext navigations without modification.

Compared to direct access from a fixed identification, hypertext-driven navigation seems less efficient. For example, navigating to resource *c* from resource *a* requires 4 navigation messages, whereas the direct access to resource *c* from URI_c requires 0 navigation message. However, direct access to a resource is possible only if the resource already exists. If the resource (e.g. *c*) does not exist and must be created by another resource (e.g. *b*), hypertext-driven navigation is the only option. Direct access binds resource identifications to resource connections and makes them difficult to change at runtime. For example, if a REST API advertises URI_c = http://www.example.com/a/b/c for direct access, it removes the possibility for a client to find the changes to the connection at runtime - the only way is to modify the client to use the new URI.

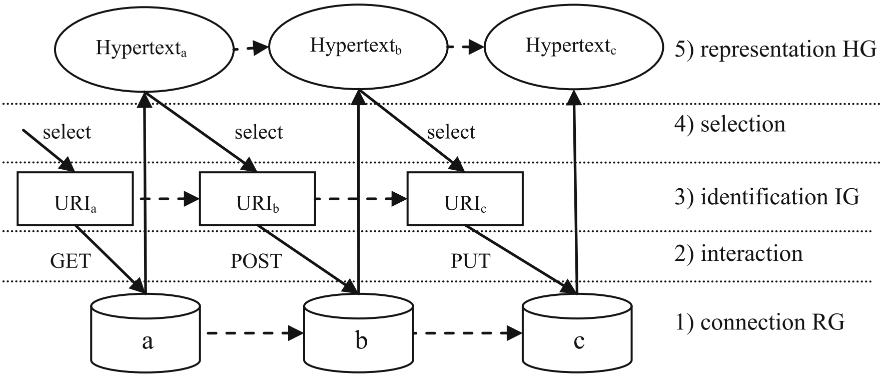


Fig. 1. Layered REST API structure for hypertext-driven navigation.

Although hypertext-driven navigation is a powerful mechanism in the RESTful design framework, it has limitations in dealing with changes in REST APIs. For hypertext-driven navigation to work, a client must recognize the objects and perform the operations in Fig. 1.

If a change introduces unrecognized objects or operations, a client will not be able to navigate through the REST API. In such cases, the client has to be modified in order to work with the new changes. To understand these limitations, we analyze the possible changes at each layer and describe how these changes in one layer can impact the other and affect the clients.

In our analysis, we treat 3 layers of objects as 3 directed graphs so that we can determine how changes in one graph can impact other graphs:

1. Connection Graph *RG*: a directed graph whose nodes are resources and edges are connections between the resources. In particular, we use triple *(x, r, y)* to represent connection *r* from resource *x* to resource y.
2. Identification Graph *IG*: a directed graph whose nodes are identifications and edges are relations between the identifications. In particular, we use *u(x)* to denote identification *u* of resource *x*.
3. Representation Graph *HG*: a directed graph whose nodes are hypertexts and edges are relations between the hypertexts. In particular, we use *h(u(x))* to denote hypertext *h* obtained from the identification *u(x)* and *l(r, u(x))* to denote a hyperlink *l* corresponding to the connection *r* to identification *u(x)*, and *m(h(u(x)))* to denote the media type of the hypertext *h*.

Because the selection operations are performed by a client and not controlled by a REST API, we only analyze interaction operations. In particular, we use $p(u(x))$ to denote protocol p to interact with identification $u(x)$. Within these notations, we consider three types of basic changes at each graph: (1) replacing element x by y is denoted by $x \rightarrow y$; (2) adding element x is denoted by $+x$; and (3) removing element x is denoted by $-x$.

3.1 Connection

The basic changes to a resource graph RG are summarized in Table 1.

Table 1. Impacts of the basic changes to RG.

(x,r, y)	Interaction	Identification	Representation
$\rightarrow(x, r, z)$	$+p(u(z))$	$+u(z)$	$l(r, u(y)) \rightarrow l(r, u(z))$ *in* $h(u(x))$
$+(x, r, z)$	$+p(u(z))$	$+u(z)$	$+l(r, u(z))$ *in* $h(u(x))$
$\rightarrow(x, s, y)$	*no*	*no*	$l(r, u(y)) \rightarrow l(s, u(y))$ *in* $h(u(x))$
$+(x, s, y)$	*no*	*no*	$+l(s, u(y))$ *in* $h(u(x))$
$-(x, r, y)$	$-p(u(y))$	$-u(y)$	$-l(r, u(y))$ *in* $h(u(x))$

When a REST API replaces resource y by z or adds resource z, the REST API must create new identification $u(z)$ in graph IG. The new identification leads to new interaction operation $p(u(z))$. However, if resource z already exists in the REST API, there will be no changes to the interaction operations or identification graph.

A REST API can replace the connection between the resources. Since the resources remain the same, no changes are necessary to the interaction operations and graph IG, except that the new connection must be advertised in the HG as hyperlinks.

When a REST API removes a connection including its resource, it needs to remove the corresponding interaction operation, identification, and hyperlink properly. However, if resource y is still connected in the REST API, then no change to graph IG is needed.

When a REST API changes connections between the same resources, it will break those clients that navigate those connections. To deal with such changes, a client can backtrack to rediscover the connections. For example, if the REST API changes connection $a \rightarrow b \rightarrow c$ to $a \rightarrow d \rightarrow c$, then the client cannot navigate to resource c from b. The client can backtrack to resource a and follow d to c instead. This technique can eventually find any resources connected to an entry resource.

3.2 Interaction

The basic changes to interaction operations are summarized in Table 2.

The first two rows show that a REST API can change the protocol to interact with identification $u(x)$, by associating different URI resolution procedures with the hypertext.

Table 2. Impact of basic changes to interaction operations.

$p(u(x))$	Connection	Identification	Representation
$\rightarrow q(u(x))$	*no*	*no*	$\rightarrow h(u(x))$
$+q(u(x))$	*no*	*no*	$\rightarrow h(u(x))$
$-p(u(x))$	*no*	*no*	$-l(r, u(x))$ *in* $h(u(x))$

The last row means a REST API can remove the protocol for a URI without removing the URI itself. A URI without a protocol identifies a resource that cannot be resolved from this hypertext.

When a REST API replaces a protocol, it will break those clients that depend on the protocol. Although a client can prepare some common protocol stacks, it will increase the footprint of the client, and there is still no guarantee that it will cover the protocols used by a new REST API. For the same reason, it is better to install a small set of common protocol stacks in a client, and add the new ones required by a REST API when needed.

3.3 Identification

The changes to *IG* and impacts are summarized in Table 3.

Table 3. Impact of changes to identification graph IG.

$u(x)$	Connection	Interaction	Representation
$\rightarrow v(x)$	*no*	*maybe*	$l(r, u(x)) \rightarrow l(r, v(x))$
$+v(x)$	*no*	*maybe*	$+l(r, v(x))$
$-u(x)$	*no*	*maybe*	$-l(r, u(x))$

A REST API can replace, add or remove identifications without modifying the underlying connection graph *RG*. When a new identification *v(x)* is introduced, a new interaction operation may be needed, and a new hypertext is always necessary. When identification *u(x)* is removed, the corresponding interaction operation becomes useless, if no other identifications use the operation.

Replacing identification *u(x)* may break those clients that do not know or understand the new identification. A REST API can use HTTP redirection to “teach” a client the new identification using the old identification. A REST API can also use a URI template [28] to prepare its clients for the range of URI changes. However, these techniques require that the new and old identifications use the same interaction operations. If new interaction operations are introduced, the client has to be modified.

3.4 Representation

The changes to HG and impacts are summarized in Table 4.

Table 4. Impact of changes to a representation.

m(h(u(x)))	Connection	Interaction	Identification
→*n(h(u(x)))*	*no*	*yes*	*no*
+*n(h(u(x)))*	*no*	*yes*	*no*

In these changes, we assume the new media type *n* (e.g. JSON) is equivalent to *m* (e.g. XML). Therefore, there is no change to the resource connections or identifications. However, the interaction protocol messages have to be changed to transmit the new media type.

Although a client can use content negotiation to accept a set of media types, there is no guarantee that these media types will overlap with those supported in a REST API. When a REST API replaces a media type, it breaks those clients that depend on that removed media type. Similarly, adding a new media type requires reprogramming the client in order to use it.

If a REST API modifies the specification of a media type, including its structure and processing rules, a client must be reprogrammed, because hypertext-driven navigation procedures depend on the media types.

4 REST Chart Model

To localize the impact of REST API changes on client, we decompose a client into two functional components: (1) client oracle that copes with changes in connection and identification; and (2) client agent that copes with changes in representation and interaction. In hypertext-driven navigation, a client oracle selects hyperlinks to visit from hypertext, while a client agent interacts with the selected hyperlinks to obtain the new hypertext. In order to define the compatibility between two REST APIs from the perspective of clients, we formalize these two concepts into a client model. Since any REST client model is dependent on the REST API it uses, we develop a formal model of REST API and use it to derive the client model. By defining compatibility based on the abstract models instead of concrete REST API implementations, we are able to derive a generic and efficient algorithm to check the compatibility without REST API implementations.

The REST API model we choose is REST Chart (RC) [14], which is originally proposed to design and describe REST APIs without violating the REST principles [1]. REST Chart models a REST API as a Colored Petri Net [23] that can model concurrent system with complex data structures as colors. The structure and behavior of REST Chart can be explained with a simple example in Fig. 2. This REST Chart with one transition, two input places, and one output place. It models a login REST API with one resource. The login place contains a token x_1 which is the entry point to the REST API. The credential place is initially empty. To navigate to the account place, a client must deposit a valid token x_2 in the credential place. The two tokens x_1 and x_2 cause the transition to fire, emulating the interaction with the resource. If the credential x_2 is

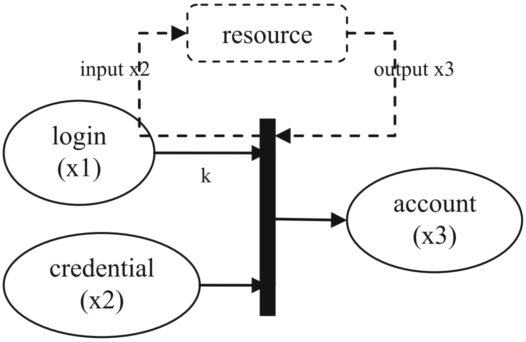


Fig. 2. Example of a basic REST Chart.

accepted by the resource, a token x_3 representing the account information will be deposited in the account place by the REST API.

The above interaction can be modeled by a sequence of token markings of the Petri Net, where each token is a valid resource representation. As this REST Chart has 3 places, each token marking is a 3 dimensional vector and the interaction involves 3 token markings:

$$(x_1, 0, 0) \rightarrow (x_1, x_2, 0) \rightarrow (0, 0, x_3).$$

Token x_1 may have many hyperlinks besides the login. To distinguish the login hyperlink *h* from the rest, REST Chart adopts Predicate/Transition Petri Net [22] and attaches a hyperlink predicate *k(h)* to the arc from the login place to the transition arc to the account. Hyperlink predicate *k(h)* qualifies a hyperlink *h* with two information items [25, 26]:

1. *[service]:* a URI [27] that represents the service provided by the hyperlink.
2. *[reference]:* a URI Template [28] that identifies the locations of the resource.

Two hyperlink predicates *k1* and *k2* are equal if *k1.[service] = k2.[service]* and *k1.[reference] = k2.[reference].*

Predicate *k(h)* is true if and only if the following conditions hold:

1. *k.[service] = h.[service]*;
2. *match(k.[reference], h.[reference]) is complete.*

Function *match(x, y)* returns a set of *v* = *s* pairs, for each variable *v* of URI template *x* that is instantiated by string *s* from URI string *y*. Function *match(x, y)* is complete if and only if all the variables of *x* are instantiated by *y*. The following XML represents a hyperlink predicate *k*, where *k.[service]* = *link/rel/@value*, and *k.[reference] = link/href/@value*:

```
<link id="k">
 <rel value="http://a.b.com/login" />
 <href value="http://{d}/users/{u}/account" />
</link>
```

The following XML represents an ordinary hyperlink *h*, where *h.[service] = link/@rel* and *h.[reference] = link/@href*:

```
<link id="h" rel="http://a.b.com/login"
href="http://a.b.com/users/john/account" />
```

Clearly *k(h)* is true because the two conditions hold:

(1) *k.[service] = h.[service] =* http://a.b.com/login;
(2) *match(k.[reference], h.[reference]) = {d =* http://a.b.com*, u = john}.*

With hyperlink predicate, we can represent the client state in Fig. 2 as a sequence of *p-k* pairs, where *p* denotes a place, *k* denotes the hyperlink predicate selected at *p*, and *0* means no hyperlink is selected:

$$login{-}k \; account{-}0$$

This *p-k* representation is equivalent to the token marking vectors, but it highlights the two main operations performed by client oracle and client agent: (1) the oracle selects a hyperlink at each place to move towards the goal place, in this case the account place; (2) the agent moves tokens, e.g. x_2 and x_3, between the places by interacting with the selected hyperlink.

To understand the distinction between these two components, we can regard a REST Chart as a maze where the transitions are the locked "doors" that protect the places. The oracle knows which door (hyperlink) to open at each place, but it does not have the keys (interactions) to unlock the doors. The agent has the keys, but does not know which doors to open. To move through a maze, a REST client needs the right kind of client oracle and client agent for that maze.

Within the REST Chart model, the changes in the layers of a REST API correspond to changes to a *p-k* path:

$$path = p_0{-}k_0\, p_1{-}k_1\, p_2{-}k_2\, p_3{-}0$$

Without losing generality, we assume that this path exists in version 1.0 of a REST API and the version 2.0 changes the path in the following ways.

Case 1: the new path is identical to the original *path*. Obviously, a version 1.0 client can reuse its client oracle and client agent to traverse the new path.

Case 2: the new path consists of the same pairs but different inter-pair relations. For example, new path p_0-k_0 p_2-k_2 p_3-*0* removes pair p_1-k_1, whereas new path p_0-k_0 p_2-k_2 p_1-k_1 $p_{3\text{-}0}$ reorders the original pairs. A version 1.0 client can keep its client oracle and client agent, because the client oracle can select the same hyperlink at the same place in version 2.0.

Case 3: the new path consists of different pairs combined from the same *p* and *k*. For example, new path p_0-k_2 p_1-k_0 p_3-*0* changes the hyperlinks selected at p_0 and p_1. For a version 1.0 client to traverse the new path, it needs a new client oracle that selects k_2, instead of k_0, at p_0. But the client can reuse its client agent, since all *p* and *k* in the new path occur in the original path.

Case 4: the new path consists of different pairs combined from different *p* and *k*. For example, new path p_0-k_3 p_4-k_4 p_3-*0* introduces new hyperlink predicates {k_3, k_4} and places {p_4} to reach the original goal place p_3. New hyperlink predicates mean new services and protocols that a version 1.0 client agent cannot fire, while new places mean new schemas and tokens that the client agent cannot process. For this reason, the version 1.0 client has to update both its client agent and client oracle. However, if the new places and transitions in version 2.0 are *covered* by some places and transitions in version 1.0, then the client can update its client oracle but keep its client agent.

5 RC Operational Model

In order to identify the compatible paths between REST Charts, this section introduces a deterministic operational model for REST Chart based on a state-based Petri Net behaviour model (Murata 1989). This model leads to a formal model of client oracle and client agent, from which the REST Chart compatibility checking algorithm is derived.

5.1 RC Behaviour Model

A REST Chart is a bipartite graph *RC = (P, T, F, M0, p0, L, S, K, type, link, bind)}*, where:

- *P* is the finite set of places.
- *T* is the finite set of transitions.
- $F \subseteq (P \times T) \cup (T \times P)$ is the set of arcs from places to transitions and from transitions to places.
- M_0: $P \rightarrow \{0, 1, 2, ...\}$ is the initial marking, a function that maps each place in *P* to 0 or more tokens.
- p_0 is the initial place.
- *L* is a finite set of media type definition language.
- *S* is the finite set of schemas in some type definition language in *L* and *valid(s, x)* indicates token *x* is an instance of schema *s*.
- *K* is the finite set of hyperlink predicates.
- *type:* $P \times L \rightarrow S$ maps each place and a media type language to a schema.
- *link:* $P \rightarrow 2^K$ maps a place to a set of hyperlink predicates.
- *bind:* $P \times K \rightarrow T$ binds a hyperlink predicate in a place to a transition.

We assume that RC has no isolated places or transitions as typical. For a REST Chat RC with *m* places and *n* transitions, let *A = [aij]* be the $n \times m$ incident matrix of integers, whose entry is given by:

$$a_{ij} = a_{ij}^{+} - a_{ij}^{-} \qquad (1)$$

$$a_{ij}^{+} = w(T_i, P_j), a_{ij}^{-} = w(P_j, T_i) \qquad (2)$$

where *w(T*$_i$*, P*$_j$*)* is the weight of the arc from transition T_i to its output place P_j and *w(P*$_j$*, T*$_i$*)* is the weight of the arc to transition T_i from its input place P_j.

For a given token marking *M*, let *M(P*$_j$*)* denote the number of tokens in place P_j. Transition T_i can fire if and only if:

$$a_{ij}^{-} \leq M(P_j), 1 \leq P_j \leq m \quad (3)$$

In the deterministic operation model, only one transition fires at each step. To select a transition to fire at the k-th step, we define a *n* × *1* column *control vector* u_k with exactly one 1 in the i-th position and 0 elsewhere to indicate that transition *i* fires. If *g* is the goal place, then the necessary condition to reach marking $M_d(g) > 0$ in *d* steps from M_0 is:

$$\Delta M_k = M_k - M_{k-1} = A^T u_k \quad (4)$$

$$M_d(g) = M_0 + A^T \sum_{k=1}^{d} u_k \quad (5)$$

Among the three factors of Eqs. (4) and (5), A^T is fixed by the REST Chart, u_k is controlled by the *client oracle* that selects a transition to fire, and ΔM_k is handled by the *client agent* that moves tokens between the places of the fired transition. The procedures of these two components are defined in Sect. 5.2.

5.2 REST Client Model

Our REST client model represents the hypertext-driven navigation shown in Fig. 1 in terms of Petri-Net. In particular, a *client agent A* of a REST Chart *RC* consists of the following abstract procedures that operate on tokens in a place:

- *(H, d) = decode(p, l, x)*: decode a token *x* in type language *l* in place *p* into a set of hyperlinks *H* and data *d*, such that:
  - $(\forall\ h \in H\ \exists\ k \in link(p))k(h)$: every decoded hyperlink *h* matches a hyperlink predicate *k* at place *p*.
  - *valid(type(p, l), x)*: if it is true, then it indicates that token *x* is an instance of schema *type(p, l)* at place *p* with language *l*.
- *x = encode(p, l, (H, d))*: encode a set of hyperlink *H* and data *d* into a token *x* in type language *l* in place *p*, such that:
  - *valid(type(p, l), x)* is true as described above.
- *(p, x*$_{out}$*) = fire(t, h, x*$_{in}$*)*: send token x_{in} to the resource identified by hyperlink *h* and receives token x_{out} in place *p* according to protocol defined by transition *t*.

In this model, each place and language pair defines a schema, and each token in a place is processed as an instance of the schema. To decode in place p_j, a token encoded in place p_i requires that these places maintain the *coverage* relation denoted by $p_i \subseteq p_j$. More precisely, for any media type definition language *l* and token *x*, $p_i \subseteq p_j$ if and only if *type(p*$_i$*, l)* $\subseteq$ *type(p*$_j$*, l)* such that:

$$valid(type(p_i, l), x) \rightarrow valid(type(p_j, l), x).$$

It is evident that if $p_i \subseteq p_j$, then any token encoded in p_i can be decoded in p_j such that:

$$(H, d) = decode(p_j, l, encode(p_i, l, (H, d))).$$

A *client oracle Q* of a REST Chart *RC* consists of the following abstract procedures that operate on the control vector:

- *$(k, t, p_j, H_j, d) = select(p_i)$:* select a hyperlink predicate *k*, transition *t* for *k*, input place p_j for *t*, data *d,* and hyperlinks H_j for p_j, based on the current place p_i.
- *Bool = goal(d):* return true if data *d* satisfies the target place.

A client oracle can be derived from a REST Chart to select a shortest path to reach the goal place. It could be implemented as a rule-based system or finite-state machine that is easy to reconfigure when *RC* or the goal changes.

The client operation model can be represented by a recursive procedure by which the client agent moves towards the goal place guided by the client oracle. More precisely, a REST client *C = (Q, A, reach)*, where *Q* is a client oracle, *A* is a client agent, and *reach* is a control procedure that combines *Q* and *A* to reach the goal place, starting from the initial place p_0 and token x_0 (Listing 1). Variable *V* collects the traversed places and transitions with an empty set as the initial value.

```
reach(l, Q, A, V, $p_0$, $x_0$)
    ($H_0$, $d_0$) = A.decode($p_0$, l, $x_0$)
    if Q.goal($d_0$) then return V
    (k, t, $p_1$, $H_1$, $d_1$) = Q.select($p_0$)
    if h $\in H_0 \wedge$ k(h) then
      V = V $\cup$ {($p_0$, $x_0$, t)}
     $x_1$ = A.encode($p_1$, l, $H_1$, $d_1$)
      ($p_2$, $x_2$) = A.fire(t, h, $x_1$)
      return reach(l, Q, A, V, $p_2$, $x_2$)
    end
    return V
end
```

Listing 1. REST client operational model.

6 REST Chart Compatibility

The compatibility between two REST Charts RC1 and RC2 can be defined in terms of the client operation model introduced in Sect. 5. More formally, a place *p* in REST Chart RC_2 is compatible with REST Chart RC_1 for client *C*, if and only if the following conditions hold:

1. *$C = (Q_{RC2}, A_{RC1}, reach)$;*
2. *$(p, x) = reach(Q_{RC2}, A_{RC1}, M_0, p_0, x_0)$*;

where:

- Q_{RC2} is a client oracle for RC_2;
- A_{RC1} is a client agent for RC_1;
- M_0 is the initial state of C;
- p_0 is the initial place of RC_2;
- x_0 is the initial token in p_0.

By the maze analogy in Sect. 4, this definition implies that RC_2 is compatible with RC_1 if client C can reuse the keys for RC_1 to open the doors in RC_2, when guided by oracle Q_{RC2}. Here a key refers to the *decode()*, *encode()* and *fire()* procedures defined in Sect. 4.2. The situation is illustrated in Fig. 3, where client C has a token in place p_{02} and its oracle Q_{RC2} selects door t_2 to enter place p_{22}.

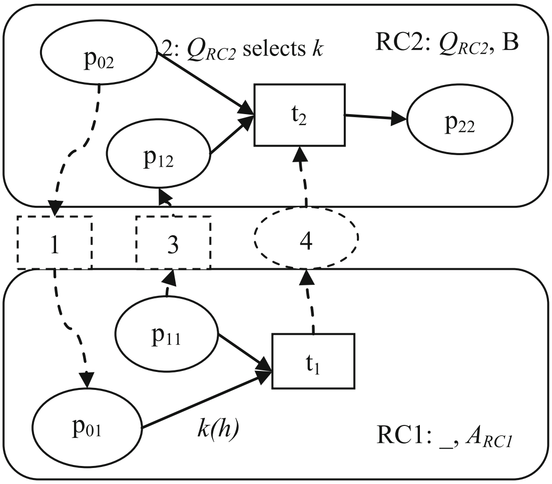


Fig. 3. Client C uses agent A of RC_1 to fire a transition of RC_2.

To find a reusable key, we introduce agent B to RC_2 whose job is to search RC_1 for a door (transition) t_1 equivalent to t_2 so that C can use the key for t_1 to open t_2. This equivalent relation can be defined with an auxiliary Petri Net that connects RC_1 and RC_2 with dashed places and transition 1, 3, 4 shown in Fig. 3. Transition t_1 and t_2 are equivalent if C can fire transition t_2 in the following 4 steps. At **step 1**, agent B encodes token x_0 in place p_{02} and sends it to place p_{01} where agent A_{RC1} decodes x_0 to extract the hyperlinks H_0. At **step 2**, oracle Q_{RC2} selects from p_{02} hyperlink predicate k that leads to place p_{22}. Client C applies k to H_0 to choose hyperlink h to follow at place p_{01}. At **step 3**, B finds place p_{11} for A_{RC1} to encode token x_1 (request). Agent A_{RC1} sends token x_1 to place p_{12} for B to decode it. At **step 4**, agent A_{RC1} fires transition t_1 which in turn fires transition t_2 to produce token x_2 (response) in p_{22}. The procedures of A_{RC1} and B at each step are correlated in Table 5 (for brevity, the subscripts of Q and A are omitted).

Table 5. Procedures called by client agents A and B.

	RC2: Q, B	RC1: Q, A
1	$x_0 = B.encode(p_{02},l,(H_0,d_0))$ $valid(type(p_{02}, l), x_0)$	$(H_0, d_0) = A.decode(p_{01}, l, x_0)$ $valid(type(p_{01}, l), x_0)$
2	$(k, t, p_1, H_1, d_1) = Q.select(p_{02})$ $\exists h \in H_0 \wedge k(h)$	
3	$(H_1, d_1) = B.decode(p_{12}, l, x_1)$ $valid(type(p_{12}, l), x_0)$	$x_1 = A.encode(p_{11},l,(H_1,d_1))$ $valid(type(p_{11}, l), x_0)$
4	$(p_{22}, x_2) = B.fire(t_2, h, x_1)$ $t_2 \subseteq t_1$	$(p_{22}, x_2) = A.fire(t_1, h, x_1)$ $t_1 \subseteq t_2$

Table 6. Constraints between RC_2 and RC_1.

1	$p_{02} \subseteq p_{01}$
2	$k \in RC_1.link(p_{01}) \cap RC_2.link(p_{02})$
3	$p_{11} \subseteq p_{12}$
4	$RC_1.bind(p_{01},k) = t_1 = t_2 = RC_2.bind(p_{02}, k)$

For client agent A_{RC1} to fire the transition, all the procedures in Table 5 must succeed in the right order. However, these conditions rule out non-validating client agents that do not use schemas. For this reason, Table 5 summarizes the necessary conditions for finding a compatible path. Using the hyperlink predicate equality relation defined in Sect. 4 and the schema coverage relation defined in Sect. 5.2, the Table 5 conditions can be reduced to Table 6, where the dependences on the operational procedures are removed and the conditions depend only on the structures of RC_1 and RC_2. These structural conditions allows us to compare RC_1 and RC_2 with a Depth-First search algorithm to traverse RC2 aided by RC1 as outlined in Listing 2.

```
traverse(RC2, RC1, V, $p_0$)
    if $p_0 \in$ V then return
    V = V $\cup \{p_0\}$
    K = RC2.link($p_0$)
    for each k $\in$ K
        $t_2$ = RC2.bind($p_0$, k)
        if $t_1$ = RC1.cover($t_2$) then
            V = V $\cup \{(t_2, t_1)\}$
            $p_2$ = RC2.output($t_2$)
            traverse(RC2, RC1, V, $p_2$)
        end
    end
end
```

Listing 2. REST Chart comparison algorithm.

Procedure *RC_1.cover()* finds a transition t_1 in RC_1 equivalent to transition t_2 in RC_2 based on the Table 6 conditions without actually constructing the auxiliary places and transitions in Fig. 3. For each transition in RC_2, this procedure may need to search up to $|P_1|$ places of RC_1 in the worst cases, where P_1 is the set of places of RC_1, since hyperlink predicate k may occur in all places of RC_1. As the algorithm traverses all transitions and places of RC_2, its time complexity is $O(|P_1|(|P_2| + |T_2|))$, where P_2 is the set of places and T_2 is the set of transitions of RC_2.

7 Prototype and Experiments

We implemented the REST Chart comparison algorithm (Listing 2) in Java and tested it on several REST Charts. The Java tool uses JDOM package to parse two REST Charts, each defined by some XML files, and outputs compatible places and schema relations. An example output of the REST Chart comparison is illustrated in Fig. 4, where a compatible place in the new version is marked by arrows pointing to the old places that cover it.

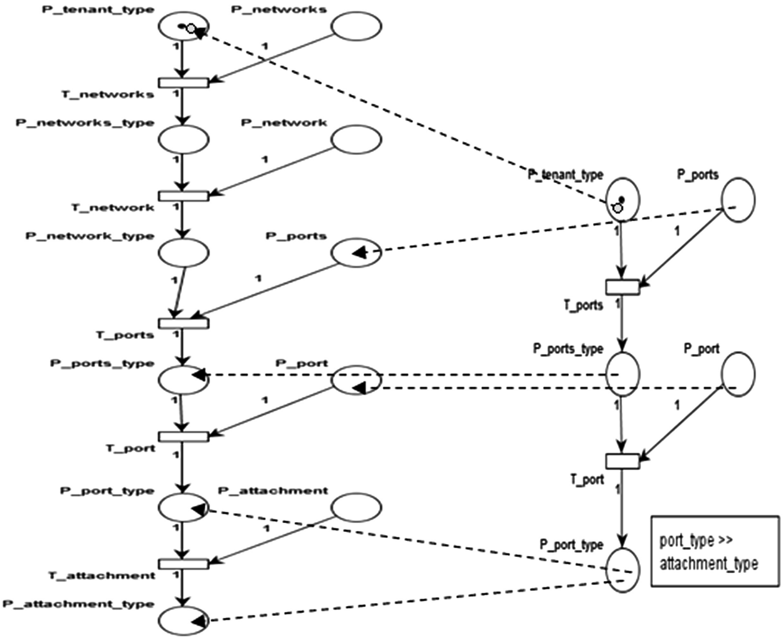


Fig. 4. Compatible places between version 1.0 (left) and version 2.0 of Floodlight REST API.

The *RC_1.cover()* procedure is based on the SOA membrane package (SOA Membrane XSD tool 2014) that compares XML schemas and identifies their differences. Given two XML schemas s_1 and s_2, the procedure *compare(s_1, s_2)* returns the differences between them as set D of pairs *(e, op)*, where e denotes an element in s_1 or s_2 and *op* denotes the operation that adds, removes or modifies the position or type of e:

$$D = \{(e, op) | op = \{add, remove, move, type\}\}.$$

Certain pairs in D create incompatible differences such that some XML documents validated by s_1 are not validated by s_2. Let B be the set of such incompatible differences, where *e.min* is the minimum occurrence of element e, and $t_1 \succ t_2$ indicates that type t_1 is a super type of type t_2:

$B = \{\mathrm{e} | (e, remove) \in D \vee (e, move) \in D \vee ((e, add) \in D \wedge e.min \neq 0) \vee (e, type, t_1, t_2) \in D \wedge (\mathrm{t}_1 \succ \mathrm{t}_2))\}.$

Let G be the set of compatible differences:

$$G = \{e | e \in s_1 \wedge e \notin B\}.$$

Then we have the following decision rules:

1. $D = \{\}$: $s_1 = s_2$;
2. $B = \{\}$, $G \neq \{\}$: $s_i \subseteq s_j$;
3. $B \neq \{\}$, $G \neq \{\}$: *s_1 is partially covered by s_2.*

Table 7. Performance summary.

Charts	RC1	RC2	Time (ms)
Place	17	19	1179.4
Transition	17	21	30.3
Place	8	10	1063.8
Transition	10	11	55.9

To test the correctness of the algorithm, we took a REST Chart and generated a dozen versions of it by changing the places, transitions, and schemas in various ways. Then the outputs of the algorithm against the changes were verified. The performance of the algorithm is summarized in Table 7 for two REST APIs: SDN REST Chart (rows 1 and 2) and flat Coffee REST Chart (rows 3 and 4). The results are averaged over 5 runs of the algorithm on a Windows 7 Professional notebook computer (Intel i5 CPU M560 Dual Core 2.67 GHz with 4 GB RAM). The results show that the algorithm spent extra (1179.4 − 1063.8) = 115.6 ms when the complexity factors increased by (19 + 21) * 17/((10 + 11) * 8) = 4 times from the Coffee REST API to the SDN REST API.

8 Conclusions

Compatibility checking of REST API is important, because extensibility is one of the main but underutilized advantages in RESTful service framework. Dealing with compatibility at large scale requires not only a well-designed REST API, but also automated methods to detect changes in a REST API. This paper defines compatibility between REST API from the perspective of its clients, and it develops an efficient algorithm to check the compatibility between two REST API models based on Coloured Petri Net. Our approach is independent of the REST API implementations, and it promotes REST client reusability by decomposing the client into the functional modules of client oracle and client agent. For future work, we plan to extend the REST Chart comparison algorithm to more complex cases and implement an automated client migration process based on the comparison.

Acknowledgements. We would like to thank Mr. Tony Tang for implementing the prototype system.

References

1. Fielding, R.T.: Architectural styles and the design of network-based software architectures. Dissertation, University of California, Irvine (2000)
2. GSMA OneAPI (2013). http://www.gsma.com/oneapi/voice-call-control-restful-api/. Accessed 20 August 2015
3. OpenStack REST API v2.0 references. http://developer.openstack.org/api-ref.html. Accessed 20 August 2015
4. Floodlight REST API. http://www.openflowhub.org/display/floodlightcontroller/Floodlight+REST+API. Accessed 20 August 2015
5. OpenStack API Complete Reference. http://developer.openstack.org/api-ref.html. Accessed 20 August 2015
6. OpenStack Releases. https://wiki.openstack.org/wiki/Releases. Accessed 20 August 2015
7. Hadley, M.: Web Application Description Language, W3C member Submission, 31 August 2009. http://www.w3.org/Submission/wadl/. Accessed 20 August 2015
8. RAML Version 0.8. http://raml.org/spec.html. Accessed 20 August 2015
9. Swagger 2.0. https://github.com/swagger-api/swagger-spec. Accessed 20 August 2015
10. Robie, J., et al.: RESTful Service Description Language (RSDL), Describing RESTful services without tight coupling. In: Balisage: The Markup Conference 2013 (2013). http://www.balisage.net/Proceedings/vol10/html/Robie01/BalisageVol10-Robie01.html. Accessed 20 August 2015
11. API Blueprint Format 1A revision 7. https://github.com/apiaryio/api-blueprint/blob/master/API%20Blueprint%20Specification.md. Accessed 20 August 2015
12. Gomadam, K., et al.: SA-REST: Semantic Annotation of Web Resources, W3C Member Submission, 05 April 2010. http://www.w3.org/Submission/SA-REST/. Accessed 20 August 2015
13. Alarcon, R., Wilde, E.: Linking data from RESTful services. In: LDOW 2010, Raleigh, North Carolina, 27 April 2010
14. Li, L., Chou, W.: Design and describe REST API without violating REST: a petri net based approach. In: ICWS 2011, Washington DC, USA, pp. 508–515, 4–9 July 2011
15. Robie, J.: RESTful API Description Language (RADL) (2010). https://github.com/restful-api-description-language/RADL, 2014. Accessed 20 August 2015
16. Champin. P-A.: RDF-REST: a unifying framework for web APIs and linked data. In: Services and Applications over Linked APIs and Data (SALAD), Workshop at ESWC, May 2013, Montpellier (FR), France, pp. 10–19 (2013)
17. SOA Membrane WSDL tool. http://www.membrane-soa.org/soa-model-doc/1.4/cmd-tool/wsdldiff-tool.htm. Accessed 20 August 2015

18. WSDL Auditor. http://wsdlauditor.sourceforge.net/. Accessed 20 August 2015
19. WSDL Comparator. http://www.service-repository.com/comparator/compareWSDL. Accessed 20 August 2015
20. Thompson, H.S., et al.: XML Schema Part 1: Structures Second Edition, W3C Recommendation, 28 October 2004. http://www.w3.org/TR/xmlschema-1/. Accessed 20 August 2015
21. SOA Membrane XSD tool. http://www.membrane-soa.org/soa-model-doc/1.4/cmd-tool/schemadiff-tool.htm. Accessed 20 August 2015
22. Murata, T.: Petri Nets: properties, analysis and applications (invited paper). Proc. IEEE **77**(4), 541–561 (1989)
23. Jensen, K., Kristensen, L.M.: Colored petri nets: a graphical language for formal modeling and validation of concurrent systems. Commun. ACM **58**(6), 61–70 (2015)
24. Cassandras, C.G., Lafortune, S.: Introduction to Discrete Event Systems, 2nd edn. Springer, New York (2008)
25. Li, L., Chou, W.: InfoParser: infoset driven XML processing for web services. In: ICWS 2008, Beijing, China, pp. 513–520, September 2008
26. Li, L., Chou, W.: Infoset for service abstraction and lightweight message processing. In: ICWS 2009, Los Angeles, pp. 703–710, July 2009
27. Berners-Lee, T., Fielding, R., Masinter, L.: Uniform Resource Identifier (URI): Generic Syntax, Request for Comments: 3986, January 2005. https://tools.ietf.org/html/rfc3986. Accessed 20 August 2015
28. Gregorio, J., et al.: URI Template, Request for Comments: 6570, March 2012. https://tools.ietf.org/html/rfc6570. Accessed 20 August 2015

Internet Technology

Augmented ODV: Web-Driven Annotation and Interactivity Enhancement of 360 Degree Video in Both 2D and 3D

Maarten Wijnants(✉), Kris Van Erum, Peter Quax, and Wim Lamotte

Hasselt University – tUL – iMinds, Expertise Centre for Digital Media,
Wetenschapspark 2, 3590 Diepenbeek, Belgium
maarten.wijnants@uhasselt.be

Abstract. Despite recent technological innovations in the media authoring ecosystem, one example being the ability to video record a scene with a 360 degree or so-called omni-directional field of view, video consumption to date largely remains a lean-back type of endeavor. This paper fuses the *Augmented Video Viewing* paradigm with 360 degree video technology to give rise to *augmented Omni-Directional Video* (ODV), a novel content format that holds promise to more deeply engage viewers and that unlocks more active ways of interacting with omni-directional video footage. The augmented ODV principle is exposed through a collection of standards-compliant Web interfaces to ease adoption by developers. At the same time, its Web-driven design maximizes the applicability of the technology, both in terms of consumption platforms and usage domains. Two use case prototypes showcase the artistic expressiveness of the augmented ODV methodology, while performance evaluation results establish the modest computational footprint of its implementation.

Keywords: Augmented Video Viewing (AVV) · Omni-Directional Video (ODV) · 360 degree video · Interactive video · Augmented video · Hypervideo · Web technology · WebGL · JavaScript

1 Introduction

Recent advancements in the capturing and authoring process, both in terms of hardware and software, have paved the way for technological innovations in the media landscape. One such innovation is *Omni-Directional Video* (ODV) or so-called 360 degree video. As its name implies, ODV content refers to video footage that is recorded with a 360 degree (i.e., cylindrical or spherical) Field of View. Typical ODV player implementations allow viewers to freely adapt their viewing angle inside the omni-directionally captured video scene.

Technical evolutions like the ODV concept unfortunately cannot conceal the fact that typical media consumption environments continue to deliver largely passive, static and non-interactive experiences, just like they did upon their

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 47–69, 2016.
DOI: 10.1007/978-3-319-30996-5_3

inception at the beginning of the 20th century. For sure, by affording the ability to spatially navigate through a 360 degree video scene in real-time and in an unconstrained fashion, ODV technology holds important promises in terms of granting viewers an enhanced feeling of belonging and immersion when compared to traditional video. Unfortunately, besides perspective personalization, no additional advanced interaction options are scaffolded by typical ODV installations. Interactivity has nonetheless proven to be a vital tool to attract the attention of video consumers and to maximize viewer retention (e.g., [1]).

To mitigate the laid-back characteristics of video as a medium, we have previously proposed the *Augmented Video Viewing* (AVV) paradigm and its associated Web-compliant codebase [2]. The AVV mindset aims to factor in interactivity and dynamism in the core fabric of the video consumption process. In effect, typical AVV experiences offer users interaction possibilities that go well beyond basic playback control. This is accomplished in a pragmatic fashion, by embellishing traditionally produced (passive) video material with interactive constructs that are exclusively realized using Web technologies. In particular, out-of-the-box HTML5, CSS and JavaScript features are leveraged to respectively define the structure and substance of the interactive assets, their styling, and their dynamic behavior (e.g., their spatio-temporal constraints). The interactive constructs are encoded as *hotspots* that are superimposed on top of the video playback in order to closely integrate the video content with its interaction provisions.

In this article, we give rise to the *augmented ODV* concept by mapping the AVV methodology to ODV data, this way effectively converting the latter from a passive into an interactive content format. All technical measures that had to be taken to reconcile the two constituting technologies will be described in detail and, in this process, our solution's far-reaching integration with contemporary Web standards will be emphasized. At the same time, the computationally lightweight nature of the augmented ODV paradigm will be underscored by presenting the outcome of an extensive performance benchmark. These prime achievements are accompanied by a total of three peripheral scientific contributions. First, through the presentation of two representative prototype realizations, we provide a hint of the advantageous traits of augmented ODV with regards to end-user envelopment, engagement in and participation with (omni-directional) video content. Secondly, the showcased prototypes offer a glimpse of the creative and artistic design options that are unlocked by the augmented ODV concept. In effect, augmented ODV is expected to pave the way for the delivery of a whole new breed of highly interactive, compelling and engaging video sensations in a myriad of application domains. As such, the augmented ODV approach holds important business opportunities for content producers by allowing them to cater to and capitalize on new consumer profiles. Finally, the third contribution of this article takes the form of a Web interface that offers video artists a graphical toolkit to facilitate the authoring and editing of rich AVV-based interactive (omni-directional) video experiences.

It is explicitly stressed here that the focus of this article is purely on the technological solutions that substantiate the augmented ODV paradigm and that jointly constitute a completely interactive and accessible system for the consumption and creation of augmented ODV experiences. As such, formal user experience validation is out of scope with regards to this publication. It is advocated to qualitatively assess user-oriented metrics on a case-by-case basis anyway, as they tend to depend largely on the content at hand and on the actual augmented ODV application scenario.

2 Related Work

The elementary notion of extending video content with interactive traits has already been studied and approached from a number of different perspectives. Meixner et al. have published an impressive and fairly up-to-date reference work pertaining to this subject [3]. In particular, they have devised an extensive taxonomy of solutions described in the academic literature by classifying them into four categories: *interactive video*, *annotated video*, *non-linear video* and *hypervideo*. Instead of needlessly recapitulating their work here, this section will deliberately concentrate on a critical selection of the most relevant scientific efforts only.

Given its Web-focus, the AVV paradigm is most akin to the *hypervideo* category of related systems identified in Meixner et al.'s taxonomy. The basic hypervideo objective consists of translating the hyperlink concept that we have become familiar with via the Web (mostly in textual form) to the video medium. As such, a hypervideo can be defined as a video document in which one or more user-clickable anchors are embedded. Two pioneering contributions in this research domain were delivered by the *HyperCafe* [4] and *HyperSoap* [5] experiments. Other notable academic hypervideo contributions include *Hyper-Hitchcock* and its *detail-on-demand* video concept [6], the *non-linear video* approach proposed by Fraunhofer Institute FOKUS [7], *HyLive* due to its emphasis on live broadcast scenarios [8], *SIVA Producer* [3], the *Component-based Hypervideo Model* (CHM) by Sadallah et al. [9], and the *360° hypervideo* solution proposed by Neng and Chambel [10].

Hypervideo-inspired systems have also been developed outside of the academic world, either in the form of freeware frameworks or commercialized offerings. Examples include cacophony.js (http://www.cacophonyjs.com), Mozilla's Popcorn.js HTML5 media framework (http://popcornjs.org), Gravidi (http://www.gravidi.com), and YouTube Video Annotations (http://www.youtube.com/t/annotations_about).

A somewhat different cluster of related work is that of Web-compliant multimedia presentation technologies. Systems in this solution category aim to deliver online interactive experiences involving a mixture of media types (i.e., they are not necessarily video-centric). A representative example of such a technology is *SMIL State* [11]. Unfortunately, native support for SMIL State is still largely lacking in contemporary Web browsers.

This article describes a pragmatic, Web-compliant approach to attach interactive features to ODV content. With the exception of the *360° hypervideo* solution by Neng and Chambel, none of the technologies cited above have explicitly considered ODV content. As a result, it is highly questionable whether they will be directly applicable to ODV contexts.

The proposed augmented ODV methodology and the *360° hypervideo* system are similar in the sense that they are both Web-driven. However, it is unclear how expressive the latter solution is in terms of overlay element specification. Our approach imposes no creative boundaries whatsoever in this regard. Also, the solution by Neng and Chambel appears to lack a visual authoring environment. Finally, the computational overhead imposed by the *360° hypervideo* system is unknown. In contrast, the performance analysis presented in Sect. 8 will establish that augmented ODV applications are readily consumable on commodity hardware, including tablet devices.

3 ODV Web Player

The prototypes that will be demonstrated later on in this paper have all been realized on top of an in-house developed ODV player for the Web. Although not the focus of this work, we will briefly touch on the player's functionality, design and implementation in this section to grant readers a comprehensive insight into our contributions. It is important to note however that the applicability of the AVV paradigm is by no means confined to this particular ODV player implementation; instead, the AVV codebase can readily be ingested in any Web-compliant ODV setup.

3.1 Functionality

The ODV player presents users a spatially restricted viewport into the omnidirectional video scene that is controllable via the Pan-Tilt-Zoom (PTZ) principle. This implies that viewers are granted directional control in two dimensions with regard to the positioning of the view window. At the same time, users can zoom in and out in order to narrow or widen the spatial spread of the viewport, respectively. On the horizontal axis, the player imposes no navigational restrictions. Stated differently, viewers can perform seamless 360 degree panning as they see fit. In the vertical direction on the other hand, the tilt movement is confined to 180 degrees in order to reduce the cognitive load on the viewer and to anticipate potential motion sickness. Intuitively, this means that users can freely look left and right, but cannot loop in their tilt movement (see Fig. 1).

3.2 Implementation

The ODV player is intended to be embedded in a HTML page. It is completely Web-compatible in the sense that it exclusively leverages standardized HTML5 technologies instead of resorting to the use of third-party plug-ins like Adobe Flash or Microsoft SilverLight. Input-wise, the player expects the ODV content to have undergone an equirectangular projection (see again Fig. 1).

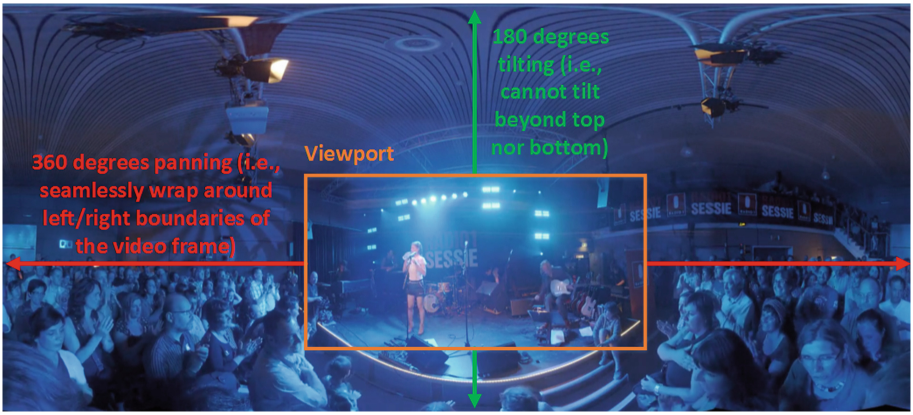
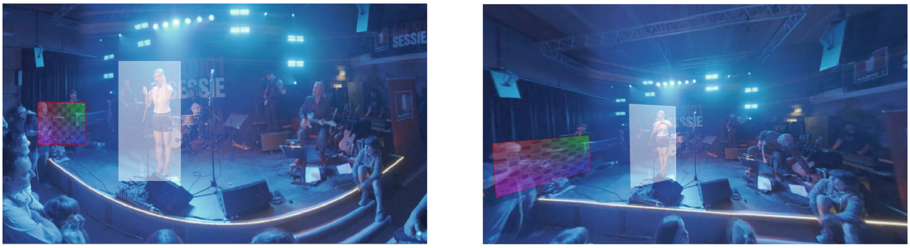

180 degrees tilting (i.e., cannot tilt beyond top nor bottom)
360 degrees panning (i.e., seamlessly wrap around left/right boundaries of the video frame)
Viewport

Fig. 1. Equirectangular projection of a single full frame of example ODV content, with an indication of the pan and tilt limitations imposed by the ODV player.



(a) 2D planar rendering

(b) 3D spherical projection

Fig. 2. The two instantiations of the employed ODV Web player. The 2D planar implementation introduces noticeable visual distortions which are induced by the equirectangular representation of the input ODV content.

Two alternative versions of the player exist that differ in the way ODV content is rendered and presented to the viewer. The first implementation adopts a two-dimensional (i.e., planar) rendering approach in which the currently active viewport is directly cropped from the equirectangular projection and subsequently rendered inside a standard HTML5 `<canvas>` element. The second version relies on WebGL (Web Graphics Library), a JavaScript API for plug-in-less rendering of hardware-accelerated 3D graphics inside Web browser instances [12]. Here, the ODV frames are textured onto the interior of a 3D sphere rendered through WebGL. Users control a virtual camera that is positioned inside this sphere in order to define their viewport into the ODV footage.

Both instantiations of the ODV player have their merits and shortcomings. First, as will be examined in Sect. 8, the planar visualization scheme generally has a lower computational expense compared to the 3D WebGL solution. Secondly, the 2D implementation only leverages basic HTML5 and JavaScript functionality, which maximizes portability. In comparison, its 3D counterpart requires

WebGL API support, which unfortunately is not yet universally available. As an example, while the majority of desktop Web browsers were pretty eager to adopt WebGL, one of the major mobile platforms (i.e., iOS) has only done so in the latest revision of its operating system (which was released on September 17th, 2014). Third, informally conducted qualitative experiments involving the ODV player have indicated that viewers generally appreciate the ability to zoom out the viewport to a global overview level (akin to the full frame visualization illustrated in Fig. 1). While the 2D instantiation can readily support this kind of behavior, the WebGL variant cannot (due to the fact that the movement of the virtual camera in this case is bounded by the interior of a sphere). Finally, concerning graphical fidelity, the WebGL-based ODV implementation outperforms its planar peer. This is evidenced in Fig. 2, which correlates the way the two player implementations visualize an ODV recording made during a small-scale concert that took place on a rectangular stage. The 2D planar visualization causes visual deformations to arise that are not present in the 3D spherical projection. This is best witnessed in the ill-shaped visualization of the podium in the 2D implementation.

4 Augmented Video Viewing

The basic premise of the AVV mindset consists of transforming video consumption from a purely passive, laid-back activity into a much more dynamic and (inter)active pursuit. This philosophy has bearing on a plethora of application contexts and use cases, including entertainment, online video-driven advertising and video-assisted remote tutoring. For example, in the video entertainment industry, the ultimate objective of content authors consists of producing thrilling and engaging drama that succeeds in not only captivating but also retaining the attention of the audience over time. However, due to the linear and non-interactive nature of classic video content, its initial appeal and viewer retention success largely depends on the skills of the different types of human operators who are involved in the content production pipeline (e.g., visual designers, camera operators, director). In case the presented content at some point fails to capture and engage the viewer, the risk arises that the viewer starts multitasking or even completely cancels the playback of the video. From the perspective of the content producer and distributor, this of course is highly unwanted behavior. Integrating interactive features in the video playback (e.g., in the form of gamification constructs [13]) holds promise to prevent the consumer from losing interest in and focus on the content.

As stated in the introduction, the AVV principle and its underlying implementation has previously been introduced [2]. Since then, the AVV codebase has been actively maintained and extended with additional functionality. In this section, the basic concepts of the AVV methodology will first concisely be recapitulated. Next, the newly developed AVV features will be described.

4.1 Interactive Video Overlays

The AVV framework offers a JavaScript API (the so-called JAVV API [2]) that facilitates the superimposing of interactive *video overlay elements* (VOEs) or so-called *hotspots* over a HTML5 video player. Overlay elements have spatio-temporal constraints that respectively define at which location and at what point during video playback they must be rendered. Spatial constraints are hereby not expressed as absolute screen coordinates but instead in terms of video coordinates (i.e., relative to the position of the video element in the surrounding HTML page). Furthermore, both static and animated positioning of video overlay elements is supported. Finally, overlay elements can have arbitrary programmatic logic associated with them. Typically, the execution of such logic is triggered by end-user interaction with the element or is timer-based.

Implementation-wise, each overlay element is represented by a `<div>` node in the HTML DOM. By assigning these dedicated nodes an elevated value for their CSS `z-index` property, it is guaranteed that they are always visibly overlaid on top of the video playback elements. HTML is exploited as markup language for the content that is embedded in overlay elements. This implies that hotspots can communicate a wide spectrum of information, as semantic information can be included through the application of appropriate HTML constructs. Analogously, the visual appearance and styling of overlay elements is controlled through the CSS standard. Finally, and again in line with the Web-focused design of the AVV framework, an overlay element's programmatic logic is expressed in JavaScript, with JavaScript's event-driven scripting model being exploited to couple dedicated interaction handlers to different types of viewer actions. In a desktop environment, for example, it will in many cases make sense to respond to `click`, `mouseover` and `mouseout` interactions that occur on the DOM representations of video overlay elements.

4.2 Motion-Tracked Video Overlays

The AVV framework comprises animation facilities for overlay elements. Looking at it from a software engineering perspective, the JAVV API codebase adopts the *strategy* design pattern to abstract overlay animation handling, this way introducing a certain level of flexibility in the animation subsystem. In particular, the software architecture defines an extensible family of interchangeable animation engines (each implementing a concrete animation style or technique), any of which can be attached at run-time to an overlay element in order to determine its spatial behavior over time. The initial version of the framework included only a single such animation engine, in particular one that implements a keyframing-like solution by performing linear interpolation between an overlay element's begin and end location over the course of the element's visible state [2]. Basically speaking, this animation engine causes overlay elements to follow a linear path.

Empirical insights obtained from developing concrete AVV-based test cases revealed that many scenarios would benefit from the ability to conceptually attach an overlay element to an in-scene object in the underlying video. In other

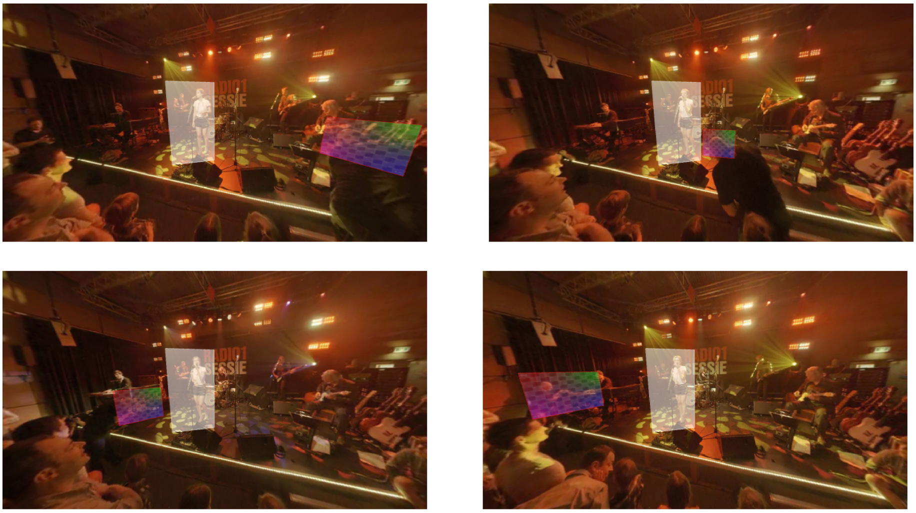


Fig. 3. Illustration of a *motion-tracked video overlay*. In this specific case, (the head of) an in-video actor was tracked so that an overlay element could be imposed on top of it.

words, we identified the need for an animation engine that enables overlay elements to follow the movement of subjects that appear in the video scene. To this end, we integrated a motion tracking animation engine in the JAVV API software architecture. This animation engine needs to be fed with cornerpin-based 2D tracking data as generated by off-the-shelf video editing software (e.g., Adobe After Effects or likewise) and subsequently applies the captured movement information to control both the position and spatial extent of the overlay element. As can be seen in Fig. 3, this animation engine is ideally suited to render overlay elements on top of (mobile) in-scene video items. Note that the visual accuracy of this animation engine largely depends on the performance and efficacy of the external algorithm that is responsible for implementing the motion tracking. Also note that the 2D motion tracking needs to be performed offline, as a pre-processing step.

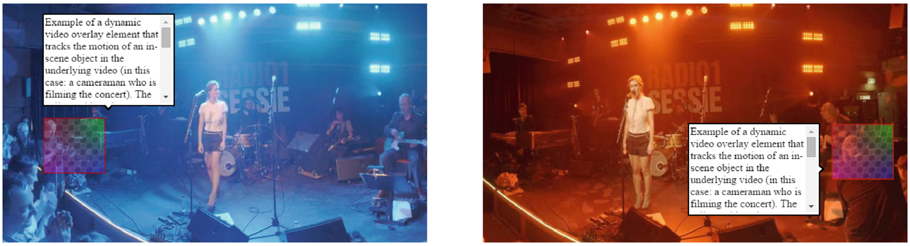
4.3 Reaction Video Overlays

Another desirable feature that emerged from practical experimentation with the JAVV API, was support for the specification of some form of parent/child relationship amongst individual overlay elements. Therefore, a specialized type of overlay element was implemented whose spatial positioning is defined relative to that of another hotspot (i.e., its parent). Since these types of overlays are typically visualized in response to some form of user interaction with the parent hotspot, they will in the remainder of this article be referred to as *reaction overlay elements*.

Three configuration settings control the spatial behavior of reaction overlay elements. The first configurable parameter is the absolute spacing that needs to be enforced between the reaction element and its parent. The reaction hotspot will always be offset the specified amount of pixels against the nearest edge of its parent. Secondly, experience designers can specify positioning preferences that define the order in which potential placement locations for the reaction overlay element are considered. These placement location alternatives are expressed relative to the parent item, at a high level of abstraction (i.e., to the left or right of the parent, above or below it, and so on). Reaction overlay elements are only rendered at a potential placement location in case sufficient screen real estate is available to host the element there. Since reaction items are never rendered partially, the following variables play a role in determining the eligibility of a placement location: the coordinates of the nearest border of the parent, the to-be-enforced spacing between the parent and the reaction element, the spatial dimensions of the reaction element and the spatial extent of the video player (or the view window in case of the ODV player, see Sect. 3.1). The third configuration setting is a Boolean flag indicating whether the reaction element must automatically be repositioned in case a placement location alternative that has precedence over the currently active one becomes available (e.g., due to changes to the viewport state in augmented ODV setups).

An advantageous trait of the reaction overlay element technology is that it interplays well with both animated parent items and viewport modifications in the augmented ODV player. In effect, reaction elements will follow the motion of their parent item and, in the course of the parent's animation, be dynamically repositioned as needed. This kind of behavior is illustrated in Fig. 4. Equivalent behavior is exhibited in the augmented ODV player when the user modifies his viewport into the ODV content.

One example of an interesting application area for the reaction overlay technology is the implementation of call-out widget-like functionality. As is demonstrated in Fig. 4, a reaction hotspot could present additional information about an in-scene object that is marked with a (potentially transparent) overlay element. In addition, by attaching some basic programmatic logic to the parent


Example of a dynamic video overlay element that tracks the motion of an in-scene object in the underlying video (in this case: a cameraman who is filming the concert). The

Example of a dynamic video overlay element that tracks the motion of an in-scene object in the underlying video (in this case: a cameraman who is filming the concert). The

Fig. 4. Parent object motion causes the relative location of its associated reaction element to be on-the-fly modified in accordance with the latter's positioning preferences (in this example, the preferred positioning order equals: above, right, below, left).

object, the visibility of the call-out box could dynamically be controlled on the basis of end-user interaction with the parent. As such, viewers can at run-time prevent the reaction overlay from cluttering the video playback at times when it is undesirable to do so. Please note that in this example, some simple HTML chrome was added to the appearance of the reaction element in order to give it an archetypical call-out look-and-feel.

5 Augmented ODV

This section will address the fundamental contribution of this article, namely the coupling of the JAVV API to ODV setups in general and to the ODV Web player that was introduced in Sect. 3 in particular. The incorporation in both instantiations of the ODV Web player will be discussed disjointly, due to the largely divergent technological requirements that were posed by each integration.

5.1 2D Planar Augmented ODV

Like any traditional video player, the 2D planar ODV renderer in essence presents two-dimensional images to the user. Due to this conceptual analogy, porting the AVV framework to this ODV setup turned out to be relatively straightforward. To be more precise, it mostly sufficed to correctly cover a number of "boundary cases" which all originate from the observation that, in contrast to a traditional video application, the 2D ODV implementation not necessarily displays complete video frames. In effect, depending on the ODV player's zoom level, the video footage might be cropped to fit the current viewport.

Conceptual sketches of each boundary case that had to be addressed are shown in Fig. 5. First of all, video overlay elements that integrally fall outside of the current viewport must be completely excluded from the rendering pipeline. Secondly, hotspots that are only partially contained in the viewport need to be clipped appropriately. This is achieved by modifying the value of the CSS `clip` property of the DOM representation of the involved overlay element. Finally, overlay elements with a large horizontal extent might need to be broken up into two discrete `<div>` representations for them to be rendered correctly.

Besides the boundary cases, the ODV player's support for zooming operations also required special attention. When zooming in or out, care has to be taken that overlay elements retain their logical positioning in the video scene. On the other hand, hotspots need to scale proportionally to the applied zoom level (in both the horizontal and vertical dimension). This latter requirement is fulfilled by applying the `scale` function of the CSS `transform` property to the concerned DOM element.

5.2 3D Spherical Augmented ODV

Whereas the JAVV API software architecture could quite readily be reconciled with the 2D ODV player implementation, this was not the case for its WebGL-based counterpart. In effect, in this latter instantiation, there is no longer a

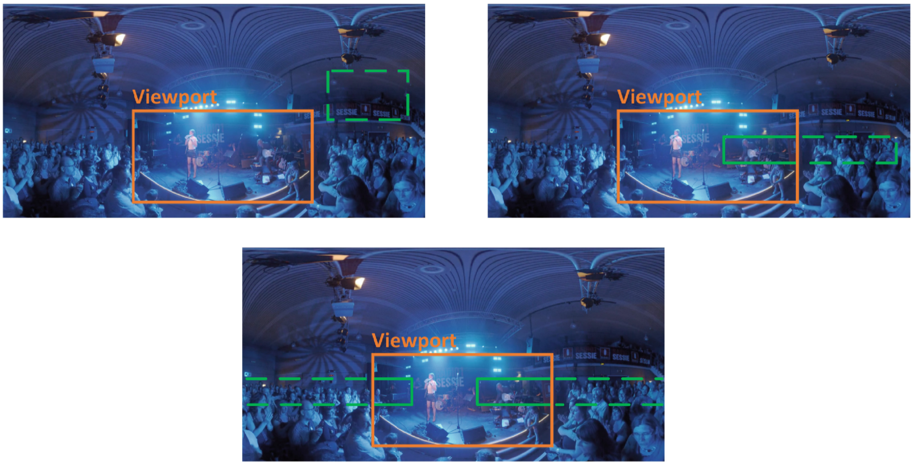


Fig. 5. Conceptual drawings of boundary conditions that had to be dealt with when porting the JAVV API to the 2D planar ODV setup. Dashed lines represent (portions of) hotspots that are not actually visualized.

notion of a planar presentation of video frames. Instead, the video content is applied to a curved surface (i.e., a sphere). If one would draw (flat) DOM elements as an exogenous layer on top of this three-dimensional scene, their graphical appearance would not necessarily blend in nicely with the ODV footage. Two alternative solutions were developed to remedy this problem.

CSS 3D Transformations. In the first approach, overlays remain to be represented by traditional DOM items, yet they are manipulated using CSS 3D transformations in order to correctly position them on top of the WebGL scene. In short, CSS defines a number of properties that allow DOM elements to be translated, rotated and scaled in a three-dimensional space [14].

We exploited the three.js JavaScript graphics library (http://threejs.org/), a convenient wrapper for WebGL, to mix the CSS 3D transformed DOM items with the WebGL scene. In particular, the three.js library allows for the addition of a CSS 3D renderer to an HTML page. This renderer is installed in such a manner that it spatially coincides with the WebGL renderer, yet in a superimposed manner.

Recall from Sect. 3.2 that the 3D ODV player applies the ODV footage to a curved surface. This causes the video content to be non-uniformly deformed (i.e., the deformation increases towards the edges of the viewport). The overlay elements must undergo an identical distortion for their visualization in the 3D space to be visually convincing. To this end, the two-dimensional video coordinates of each of the vertices that outline an overlay element, combined with the element's depth information (as specified by its CSS `z-index` value), are first expressed in a spherical coordinate system. The spherical coordinates (r, θ, ϕ)

are subsequently mapped to Cartesian positions (x, y, z) in 3D space [15] and are then fed to the CSS 3D renderer.

Representing Overlays as WebGL Objects. The second solution embraces the three-dimensional context not only for the visualization of the ODV material, but also for the rendering of the overlay elements. Like in the CSS 3D transformations scheme, the vertex outline of an overlay element is first spherically projected. Instead of providing the calculated 3D coordinates to the CSS 3D renderer, they are now directly interconnected in the 3D scene by means of WebGL-rendered lines. As a next step, the resulting shape is filled with a low-polygon 3D mesh. This approach enables the 3D representation of video overlay elements to display a background image (by attaching it as a texture to the mesh). Finally, the constructed mesh is equipped with a WebGL material so that, for example, a background color can be set for the overlay element.

The fact that video overlay elements are now rendered as 3D objects in a WebGL scene, as opposed to being represented as items in the DOM of the encapsulating webpage, has two important implications. First, it is no longer feasible to directly stylize hotspots via CSS. Therefore, CSS properties defining the visual appearance of overlay elements are parsed and (a subset of them) are translated to corresponding settings in WebGL. Examples of recognized CSS properties include border width, border color, background color and transparency level. A similar remark applies to the use of HTML to define hotspot content (e.g., an `<img>` tag is correctly translated to a WebGL texture, yet plain text in the HTML markup is simply discarded due to WebGL's poor text rendering support). Secondly, it invalidates the approach of handling end-user interaction with hotspots by listening for DOM events. As a workaround, a raycasting-based picking solution was adopted to determine which object(s) in the 3D scene the user is pointing at. In case the casted ray would intersect with multiple overlays, the one nearest to the virtual camera will be returned.

Comparison. The WebGL-integrated solution exhibits the detrimental characteristic that out-of-the-box support for the complete spectrum of existing CSS and HTML features is lost. Likewise, this implementation suffers from maintainability issues (if a new desirable CSS or HTML feature would be released, additional code would have to be written to explicitly support it). In contrast, the CSS 3D transformations-based scheme maximally retains compliance with the original AVV modus operandi. This implies that it inherits not just all AVV functionality that has previously been developed for non-ODV setups, but also all the off-the-shelf HTML and CSS constructs around which the AVV framework has been designed. For example, the layout and style of overlay elements remain controllable via CSS, the full HTML syntax remains exploitable to describe their contents, and interaction handling can still occur on the basis of DOM events. In terms of visual realism however, the WebGL-integrated approach will typically intertwine the overlay elements and the 3D scene in a visually somewhat more convincing fashion than the CSS 3D transformations-powered implementation.

This is due to the fact that the latter uses dedicated renderers for respectively the 3D scene and the overlay elements, which prohibits them from being tightly integrated. Finally, it is also important to note that the WebGL-integrated solution outperforms its CSS 3D transformations-based counterpart in terms of computational complexity (see Sect. 8 for concrete performance figures).

All reaction overlay elements that appear in a 3D ODV player screenshot in this paper were rendered using CSS 3D transformations (i.e., see Fig. 8). This is motivated by the fact that this class of hotspots typically relies extensively on CSS style features and HTML constructs (e.g., to implement the call-out chrome, see Sect. 4.3). Furthermore, they often carry text-based content. Based on the just presented comparison, it should be apparent that implementing such hotspots as WebGL objects would be cumbersome. On the other hand, the WebGL-integrated approach is the preferred solution for less complex overlay elements, due to its improved efficacy with regard to visual fidelity as well as its smaller computational footprint. Consequently, this technique was applied to visualize the non-reaction overlays in the 3D ODV player screenshots (i.e., see Figs. 2(b), 3 and 8).

6 Authoring

The targeted authoring audience of the AVV framework encompasses not only Web developers, but also video enthusiasts (both amateurs and professional practitioners). The technological expertise of these two user categories likely diverges largely. As an example, it is fairly safe to assume that members of the former have profound knowledge of prevailing Web practices. As a result, they might be sufficiently versed to define overlay elements directly in JavaScript. It is however highly improbable that the same is true for people with a background in visual design or video production. Such users would therefore benefit from the ability to construct AVV experiences from a more high-level point of view, using a graphical user interface. In this section, the alternative authoring solutions that were developed to cater to the desires and competences of both user bases will be presented. Of course, AVV experience designers are by no means confined to a single editing method and are free to fuse the different approaches as they see fit. For example, it could make perfect sense to draft a rough version of the envisioned AVV setup using the graphical editor, and then to refine the result by manually tweaking some settings directly in JavaScript.

6.1 Direct Video Overlay Instantiation

At the lowest end of the abstraction scale, authors can define video overlays directly in JavaScript, by instantiating their corresponding representation in the JAVV library. This approach basically boils down to the writing of JavaScript code, and will therefore probably only be viable for users that exhibit at least elementary Web development skills.

6.2 JSON Specification

The JAVV software architecture applies the *factory* design pattern to allow for the construction of overlay elements on the basis of JavaScript Object Notation (JSON) input. The code listing below provides an example of a JSON-encoded video overlay element representation. Since in this approach it suffices for authors to draft JSON documents instead of doing actual JavaScript coding, this solution is situated somewhat higher up the abstraction ladder than the previous one.

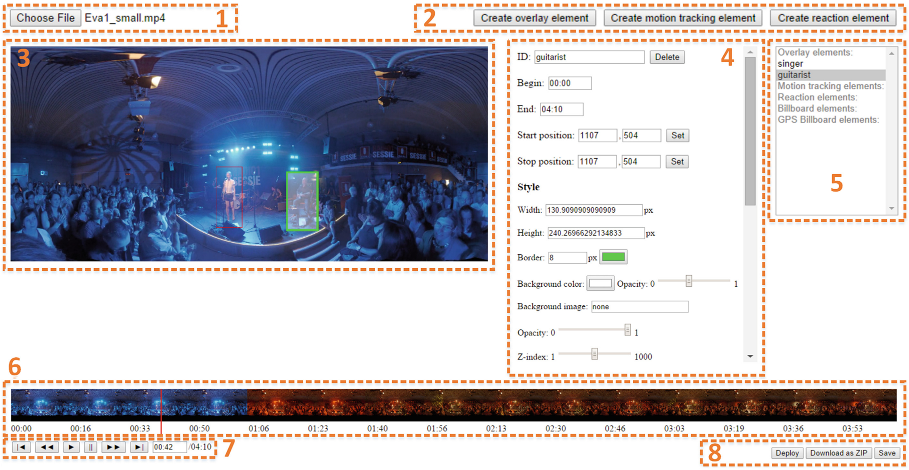
```
{
  "id": "MyId",           // Unique VOE identifier
  "timeStart": 100,       // Visibility start time
  "timeStop": 200,        // Visibility end time
  "positionStart": {      // 2D position at time "timeStart"
    "x": 10,
    "y": 10
  },
  "positionStop": {       // 2D position at time "timeStop"
    "x": 20,
    "y": 30
  },
  "htmlContent": /* Arbitrary HTML markup content goes here */,
  "interactionHandler": {
    "click": function(e, overlay) { ... },
    "mouseover": function(e, overlay) { ... },
    "mouseout": function(e, overlay) { ... }
  },
  "htmlStyle": /* CSS customization instructions go here */
}
```

Supporting indirect video overlay element instantiation via an open and language-independent standard for data interchange like JSON entails clear benefits in terms of flexibility, portability and interoperability. An additional advantage of embracing a widely accepted standard is that it is typically blessed with a wealth of facilitating tools. As an example, the JAVV library exploits JSON Schema [16] to validate the syntactical structure and semantics of JSON documents describing video overlay elements.

6.3 Graphical Editor

The most high-level AVV authoring solution is provided by a graphical editing interface that is accessible via a standard Web browser. An annotated screenshot of this Web service is shown in Fig. 6. The editor supports the augmentation of both ODV and traditional video content.

Without going into too much detail, the editor's design is centered around important HCI guidelines such as providing direct feedback (visual or otherwise) and supporting direct manipulation. The former design principle is, for instance, applied by ensuring that the actions performed by users are, whenever possible,



ID: guitarist

Begin: 00:00

End: 04:10

Start position: 1107 , 504

Stop position: 1107 , 504

Style

Width: 130.9090909090909 px

Height: 240.26966292134833 px

Border: 8 px

Background color: Opacity: 0 1

Background image: none

Opacity: 0 1

Z-index: 1 1000

Overlay elements:
singer
guitarist
Motion tracking elements:
Reaction elements:
Billboard elements:
GPS Billboard elements:

00:42 /04:10

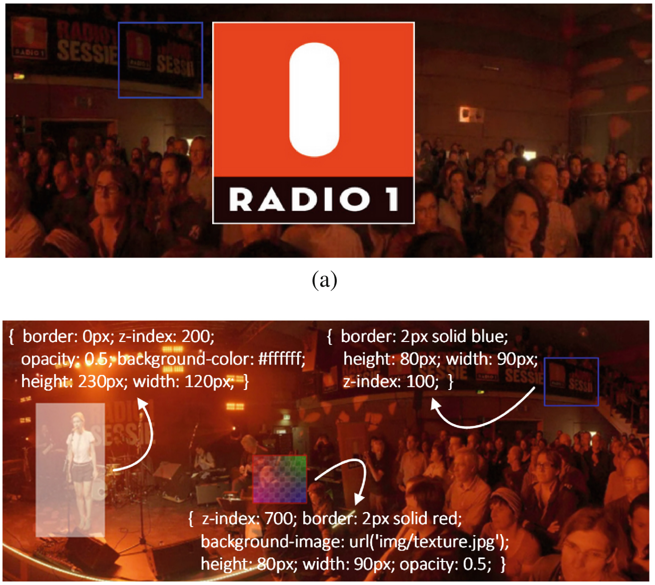
Fig. 6. The graphical interface of the editor Web service: (1) Video content selection; (2) Overlay element instantiation; (3) Preview widget (visualizes the currently active video frame and the overlay elements that apply to it); (4) Management form for overlay element properties; (5) Enumeration of defined overlay elements (i.e., selection pane); (6) Video playback timeline (including thumbnails of individual video frames to facilitate temporal navigation); (7) Video playback controls; (8) Export and deployment options.

directly reflected in the preview widget. Another illustration of the adoption of this principle is given by the WYSIWYG text editor that is included in the Web service (not shown in Fig. 6). This tool allows users to specify the textual contents of overlay elements without mandating them to be familiar with the HTML syntax. On the other hand, an example of direct manipulation support can be found in the fact that users are able to apply drag-and-drop interaction to intuitively reposition already defined overlay elements in the preview window.

The result of the editing process can be exported to an intermediary format that in turn can be applied by the JAVV API in order to incorporate the authored AVV experience inside a HTML page. The Web service even includes the option to publish the editing outcome (i.e., the combination of the involved video clip and the overlay elements that were defined for it) in a directly consumable manner to an HTTP server.

7 Showcases

In this section, we will present two augmented ODV prototypes. The first prototype chiefly acts as a technological proof-of-concept and is hence intended to showcase the feasibility as well as the functional features of the proposed technology. The second demonstrator on the other hand involves a real-life use case and consequently provides a hint of the valorization potential of our work.



(a)

```
{ border: 0px; z-index: 200;
  opacity: 0.5; background-color: #ffffff;
  height: 230px; width: 120px; }
```

```
{ border: 2px solid blue;
  height: 80px; width: 90px;
  z-index: 100; }
```

```
{ z-index: 700; border: 2px solid red;
  background-image: url('img/texture.jpg');
  height: 80px; width: 90px; opacity: 0.5; }
```

(b)

Fig. 7. Additional screenshots of the augmented concert showcase: (a) Reaction video overlay carrying image content; (b) Overview of the three non-reaction elements, each annotated with its respective CSS style string.

7.1 Augmented Concert Capture

The first demonstrator is built around ODV content that was captured during a small-scale musical performance involving an audience of approximately 150 people. A stationary ODV camera was installed in front of the stage, amidst the audience, to record the concert. The screenshots that have been included in the paper up to this point all originate from this demonstrator.

The demonstrator itself is implemented as two distinct HTML pages which respectively host the 2D planar and the WebGL-based ODV player. Via a HTML link embedded in the pages, one can switch between the two ODV player instantiations. The video footage is augmented with a total of five overlay elements. Two of these are statically positioned, one is animated on the basis of motion tracking data, and the final two are reaction overlays. The statically positioned overlay elements mark the singer of the band and a segment of a promotional banner, respectively. Both are rectangularly shaped and their styling respectively consists of a semi-transparent fill color and colored edges (see Fig. 7(b)). Interacting with the singer's hotspot causes a short biography of the band to be displayed in a dedicated portion of the webpage, external to the ODV player. On the other hand, selecting the overlay element on top of the banner results in a reaction element popping up which holds an image of the radio station that organized the concert. This effect is shown in Fig. 7(a). The motion-tracked overlay element follows a camera operator as he moves around in front of the stage. Figure 4 illustrates that this particular element is visualized as a semi-transparent image with a border colored in red and that interacting with it causes a reaction video overlay acting as a call-out box holding textual information to appear on top of the video footage.

Please remark that the styling solutions and content presentation methods that are showcased in this prototype are mere illustrations and are in fact rather simplistic. For the sake of comprehensiveness, the exact style settings that apply to the three non-reaction hotspots are communicated in Fig. 7(b). It is explicitly repeated here that, courtesy of the extensiveness of both the CSS and HTML specifications, the artistic options for overlay element design available to visual artists are nearly limitless.

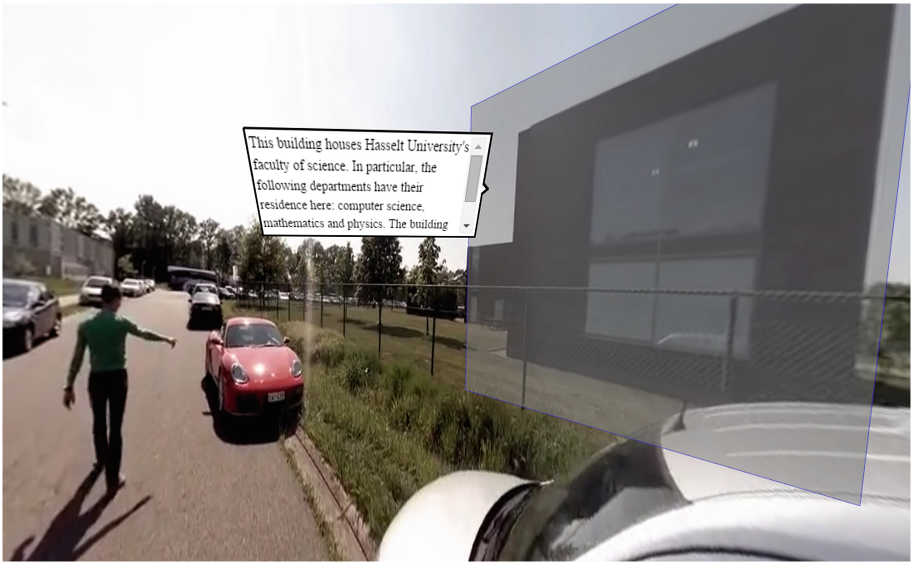
7.2 Virtual Walkthrough

The second demonstrator exerts the JAVV API to incorporate interactive features in a practical scenario, namely a virtual walkthrough use case.

Use Case. At Hasselt University, a Web application has been developed that offers students and employees a virtual walkthrough of the campus grounds in order to familiarize them with the spatial layout of the site. The application's media content consists of ODV recordings of a human guide as he walked around the campus territory. Along the way, the guide pointed out various salient features in his vicinity (e.g., university buildings or services). In a post-production phase, the ODV footage was temporally segmented on the basis of physical junction points (e.g., a crossroad) that were encountered during ODV capture.

The prototype can in a sense be regarded as an ODV-powered counterpart of the Street View functionality included in Google Maps. In effect, the typical usage scenario involves users virtually moving along the roadways and choosing their desired traveling direction at subsequent intersections in order to get a feel of the layout of the university grounds. While navigating the streets, users can play/pause the video sequences, freely change their viewing angle, and zoom in and out at their personal discretion. Please note that no specific effort was needed to realize this functionality, as the prototype directly inherits these functions from the incorporated ODV Web player. Also note that the Web application is meant for internal use only and is hence not publicly available.

AVV Integration. The AVV framework was integrated into this Web application so that relevant physical elements in the captured scenery could be tagged by means of interactive overlays. Given the mobility of the capture camera, all hotspots in this prototype are of the motion-tracked type. When selected, the hotspots display textual descriptions (encoded as reaction overlays) of the underlying object in the video footage. The AVV integration into this use case hence served a dual purpose: complementarily highlight the real-world items pointed



This building houses Hasselt University's faculty of science. In particular, the following departments have their residence here: computer science, mathematics and physics. The building

Fig. 8. Screenshot of the augmented version of the virtual walkthrough prototype (visualized using the 3D ODV player).

out by the human guide in the ODV footage, and appropriately present informative call-outs to users. A screenshot that illustrates the outcome of the AVV integration can be found in Fig. 8.

8 Performance Evaluation

To analyze the computational complexity of the augmented ODV solution, a performance benchmark was conducted. In particular, the augmented concert showcase presented in Sect. 7.1 was profiled with regard to computational resource usage on two distinct platforms: a mid-range desktop PC equipped with GPU hardware acceleration, and a low-cost laptop. The exact hardware specifications of the desktop PC were as follows: Intel Xeon W3505 CPU running at 2.53 GHz, Nvidia GeForce GTX 760 GPU, 4 GB RAM. The laptop was a DELL Latitude E6510 housing an Intel Core i3 M370 CPU clocked at 2.40 GHz, an Nvidia NVS 3100M graphics card, and 4 GB RAM. Software-wise, the desktop and laptop test platforms ran Windows 8.1 and Windows 7 SP1, respectively. On both machines, the audit was executed using Google Chrome version 39.0.2171.95 m (32-bit).

The benchmark test case was encoded as a script and subsequently applied a number of times under variable configuration settings and conditions. The script first made sure that the viewport of the ODV Web player was positioned in such a way that the band and the rectangular stage were in view. From that point on, no viewport modifications whatsoever occurred during the remainder of the test scenario. Once the viewport was appropriately positioned, the benchmark actually commenced. Approximately 1 second after the start, the overlay element that is associated with the singer of the band was selected (i.e., clicked on). After a time interval of about 1 second, the camera operator's motion-tracked video overlay was then selected. This action caused the corresponding reaction overlay element to be rendered in the viewport (see Fig. 4). Finally, again approximately 1 second later, the test case was concluded. Each benchmark run consequently consumed about 3 seconds in total.

The configuration settings that were varied across tests include (i) the resolution of the input ODV material, (ii) the output resolution of the ODV Web player (i.e., the viewport resolution), and (iii) using the 2D planar versus the 3D spherical augmented ODV Web player implementation.

The performance results were collected using the Chrome Developer Tools [17] and are summarized in Tables 1 and 2. In both tables, the figures communicate CPU consumption (i.e., method execution time), expressed as a percentage of the total running time of the involved benchmark test. As an example, a value of 5 would indicate that 5 percent of the benchmark's running time was spent processing the corresponding JavaScript function. The percentual time

Table 1. Benchmark results (% of total time) on a hardware accelerated desktop; 1920x1080 versus 960x540 video input, static viewport.

	2D Planar Augmented ODV				3D Spherical Augmented ODV				
Output	`draw`	`anim`	`clip`	**Idle**	`render`	`anim`	`uCSS`	`uWGL`	**Idle**
640x360	11.3	8.7	3.0	72.8	11.1	4.3	1.6	0.1	71.2
	1.6	8.5	2.9	82.1	2.8	4.3	1.8	0.1	79.1
1280x720	10.1	8.6	2.9	72.9	11.1	4.2	1.6	0.1	70.2
	1.6	8.2	2.9	83.0	2.8	4.7	1.8	0.1	78.4
1920x1080	9.7	8.1	2.7	72.2	10.2	4.4	1.7	0.2	70.9
	1.3	8.0	2.8	81.8	2.7	4.6	1.9	0.1	78.6

Table 2. Benchmark results (% of total time) on a commodity laptop; 1920x1080 versus 960x540 video input, static viewport.

	2D Planar Augmented ODV				3D Spherical Augmented ODV				
Output	`draw`	`anim`	`clip`	**Idle**	`render`	`anim`	`uCSS`	`uWGL`	**Idle**
640x360	10.1	12.4	4.0	65.9	11.8	6.5	2.7	0.1	62.1
	2.4	11.2	3.6	74.5	4.5	5.9	2.4	0.1	70.7
1280x720	9.5	12.4	3.9	63.7	12.1	6.8	2.7	0.1	60.2
	2.3	10.9	3.6	73.3	4.4	6.2	2.4	0.1	69.1
1920x1080	9.8	12.7	4.1	65.2	12.6	6.7	2.7	0.1	58.3
	2.3	11.8	3.8	72.3	5.2	6.4	2.5	0.1	67.9

the CPU was idle during the different audit configurations is likewise included in the tables. Each reported figure corresponds with the average of the results from 5 independent runs of the respective test, to moderate the impact of outliers. The semantics of the method names appearing in the benchmark results are as follows:

`draw` (abbreviation for `drawImage`) The default HTML5 method to draw to a `<canvas>` element; is used to render the currently active ODV viewport to the screen in the 2D planar ODV player

`anim` The JAVV API function that implements the positioning, animation and rendering of overlay elements; is separately implemented for both the 2D planar and 3D spherical augmented ODV alternatives

`clip` A subroutine of the 2D planar `anim` method that is exclusively responsible for overlay element clipping (see Sect. 5.1)

`render` The three.js function to render a WebGL scene

`uCSS` A subroutine of the 3D spherical `anim` method that manages the animation of overlay elements using CSS 3D transformations (see Sect. 5.2)

`uWGL` A subroutine of the 3D spherical `anim` method that manages the WebGL-integrated animation of overlay elements (see Sect. 5.2)

Space limitations unfortunately refrain us from delving exhaustively into the benchmark outcomes. Therefore, we will limit ourselves to enumerating four of the more salient findings. The first observation is that the 3D spherical augmented ODV renderer is more efficient at drawing and animating the overlay elements compared to its 2D planar counterpart (see the values of the respective `anim` columns in Tables 1 and 2). This finding holds true across all tested input and output resolutions, on both test platforms. Second, by comparing the `uCSS` and `uWGL` columns, it becomes apparent that updating and rendering video overlay elements using the CSS 3D transformations scheme is computationally considerably more expensive than via the WebGL-integrated approach. Recall from Sect. 5.2 that the former solution is exploited to display reaction hotspots. Exactly one such element became visible in the course of the benchmark test scenario. If the test case would not include a reaction hotspot, the difference in overlay element rendering time between the 2D and 3D implementations (as identified in the first finding) would be even larger. Thirdly, in the 2D planar augmented ODV implementation, the clipping of overlay elements consumes a considerable slice of the CPU budget (i.e., approximately one third of the animation processing is devoted to clipping). This is caused by the relatively large number of position and offset calculations that the clipping operation requires. The performance of this animation phase could potentially be improved by resorting to a clipping scheme that is based on the CSS `overflow` property. An `overflow`-based solution would transform the clipping from a manual to an automated process, which is expected to have a positive influence on performance. Investigating the validity of this hypothesis is an important subject of future work. The final finding pertains to the impact of the input and output video resolutions. In hardware accelerated settings, the input video resolution does not appear to

Table 3. Benchmark idle times (% of total time) on a Nexus 7 tablet; 1920x1080 versus 960x540 video input, static viewport.

	2D Planar Augmented ODV			3D Spherical Augmented ODV		
Output:	**640x360**	**1280x720**	**1920x1080**	**640x360**	**1280x720**	**1920x1080**
Idle time:	35.22 47.25	7.71 9.82	3.54 4.52	28.92 23.51	2.02 4.67	0.1 0.9

play an appreciable role with respect to overlay element rendering and animation performance, for either the 2D or the 3D Web player implementation. In the absence of decent GPU acceleration, a small impact on performance is however noticeable (the computational overhead of animating hotspots rises as the input video resolution increases, see the `anim` columns in Table 2). The tested ODV viewport resolutions on the other hand seem to only marginally affect the CPU workload induced by the JAVV library.

It is important to mention that, under every considered test condition, ample idle CPU cycles were available (as is evidenced by the "Idle" columns in Tables 1 and 2), which resulted in all experiments running smoothly at comfortable frame rates. This observation unfortunately only partly holds true on tablet devices. As an example, Table 3 elaborates the idle times on a Nexus 7 tablet (2012 edition) running Android 5.0 and Google Chrome 38.0.2125.509 (identical benchmark configuration permutations and test case as before, idle time expressed as a percentage of the total running time of the benchmark test, results were averaged out over 5 independent test runs). As can be derived from this table, the benchmark test conditions that yield the highest visual quality failed to render smoothly. On the other hand, the tested mid-range and low-end quality settings produced acceptable frame rates (i.e., above 25 FPS). As an example, when fed with 960x540 video input and producing 1280x720 video output, the 2D planar implementation witnessed 9.82 % idle time, which resulted in an average frame rate of approximately 25 FPS. Please note that, since the augmented ODV codebase is currently not optimized for mobile platforms, the figures reported in Table 3 definitely leave room for improvement.

In all, the presented audit findings lead us to conclude that the augmented ODV solution is computationally compatible with commodity hardware, although deployment on mobile devices might mandate quality sacrifices depending on terminal capabilities.

9 Conclusions

Over the years, the video medium has witnessed substantial technological innovations. This paper has focused on one such fairly recent innovation, namely the ability to record scenes with a spherical Field of View. Somewhat surprisingly, the medium has not seen the same degree of evolution when it comes to the types of experiences it is able to deliver to viewers. In particular, video consumption to date largely remains a passive matter, with little thought for end-user interactivity. This paper has proposed a pragmatic solution to battle this stagnation in the form of the augmented Omni-Directional Video concept. The rationale behind the concept consists of overlaying ODV footage with interactive elements that are represented and rendered in a Web standards-compliant manner, in this way effectively "activating" ODV consumption. The various building blocks required to realize the augmented ODV methodology have all been addressed.

Interactive elements to be integrated into the ODV content can be presented through divergent mechanisms (each with specific advantages and drawbacks) depending on imposed technical restrictions and the context of use. Nevertheless, all proposed mechanisms build upon well-established Web standards such as HTML5, CSS, JavaScript and WebGL. This deliberate design decision benefits the authoring process, as it allows content producers to engineer novel experiences without them facing the steep learning curve that is typically associated with the adoption of a new technology. To further improve the content creation workflow, a graphical authoring interface has been proposed that enables point-and-click interaction for tasks that would otherwise require repetitive (manual) editing.

The practical applicability and valorization potential of the augmented ODV concept has been showcased by two representative use case descriptions, which were implemented using a mixture of technologies presented in this paper. Finally, performance benchmark figures have revealed our implementation to be computationally economical.

Acknowledgements. The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 610370, ICoSOLE ("Immersive Coverage of Spatially Outspread Live Events", http://www.icosole.eu). The authors would like to explicitly thank Jori Liesenborgs for developing the ODV Web player, Jeroen Leën for porting the AVV codebase to the 2D planar ODV implementation, and prof. Philippe Bekaert for the ODV recording of the concert which served as video material in the technological demonstrator described in Sect. 7.1.

References

1. Raney, A.A., Arpan, L.M., Pashupati, K., Brill, D.A.: At the movies, on the Web: An investigation of the effects of entertaining and interactive Web content on site and brand evaluations. J. Interact. Mark. **17**, 38–53 (2003)
2. Wijnants, M., Leën, J., Quax, P., Lamotte, W.: Augmented video viewing: transforming video consumption into an active experience. In: Proceedings of the 5th Multimedia Systems Conference, MMSys 2014, Singapore, Singapore, pp. 164–167. ACM (2014)
3. Meixner, B., Matusik, K., Grill, C., Kosch, H.: Towards an easy to use authoring tool for interactive non-linear video. Multimedia Tools Appl. **70**, 1251–1276 (2014)

4. Sawhney, N., Balcom, D., Smith, I.: Authoring and navigating video in space and time. IEEE MultiMedia **4**, 30–39 (1997)
5. Dakss, J., Agamanolis, S., Chalom, E., Michael Bove Jr., V.: Hyperlinked Video. In: Proceedings of SPIE Multimedia Systems and Applications, vol. 3528, pp. 2–10 (1999)
6. Shipman, F., Girgensohn, A., Wilcox, L.: Combining spatial and navigational structure in the hyper-hitchcock hypervideo editor. In: Proceedings of the 14th Conference on Hypertext and Hypermedia, Hypertext 2003, Nottingham, UK, pp. 124–125. ACM (2003)
7. Seeliger, R., Räck, C., Arbanowski, S.: Non-linear video - A cross-platform interactive video experience. In: Proceedings of the 2nd International Conference on Creative Content Technologies, CONTENT 2010, Lisbon, Portugal, pp. 34–38 (2010)
8. Hoffmann, P., Kochems, T., Herczeg, M.: HyLive: Hypervideo-Authoring for Live Television. In: Tscheligi, M., Obrist, M., Lugmayr, A. (eds.) EuroITV 2008. LNCS, vol. 5066, pp. 51–60. Springer, Heidelberg (2008)
9. Sadallah, M., Aubert, O., Prié, Y.: CHM: an annotation- and component-based hypervideo model for the web. Multimedia Tools Appl. **70**, 1–35 (2012)
10. Neng, L.A.R., Chambel, T.: Get around 360° hypervideo. In: Proceedings of the 14th International Academic MindTrek Conference: Envisioning Future Media Environments, MindTrek 2010, Tampere, Finland, pp. 119–122. ACM (2010)
11. Jansen, J., Bulterman, D.C.: SMIL State: an architecture and implementation for adaptive time-based web applications. Multimedia Tools Appl. **43**, 203–224 (2009)
12. Khronos Group: WebGL Specification Version 1.0.2 (2013). https://www.khronos.org/registry/webgl/specs/1.0/
13. Hamari, J., Koivisto, J., Sarsa, H.: Does gamification work? - A literature review of empirical studies on gamification. In: Proceedings of the 47th Hawaii International Conference on System Sciences, HICSS-47, Hawaii, USA, pp. 3025–3034 (2014)
14. W3C: CSS Transforms Module Level 1 (2014). http://dev.w3.org/csswg/css-transforms/
15. Weisstein, E.W.: Spherical Coordinates. From MathWorld - A Wolfram Web Resource (2014). http://mathworld.wolfram.com/SphericalCoordinates.html. Online
16. Galiegue, F., Zyp, K., Court, G.: JSON Schema: interactive and non interactive validation. Internet-Draft, Internet Engineering Task Force (2013). http://tools.ietf.org/html/draft-fge-json-schema-validation-00
17. Google: Chrome DevTools Overview (2015). https://developer.chrome.com/devtools

CrySIL: Bringing Crypto to the Modern User

Florian Reimair(✉), Peter Teufl, and Thomas Zefferer

Institute of Applied Information Processing and Communications,
Graz - University of Technology, Inffeldgasse 16a, 8010 Graz, Austria
{florian.reimair,peter.teufl,thomas.zefferer}@iaik.tugraz.at

Abstract. Modern times introduced a highly heterogeneous device landscape. The landscape was populated by distributed applications. These applications are used by modern multi-device users. A modern user wants to create, process, and share potentially sensitive data among her devices. For instance, start a document at the smart phone, continue on the laptop and finish the document on a tablet. A common way to protect sensitive data against disclosure and theft is cryptography. Cryptography, however, requires for all devices in question to be able to perform appropriate operations and protect the subsequent cryptographic primitives against attacks. Unfortunately, different devices have different capabilities when it comes to cryptography. Some have hardware-backed solutions available, some cannot do any cryptography at all. In general, it is hard to provide adequate (and potentially equal) cryptographic methods on every device of the modern landscape – be it rather basic and well-known schemes or new methodologies that are long awaited to stand the challenges of the cloud. In order to tackle the above mentioned status and bring cryptography to the modern multi-device user, we present CrySIL, the Cryptographic Service Interoperability Layer. CrySIL is designed as a flexible and extensible layer between the user and the cryptographic primitive. In a nutshell, CrySIL can use local key storage solutions, offers remote key storage and crypto provider deployments, and features strong authentication methodologies to constrain access to cryptographic primitives. In this work, we explain the motivation of CrySIL, describe its architecture, highlight its deployment in a typical modern use case, and reflect on achievements and shortcomings.

Keywords: Cloud security · Central cryptographic solutions · Advanced cryptographic protocols · Heterogeneous applications · Mobile devices

1 Introduction

In recent years, a highly heterogeneous device landscape has become the standard scenario for the development and deployment of applications. Especially mobile devices introduced an unprecedented level of architectural heterogeneity. Modern

This work is an extended version of our initial conference publication [1].

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 70–90, 2016.
DOI: 10.1007/978-3-319-30996-5_4

web browsers support rich HTML5-based applications, mobile devices came to stay, new wearable computing systems claim their place in the world and classic desktop operating systems still stand their ground.

Applications present the user with use cases that became available only recently. Developing applications for today's and tomorrow's diverse environments boosts complexity of the development processes significantly. Especially security-related aspects – the deployment of cryptographic algorithms and protocols to offer confidentiality, integrity and authenticity for the processed data – suffer from this complexity in several ways: First, cryptographic algorithm and protocol implementations are not available on every device. Second, the capabilities of different devices to handle the required key data in a secure way strongly deviates. Third, the high level of complexity causes implementation mistakes that lead to significant security issues [2,3].

While existing and well-established solutions, such as smart cards and cryptographic tokens, offer a way to stand the challenge of key storage, distribution, and handling, they are only available on limited devices and cause additional cost and usability issues when being integrated in distributed applications deployed in the modern heterogeneous device landscape. Platform-based systems, i.e. TPMs [4], mobile TPMs, or proprietary solutions offer reasonable security, given that the security features are activated and configured accordingly. These system seldom share common configuration and features and therefore have to be configured and used one by one. The challenge is to get it right, every time. The challenge becomes even greater when multiple devices with different hardware or software-based encryption systems are to protect sensitive data. These examples represent only a small excerpt of the entire problem space and are augmented by the fact that the availability of cryptographic APIs is strongly device-dependent. E.g. when considering the web browser within an application scenario, one has to keep in mind that hardly any cryptographic API nor a secure key storage component is available.

While it might seem reasonable to hold sensitive cryptographic primitives directly at the user's device and refrain from storing these data on a central server, a severe drawback emerges: If stored locally, cryptographic primitives is not available everywhere and at any time. For instance, access to e-government services is infeasible in case the solicited smart card-based user authentication is impossible because the end-user device does not provide appropriate card-reading capabilities. In times, in which mobile and wearable devices are of growing importance, this is becoming a significant issue.

The industry created central cryptographic service to meet the anywhere and at-any-time requirements. We reviewed some solutions and found that their limited scope make them only suitable for specific use cases and deployment scenarios. For example, the Austrian Mobile Phone Signature solution is limited to the creation of electronic signatures, but does not feature cipher operations. The CloudHSM solution provided by Amazon is limited in terms of possible deployment scenarios. In particular, this solution is tied to a Public Cloud solution and to one specific Cloud provider. Similar limitations apply to comparable server-based solutions that offer storage of cryptographic primitive and central provision of cryptographic services. Furthermore, these solutions do not use or include locally available key data and cryptographic functions, which still might play an important role in many use cases.

To overcome limitations of existing solutions and to further improve the availability of cryptographic services everywhere and at any time, we propose the Cryptographic Service Interoperability Layer (CrySIL). CrySIL provides a flexible architecture to use cryptographic protocols and algorithms in a heterogeneous device environment and provides secure key storage and key handling capabilities. It features build-in authentication as well as transparent off-device key storage. When used as central service, CrySIL can also be seen as a cryptographic turntable for the rapid deployment of new cryptographic schemes, which are not yet supported by hardware and/or software solutions. This feature is especially important for upcoming advanced cryptographic schemes that will play an important role in cloud computing security.

The remainder of this paper is organized as follows. In Sect. 2, the motivation for CrySIL is outlined by discussing the current deployment of cryptographic algorithms and functions in heterogeneous application environments. Related work in Sect. 3 is followed by a detailed description of the CrySIL architecture in Sect. 4. Subsequently, in Sect. 5 the functionality and security challenges of CrySIL are explained by investigating the prototype of browser application for platform-independent file-encryption. Section 6 states recent on the CrySIL topic. Finally, the work is concluded by giving an outlook on future CrySIL plans and deployment scenarios.

2 Deploying Cryptographic Functionality in Modern Applications

In this section we give a detailed discussion on the challenges we face in deploying cryptographic functionality in modern applications.

2.1 Applications

The following deployment scenarios and application categories highlight the manifold use of cryptographic algorithms and protocols and the associated problems of deploying such technologies in heterogeneous application environments.

Heterogeneous Applications. In recent years mobile platforms have provoked a rush to applications that allow users to access and process their data from a wide range of devices. Nowadays a heterogeneous application comes in versions for desktop-based systems, different mobile device platforms and web browsers. In addition, specific environments, such as the Chrome Apps platform[1], further specific mobile environments (e.g., tablets, phablets, smart phones) and

[1] https://developer.chrome.com/apps/app_architecture.

vendor-specific aspects need to be covered. In recent years the first mass market wearable devices have been released and are expected to gain significant usage numbers in the near future[2]. While this heterogeneous device landscape offers significant new opportunities for customers, it comes with a major price tag – complexity. Developing for so many platforms requires different user experience design (UXD) strategies, different platform-specific programming-language and architecture skills and – especially important for the scope of this work – in depth knowledge how to provide confidentiality, integrity and authenticity for the handled data. Protecting the application data is especially relevant in the context of exchanging this data via cloud infrastructures to provide instant access on heterogeneous application platforms. Noteworthy examples for such applications are note-taking applications, messengers, applications for collaborative document processing, or cloud storage providers. Obviously, the secure handling of this data needs to be considered in the local environment as well as when the data is transferred and stored at the cloud provider. To offer the required level of protection, cryptographic algorithms and protocols are deployed.

Applications with Core Security Functionality. In a similar way, applications that advertise security features as core functionality require the deployment of low-level and high-level cryptographic algorithms and protocols. Examples for this application category are secure messengers, encrypted cloud storage solutions or password managers. On a high-level view those applications face the same challenges as the heterogeneous applications described in the previous section. However, the security requirements for those applications are typically more complex due to the higher likelihood of targeted attacks when the applications are deployed in security relevant deployment scenarios.

Enterprise-Level Applications. Security and especially the deployment of cryptographic functions are core aspects of enterprise-level applications. In terms of communication VPN solutions based on IPSEC or TLS are widely deployed. On a higher level, solutions such as S/MIME or PGP are required to ensure the confidentiality, authenticity and integrity of emails. While these communication systems are based on well established protocols their deployment also faces similar problems as other applications.

The deployment of cryptographic algorithms and protocols are also a core component of many other enterprise-level applications. Especially, the application of digital signatures to create authentic documents is one of these core functions. Legal frameworks (e.g., [5]) have established the basic fundament for the legally binding use of digital signatures. The applications of advanced digital signatures within private and corporate perspectives are manifold: signing legally binding documents, such as contracts, automatically signing documents, such as invoices, or providing authenticity and integrity for stored documents in general.

[2] http://www.businesswire.com/news/home/20140410005050/en/Worldwide-Wearable-Computing-Market-Gains-Momentum-Shipments.

The security requirements on the issued certificates, the deployed signature creation devices and the creation of the digital signature depend on the required signature level and need to be considered when deploying such applications.

Data encryption systems play an important role in securing data-at-rest and provide confidentiality when transmitting data to communication partners.

Future Applications. The wide deployment of cloud computing in recent years has brought up many issues in relation to security, privacy and data protection laws. Especially in Europe the usage of U.S. cloud computing resources is linked to significant legal problems [6]. One technical approach to solve these problems is the deployment of data encryption. However, by using current cryptographic schemes data needs to be encrypted directly at the client before it is stored on the cloud service provider. While this is a feasible method for cloud-storage systems, it completely removes the capability to process data in the cloud. However, processing data and thus outsourcing computational resources is one of the main advantages of cloud computing. Thus, current research focuses on new cryptographic protocols and algorithms capable of processing encrypted data in the cloud. The required functionality is brought – among others – by new cryptographic protocols and algorithms in the areas of homomorphic encryption [7], searchable encryption [8], verifiable encryption [9], proxy re-encryption schemes [10] or redactable signature schemes (e.g., [11]). There is a high variety in the applicability of these new protocols in production environments. While in certain cases the schemes are not yet suited for practical applications (e.g. homomorphic encryption) other schemes, such as proxy-re-encryption have already reached a prototypical stage (e.g., the NICS CRYPT Library[3]). However, even protocols from the latter category cannot yet be efficiently deployed in existing applications due to the lack of compatibility with current high level standards and the lack of hardware and software support on heterogeneous platforms.

2.2 Cryptographic Algorithms and Protocols

In heterogeneous application scenarios, many cryptographic algorithms and protocols need to be deployed depending on the specific use cases and security requirements. Thereby, the core problems are the secure provisioning, storage and sharing of key material, as well as the lack of heterogeneous platform support for the algorithms and protocols or storing and handling the key material in a secure way.

Core Cryptographic Algorithms. The core cryptographic functionality for basic and high-level cryptographic usage is based on the application of symmetric and asymmetric encryption algorithms, hash algorithms, and message authentication codes. In terms of the secure deployment in heterogeneous applications, especially the handling of asymmetric key material needs to be considered.

[3] https://www.nics.uma.es/dnunez/nics-crypto.

For data encryption systems asymmetric algorithms play an important role in hybrid encryption schemes. In such schemes, symmetric algorithms (e.g., AES) are used for bulk data encryption. The symmetric encryption keys are then either directly protected by asymmetric encryption algorithms (RSA-OAEP [12], RSA-PKCS15 [13]) or by using symmetric key material derived from asymmetric keys (e.g. via ECDH [14]). This is especially important when considering the secure handling and storage of the used key material. In general, the symmetric key material is only used in memory during the data encryption/decryption process and never stored in plain on permanent storage. However, this is not the case for the asymmetric key material, which is typically long-lived and needs to be stored in a secure storage area. When accessing data from heterogeneous platforms, this material either needs to be shared between different platforms or key agreement protocols need to be deployed. Similar considerations need to be taken when asymmetric key material is used for digital signature schemes.

Although, there is certain platform support for storing and using this key material (iOS KeyChain [15], Android KeyChain[4]) one needs to consider that the protection level of these platform functions heavily depends on the correct configuration and availability of protection mechanisms. Examples for such mechanisms are platform encryption systems and (often) associated access protection systems (PINs, passcodes, fingerprint sensors). Furthermore, this platform support does not solve the problem of securely sharing or provisioning key material. The latter functionality is especially important for use cases where advanced digital signatures need to be created. Another problem is the lack of platform support either related to missing key storage sub-systems or the lack of APIs that can be used to apply the required cryptographic algorithms. A common way to address this problem within heterogeneous applications is the deployment of password based key derivation functions described in the subsequent section.

Main Problems: secure key storage, key sharing, key provisioning, lack of platform support.

Password-Based Key Derivation Functions. Password-based key derivation functions are widely used in data encryption systems that are deployed in mobile applications. Typically, those systems are used in encryption schemes, where the actual data encryption keys are protected with a key that is the result of a key derivation function (KDF). There are two main reasons for deploying those system. First, in heterogeneous applications they provide the simplest way to implement key sharing without using dedicated key storage tokens (smart cards, tokens etc.). By using the same KDF parameters, password and salt value the same key is derived on each platform. This key is then used to gain the actual data encryption keys. Doing so does not require any dedicated hardware support. APIs that implement KDFs are widely available and can easily be deployed. Second, especially on mobile platforms password based KDFs are often applied in Bring-Your-Own-Device or consumer application scenarios. Thereby,

[4] http://developer.android.com/reference/android/security/KeyChain.html.

private devices are used that are not managed via central Mobile Device Management systems (MDM). In those scenarios, security functions such as encryption systems, or the required entry of PIN/passcodes that protect those encryption systems cannot be mandated and thus might not be available. To be able to provide data encryption functionality in such environments, applications deploy password based KDFs which can be used regardless of the state of platform security functions.

Two notable key derivation functions are the Password-Based-Key-Derivation-Function2 (PBKDF2) [16] and the SCRYPT function [17]. While the PBKDF2 is widely spread in applications as well as platform encryption systems (iOS [15,18], Android[5]) its protection against brute-force attacks is only based on key derivation time. The key derivation time is determined by the iteration count and the CPU where the KDF is executed. This problem is addressed by the SCRYPT function. SCRYPT, allows – in addition to the key derivation time – the developer to specify the desired memory resources required for the key derivation process. SCRYPT has been used in recent Android encryption systems but is not widely used in applications mainly due to the lack of API support.

Although, KDFs are based on passcodes and thus simply to use, they face two critical problems: First, the security of the derived key strongly depends on the chosen passcode and its properties. Weak passcodes allow the efficient application of brute-force attacks and thus the retrieval of the key material used for the data encryption system. Especially, on mobile platforms the use of long and complex passwords is not feasible mainly due to usability reasons. Second, the parameters of a key derivation function (e.g., salt, iterations, memory consumption) need to be chosen according to the specific deployment scenario and a good balance between security and usability needs to be found. When considering the requirement to find this balance, the influence of the user by choosing weak or strong passcodes, and the possibility to make implementation mistakes (e.g., choosing wrong parameters or static salt values [2]) it turns out that the achievable security level is subject to strong fluctuations.

While the main issues of deploying KDFs are related to weak passcodes and wrong implementations, the underlying problem is caused by the need to use them in the first place – the lack of secure and heterogeneous key storage, sharing and provisioning facilities.

Main Problems: secure key storage, key sharing, key provisioning.

High-Level Cryptographic Standards. In many deployment scenarios the application of cryptographic primitives for data encryption and digital signatures is not sufficient. E.g., for data encryption, hybrid encryption schemes are

[5] Android versions from 3.0 to 4.2 use the PBKDF2 function within the file-system encryption system. Starting with Android 4.3, the SCRYPT function is utilized. The current Android 5.0 version deploys the SCRYPT function as well, but in addition makes use of a hardware based element to strongly reduce the effectiveness of brute-force attacks.

often required that combine asymmetric and symmetric cryptography. Also, data structures for carrying the encrypted or signed data as well as providing storage for content encryption keys or signature data are required. Furthermore the algorithm identifiers for the used algorithms and modes need to be defined and stored in the data structure. Finally, identifiers for X.509 certificates need to be considered. To provide this functionality, high level standards have been defined and are employed in a wide range of applications. Wide spread standards are: The Cryptographic Message Syntax (CMS) [19] which is used for data encryption and digital signatures. CMS is also the basis for the S/MIME standard [20] that enables signed and/or encrypted emails. An important standard for the creation of digital signatures is the XMLDSIG standard [21], which allows to sign complete XML documents or parts of these documents. XMLENC is the counterpart for data encryption.

The main problem for deploying such high-level standards in heterogeneous applications is the lack of platform support. E.g., there are currently no cryptographic JavaScript APIs for web applications that could be used in productive environments, and mobile platform support depends on the specific vendor. The lack of support in web application is also based on the fact that there is currently no way to handle the secure storage of key material, which also extends to low-level cryptographic algorithms. The W3C Web Cryptography API[6] aims to solve this problem by providing standardized crypto APIs for web applications and secure key storage facilities.

Main Problems: lack of platform support, secure key storage, key sharing, key provisioning.

Advanced Cryptographic Protocols. In this context, we define the term "advanced cryptographic protocol" as any cryptographic protocol or method that is currently under research, or that has already been evaluated but is not or not widely available in productive environments. Examples for such protocols were given in Sect. 2.1.

While many of these schemes are not ready for practical applications, some of them have already reached prototypical stages. However, even for the latter category the main problem is the complete lack of platform support and compatibility with existing standards. This applies to hardware as well as software-based solutions. A good example is proxy-re-encryption where a prototypical library is already available. Still, deployment scenarios that rely on such schemes (e.g., [22]) cannot exceed prototypical stages without production-ready libraries.

Main Problems: lack of platform support.

3 Related Work

A number of server-based cryptographic services have been implemented by the industry. This section gives an overview and functional evaluation of selected cryptographic services that can be integrated into cloud-based environments.

[6] http://www.w3.org/TR/WebCryptoAPI/.

SigningHub[7] offers the creation of advanced digital signatures with unique cryptographic keys for different users. As a cloud-based service, it provides centrally-stored keys as well as signature creation using keys stored on smart cards or soft tokens. Signed documents are stored and managed on servers provided by *SigningHub*. The service can be easily integrated into applications and web services using a simple REST-based interface.

Dictao[8] and *Cryptomathic* [9] support digital signatures for transaction security and user authentication. Both services facilitate key access by authenticating clients using simple credentials, such as username/password schemes, or credentials that feature higher strength (eID cards, OTPs, mobile devices etc.). While *SigningHub* and *Cryptomathic* are deployed as cloud services, *Dictao* requires integration into an enterprise IT infrastructure. Basically, the provided functionality is limited to user authentication and signature creation.

The Austrian citizen card [23] is the official implementation of the eID concept of Austria. It is a technology-neutral concept for unique citizen identification and secure qualified signature creation. Aside from the smart card implementation [24], a cloud-based service [25] is operational. The so-called *Austrian Mobile Phone Signature*[10] is operated in a private cloud and uses a hardware security module (HSM) to store the private signature keys of all Austrian citizens. Access to these keys is protected by a strong two-factor authentication mechanism, involving a password as well as an OTP being sent to the citizen's mobile phone. Applications can access the Austrian Mobile Phone Signature and its signature creation functionality through a well-defined XML-based interface. Although the Austrian Mobile Phone Signature meets the demands of the law, the currently deployed implementation fails to support use cases other than signature creation and user identification.

With *AWS CloudHSM*[11], Amazon feeds the demand of integrating secure cryptographic operations into deployed applications without requiring an HSM available on premise. To meet regulatory requirements for data security, customers are able to acquire sole access to appliances on a dedicated HSM and therefore retain full control of the keys and the cryptographic operations of the HSM. The offered functionality of the HSM can be integrated into applications that are deployed within the Amazon Virtual Private Cloud (Amazon VPC) via the provided Java or C programming API. As *AWS CloudHSM* can only be used in conjunction with Amazon VPC, customers are bound to Amazon and a migration to other cloud providers is infeasible.

4 Cryptographic Service Interoperability Layer

Motivated by the unsolved challenges and already available solutions, we have created the Crypto Service Interoperability Layer (CrySIL) concept. CrySIL

[7] http://www.signinghub.com.

[8] https://www.dictao.com.

[9] http://www.cryptomathic.com.

[10] https://www.handy-signatur.at.

[11] https://aws.amazon.com/cloudhsm/.

creates an interoperability framework that allows the user to use her keys on any device and at any time and therefore meets the modern heterogeneous device landscape with its distributed applications and multi-device users well.

In short, the user takes one of her devices and launches an application to perform some cryptographic task. The application interfaces with the interoperability layer, CrySIL, which connects to another device. This other device has access to the actual cryptographic primitive, creates and validates authentication challenges if required and performs the requested operation. The result is returned to the interoperability layer and back to the application running on the device of the user. A graphical illustration of the work flow is given in Fig. 1.

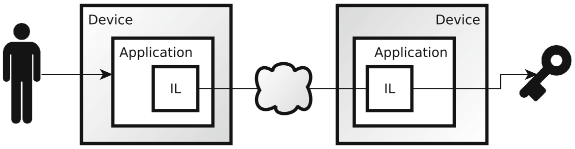


Fig. 1. CrySIL's basic architecture.

The flexible design of CrySIL allows for numerous deployment scenarios. First and foremost, the device having access to the key material can be the very same device the user interfaces with. Thus, CrySIL fulfills the use case requirements of using local cryptographic services – as provided by smart cards. Further, the device can be a cloud service provider offering the service to a broad range of users. This scenario reflects the central service paradigm as seen in industry solutions. A rather novel deployment scenario possible with CrySIL is to move the central cloud-based service to a user's mobile device. This scenario is motivated by the relative high level of security that can be offered by a mobile device when encryption systems and access protection systems are activated and correctly configured.

4.1 Interoperability Layer Node

The heart of our CrySIL approach is the interoperability layer node (denoted as IL in Fig. 1). The modular node design enables the flexibility and extensibility of our approach while keeping the overall architecture simple.

Most conventional cryptographic service providers receive commands, perform the required actions, and return the result to the caller. To achieve CrySIL's interoperability goal, CrySIL breaks the classic cryptographic provider apart. The resulting modules have different jobs and work together to form the actual cryptographic service provider. The most visible modules are command *receivers* and the modules which act on cryptographic primitives – *actors*. Another crucial module is responsible for connecting *receivers* and *actors*. Other modules handle

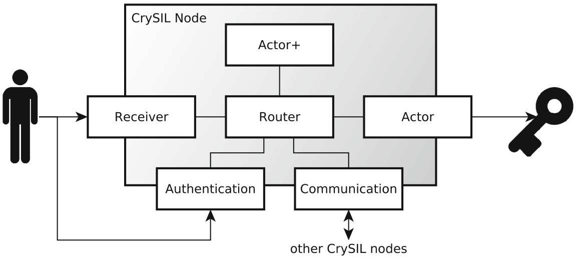


Fig. 2. CrySIL node architecture overview.

inter-node communication, protocol mappings, advanced crypto and authentication. All are considered as building blocks and are not restricted to any technology, platform, or programming language. An illustration of modules and their interconnections is given in Fig. 2.

A *receiver* offers a set of cryptographic functions to the application developer. Being a design concept, a *receiver* can be implemented to run on any device, any platform, and any technology. A *receiver* does not perform any cryptographic operations but interfaces with the routing module after having the command encoded as interoperability protocol command. A realization of a receiver can be cryptographic APIs on a programming language level like JCE, CSP, or the W3C Web Cryptography API. Another realization can serve as a web-service providing a SOAP-based cryptographic interface. Having a realization that interfaces with clients of the PKCS#11 standard or PC/Smart Card daemons (PCSCd) can bring remote key storage capabilities to existing applications.

An *actor* makes the contents of a specific key provider available to the interoperability layer. A key provider hosts key material and performs the actual cryptographic operation. The *actor* can be implemented to connect arbitrary key providers to CrySIL. Sample key providers are smart cards attached to a PC, USB-Tokens, Software Security Modules (SSMs), as well as Hardware Security Modules (HSMs). An *actor+* might provide high-level cryptographic methods such as CMS encryption or XMLDSIG signatures while using the cryptographic primitives of other *actors*.

The central routing module – the *router* – receives commands from the *receiver* modules and assigns them to the appropriate *actor* modules. CrySIL supports a many-to-many relationship between *actors* and *receivers* in a completely transparent way.

4.2 Inter-Node Communication

The use cases ask for cryptographic primitives and services to be available anywhere and at any time. CrySIL's answer is transparent off-device cryptography, which mandates inter-node communication. Off-device cryptography allows a device that is not capable of doing a certain cryptographic operation on its own to use the cryptographic engine of a remote service. The device therefore can do the operation at the cost of having to trust the remote service.

The *communication* modules are in charge of inter-node communication. They simply take a request and send it to another off-device communications module. The most basic implementation is HTTP(s). Yet, arbitrary transport protocols, such as HTML5's Cross Document Messaging, Web-sockets, or IPSec are suitable.

To enable multi-hop communications, i.e. routing a request through multiple CrySIL nodes before the request reaches its target, we integrated end-to-end encryption capabilities into the protocol. We decided to tunnel a TLS v1.2 connection through CrySIL requests/responses. That gives us a couple of advantages. First, we get a mature solution regarding the privacy and security of the data. Second, we can use the organizational overheads that come with a PKI infrastructure. I.e. certification authorities, revocation services and revocation lists as well as certificate pinning capabilities. All in all, end-to-end encryption prevents any intermediate nodes from eavesdropping on the transmitted data.

Putting the pieces together, CrySIL renders off-device crypto completely transparent to the user, the developer, and to the application while maintaining a simple architecture. With inter-node communication, there can be one or multiple nodes per device. The resulting architecture is depicted in Fig. 3. Our approach enables almost complete key flexibility. A user benefits from her ability to use a variety of different keys provided by different key providers from a variety of applications on different devices.

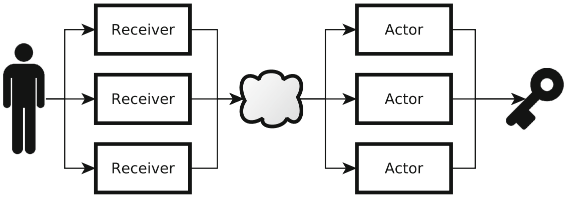


Fig. 3. Interoperability architecture view.

4.3 Authentication

Whenever cryptographic keys are used by an application, there must be access and usage policies in place. For local deployments, the simplest policy is that everyone can use the key. Similarly, a policy might require a PIN code for a local smart card. Such rather simple policies can be enforced by the device or the operating system. However, the cross-device/inter-node key access feature of CrySIL asks for a policy enforcement system that meets the requirements beyond in-device solutions.

We identified the key to be the main asset in the CrySIL infrastructure. The key belongs to the user and therefore, the user has to decide whether it can be used for a certain operation. This principle is well-established and used in different HSM realisations such as smart cards or TPMs. To unlock the key and operation, a key provider requires some information from the user to authorize the requested operation. A cryptographic smart card for example may only allow using a key when the correct PIN is provided. As a consequence, within CrySIL only the *actor* knows about the authentication requirements of its key store. To be exact, the *actor* is the only entity who can know which authentication requirements need to be fulfilled in order to attempt to unlock the key. The *actor* can also enforce additional policies. Having a custom set of policies, an *actor* can provide an interface to some sort of key database backend, where the key owner himself can request a specific set of authentication information per key.

The CrySIL infrastructure is designed to handle authentication concerns internally as well. The solution therefore keeps its ease-of-use-nature for developers but also gains security by keeping authentication concerns away from the application. The process works as follows: The *actor* requests the information needed to fulfil the constraints and requirements to use a certain key. It does so by utilizing the challenge/reply service offered by the interoperability layer protocol. The challenge is sent to the node where the crypto operation request originated from. Once there, the challenge hits the *router* and the *router* identifies the message as authentication challenge. The challenge is therefore assigned to the authentication modules which handle the user interaction. The user provides the required credentials. When the information is collected, the authentication modules create a challenge reply and send it back to the *actor*. The *actor* receives the information and attempts to perform the requested crypto operation. Usually the key store itself enforces the policy, but the *actor* may do it as well. As a result, the key store/*actor* either authorizes the operation based to the supplied information and returns the result, or aborts the request.

The authentication modules offered by CrySIL are organized in a flexible and extensible manner. Different authentication modules can handle different authentication concepts from a simple PIN query over OpenID Connect to strong two-factor authentication systems. A developer can use any service with any authentication mechanism but does not have to bother with authentication concerns. Applications can therefore offer strong authentication mechanisms with little extra effort. Furthermore, authentication modules are designed to keep sensible credentials away from the *receiver* and therefore from the application. A malicious application therefore might have a harder time to eavesdrop or attack the credentials which results in an overall security boost.

Last but not least, the interoperability layer protocol foresees the use of session information. The session feature allows an *actor* to create one of the established session management systems to allow sessions with lifetime exceeding that of a single request. An application can therefore use an authorized key a number of times before he has to re-authenticate again. That might enable the user to accept stronger authentication as a hurdle during the key unlocking procedure.

5 Evaluation

In order to demonstrate and evaluate our approach, we have implemented a number of prototypes in the course of our research. The focus lies on evaluating the flexibility and modularity of the CrySIL building blocks in order to solve different use cases.

In this section, we will describe our protected-data-at-rest prototype in terms of features, deployment scenarios, practical applicability, and benefits over other solutions. The prototype addresses the scenario where a user gets some data from a friend and wants to use the data on multiple devices. The objective is that the data is encrypted whenever it is not on a device owned by the user or the friend. The scenario is depicted in Fig. 4. The setup is done with state-of-the-art communication and crypto. Trust relationships as well as attack vectors are well known from any hardware security module deployment scenario. Therefore, a thorough security analysis would not yield new information on security and trust issues, and is therefore omitted.

5.1 Deployment Scenario

The users, denoted as user 1 and user 2, have no profound understanding of cryptography and are using web browsers on PCs and mobile devices. It is assumed that for users who are no experts in the field of IT security, it is too much of a hurdle to perform manual key exchange in a secure way.

The storage solution used in the scenario is defined to be some shared storage solution with instant sharing. For example, a public cloud storage service like Dropbox [12]. This solution solicits no manual exchange of the data like it would be when using technologies like instant messaging or electronic mail and therefore keeps the prototype clean and easy to understand.

On either device, be it the phone, the tablet, or the PC, a web browser runs an HTML5 and JavaScript browser application. The application encrypts arbitrary data before it submits the data to the shared storage service with the help of the W3C Web Cryptography API. In this case, the API is implemented by JavaScript

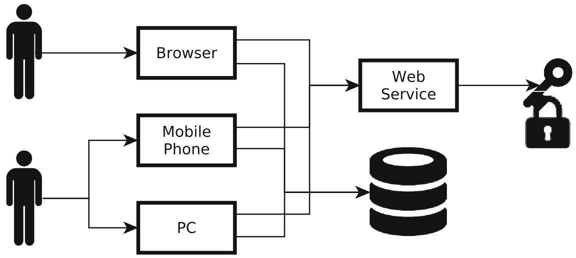


Fig. 4. Prototype: protected-data-at-rest deployment scenario.

[12] https://dropbox.com.

version of a CrySIL node. This node is referred to as the application's CrySIL node later on. Hiding behind the Web Crypto API, the application's CrySIL node offers transparent access to a remote key service over HTTPS.

The key service – acting as trusted third party – is a Java 8/Spring 4 web application hosted on a Tomcat server an referred to as key service CrySIL node later. An HTTPS communication module offers remote access. An *actor+* capable of handling CMS containers is available as well as an *actor* interfacing with a software key store. Key store access is constraint by the *actor* as follows:

- To retrieve the certificate of the key of user 1, it is sufficient to provide the correct identifier of the user.
- Using the key for decryption purpose, however, requires 2-factor authentication with a username/password tuple in the first place and a mobile TAN[13] as second factor.

To keep the prototype simple, we settled with some less secure implementations. A hardware key store (provided by an HSM) for the key service would bring a major boost to the overall security. However, for the sake of time and simplicity, the current implementations are based of software key stores.

5.2 Use Case

User 2 starts the browser application and add some data that he wants to share with user 1. Before the browser application sends the data to the cloud storage, it interfaces with the CrySIL infrastructure to encrypt the data. This is where the browser application hands over the control flow to the application's CrySIL node. The flow is only returned to the browser application when the encrypted data arrives. An illustration of the control flow is given in Fig. 5.

Being asked to encrypt data, the application's CrySIL node asks user 2 to select a certificate that should be used to encrypt the data. User 2 selects the key service and the application's CrySIL node requests a list of certificates from the key service CrySIL node. In any case, the key store actor of the key service CrySIL node answers with an authentication challenge, demanding an identifier. The challenge is picked up by the appropriate authentication module within the application's CrySIL node. The module interfaces with the human – user 2 – and shows a graphical user interface where it asks for the required identifier of user 1. User 2 provides the identifier. The authentication module sends the information back to the key service CrySIL node which answers with the certificate of user 1. Note, that the authentication information has never been processed by the browser application itself.

[13] Mobile transaction numbers (TANs) are a two-factor authentication process where the user proofs knowledge of a secret and possession of a device, i.e. a mobile phone. The secret is used to identify the user and therefore the mobile phone. Then, a nonce is sent to the phone. The user has to proof knowledge of the nonce. The user can know about the nonce if and only if she has access to the mobile phone in question. Mobile TANs are broadly used for authentication in banking, industry, and cloud services.

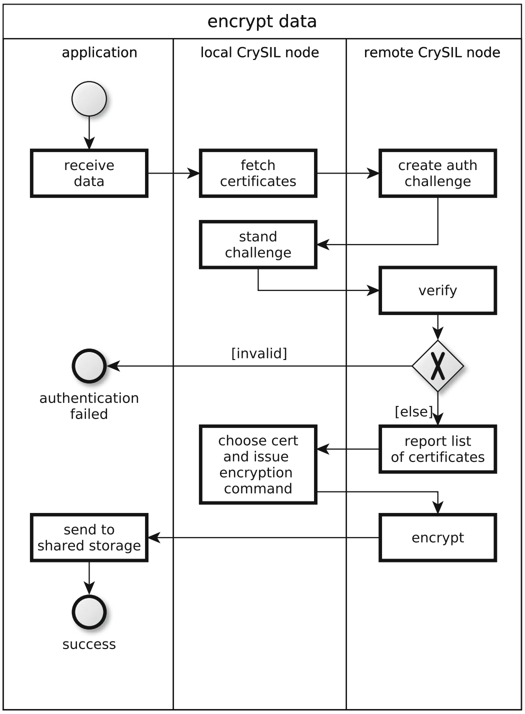


Fig. 5. Prototype encrypt command flow.

The application's CrySIL node now sends another request to the CrySIL infrastructure and asks for encrypting the data with the just retrieved certificate using the CMS standard. The *router* of the application's CrySIL node decides that it has no means of doing CMS locally and therefore forwards the whole request to the key service CrySIL node. The key service node receives the request, its *router* forwards it to the CMS *actor+* which in turn creates a CMS container from the supplied data and returns it to the *receiver* of the application's CrySIL node within the browser application. Although sending a whole document causes communication overhead, this example highlights that a platform (the web browser) lacking the required cryptographic APIs is still able to create a CMS document. With the key service being a trusted party and the communications being secured by HTTPS, the encryption process is considered as secure.

The resulting CMS container is returned to the browser application which in turn sends it to the cloud storage after authenticating there.

Now user 1 can receive – i.e. read – the CMS container which was just submitted to the shared storage by user 2. The process is similar to the one described above and illustrated in Fig. 5. The user takes one of her devices and downloads the encrypted data. She uses the same browser application to decrypt the CMS file with the help of the CrySIL infrastructure. She has to go through the process of selecting a certificate and standing the challenge and standing a two-factor authentication prior to the decryption process until the result is available in the application's CrySIL node and therefore in the application.

5.3 Discussion

Our approach has a number of advantages over conventional solutions. First and foremost, a centralized key storage location enables a user to access her keys at any time and anywhere. The only dependency is an Internet connection, but by having an Internet-accessible cloud storage service for data storage renders this dependency fulfilled whenever cloud access is possible. Therefore, this feature closes the gap between classic cryptographic service providers and upcoming requirement to serve multi-device users.

When sensitive key material is not stored on the device itself, there is no need to share key data between devices. The risk of exposing the sensitive key data during transmission is thus foreclosed completely. There is no need to align key storage solutions to be able to translate key material where interfaces and transmission channels are very restricted.

Not having access to the sensitive key material of one key on different devices reduces the attack vectors against the key drastically. Especially, since browsers for example are most vulnerable in terms of protecting sensitive key material. The private key material never leaves the key provider environment.

The level of device and cryptographic expertise required from developers is lower. The developer can focus on creating a feature-rich application, well tested and stable software instead of dealing with the peculiarities of authentication and key exchange and secure key storage on the devices in question. And, nonetheless, create an application that uses cryptography and enhances the privacy and security of the user and her data.

Last but not least, CrySIL offers not only off-device key storage but also off-device cryptographic functions. A cryptographic service provider which offers high level cryptographic methods such as CMS or XMLDSIG for remote use enables a broad range of devices and applications to use cryptography to protect the users' data and privacy. With that, the increasingly popular browser applications are enabled for the use of cryptography in a much more secure manner than with local key storage.

Finally, the CrySIL infrastructure relies solely on well-known building blocks of cryptography. The security aspects of key stores, cryptographic providers, as well as the communication solution are commonly known and well-understood.

All the advantages come with the cost of yet another trusted third party. Establishing certifications and trust relationships are still required and come with all advantages and drawbacks of this concept. Yet, the CrySIL infrastructure also supports the deployment of key stores on the user owned devices (e.g., home servers or mobile devices), or directly supports local crypto devices, such as smart cards.

5.4 Performance

As for performance, CrySIL does not implement any cryptographic service itself. It solely integrates existing solutions and makes them accessible over various APIs even on other devices.

The infrastructure adds some overhead in the process of redirecting a command to a crypto service. Having to collect authentication information does require some time for fetching the requirements, creating the challenge and reading the response. This time lost is minimal compared to the time a user needs to enter the required information.

Having an off-device scenario, there is of course some delay when sending commands via the Internet. Thus, in addition to performing the actual crypto process and the redirecting process one round-trip-time has to be added per command.

Anyhow, these performance measurements are made based on our prototypical implementation. The implementation has not received any performance optimizations due to the fact that the main goal is to create availability and not speed.

5.5 Integration Efforts

Whenever an application utilizes well-known crypto APIs, CrySIL can be integrated with an effort next to none. As of today, we work on receivers for PKCS11, JCE (for desktop and Android), MS CNG, W3C Crypto API, OpenSSL and OpenSC.

In case a platform does not have the required modules available, one can easily implement such a module. The Java JCE receiver module for example is implemented using only 1000 lines of prototypical code including some functionalities that are not supported by the JCE framework. Our Java router and sending communications modules do have 80 loc each with a common protocol definition of 1500 loc. The code of imported libraries are not included in the numbers given.

6 Recent Advances

CrySIL has been used successfully to leverage alternative key distribution approaches for today's multi-device user scenarios. With CrySIL's mobile extension MoCrySIL [26], for example, the authors managed to move the central key service to the key owners sphere of influence, namely her smart phone. Today's smart phones tend to offer hardware protected key storage and therefore can protect sensitive cryptographic primitives way better than a standard desktop computer can. Furthermore, CrySIL has been used to create an alternative to the classic PKI way of choosing a recipients key [27]. Instead of trusting a certification authority, the data sender creates a key for a certain recipient on his own key service. He then allows the recipient to use the key given the recipient is able to stand the required authentication challenge. With that, the sender has a couple of advantages. First, the sender can create a key that is suitable for the task and does not have to use what a recipient offers (which can be no key at all). Second, the sender can trust his own key and key service. And third, the recipient authenticates directly to him. Last but not least, CrySIL has been used to emulate attribute-based encryption schemes [28] and therefore make these schemes available to PKI systems.

7 Future Work

Our approach complements classic solutions so that the new requirements of distributed applications deployed in today's heterogeneous device landscape can be met. However, there are still gaps neither the related work nor our approach can solve currently.

The first gap is the need to move authentication away from the application. Our approach succeeds in moving the authentication to the library and therefore preventing the sensitive credentials to be directly processed by the application. Since the library runs inside the application and shares its memory, an attacker i.e. a malicious application might still be able to eavesdrop or tamper with the sensitive information. Having the credentials not reaching the application in the first place would foreclose this attack vector completely. For web applications, this could be realized by using a separated iFrame, in case of mobile devices, specific CrySIL apps could be used that are utilized by other apps via IPC calls.

Other future use cases include the emulation proxy-re-encryption schemes by using flexible and fine grained authentication systems. E.g., proxy-re-encryption schemes could be emulated by handing out authentication tokens to third-parties who – by supplying these tokens – are allowed to re-encrypt data for specific recipients.

These examples represent a small collection of possible future directions.

8 Conclusions

The deployment of cryptographic functions for distributed applications deployed in today's heterogeneous device landscape and storing and handling key data in a secure way faces many challenges in terms of lack of platform support and high complexity for the development teams. One way to approach these challenges is the introduction of central services that deploy secure key storage facilities and provide APIs that can be used on arbitrary devices. Several companies already offer such systems for the deployment of specific cryptographic functions. However, those system lack the flexibility in terms of supported cryptographic algorithms and protocols and have not been intended for generic use cases.

Therefore, this work presents the Crypto Service Interoperability Layer which offers a highly flexible architecture that is capable of combining central and local cryptographic services. The current system has already been successfully used for several prototypical applications and is constantly improved by adding additional support for cryptographic algorithms and APIs for different platforms and devices.

References

1. Reimair, F., Teufl, P., Zefferer, T.: WebCrySIL - web cryptographic service interoperability layer. In: Proceedings of Web Information Systems and Technologies (WebIST), pp. 35–44 (2015)
2. Egele, M., Brumley, D., Fratantonio, Y., Kruegel, C.: An empirical study of cryptographic misuse in android applications. In: Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security - CCS 2013, pp. 73–84. ACM Press, New York (2013)
3. Fahl, S., Harbach, M., Muders, T., Smith, M., Baumgärtner, L., Freisleben, B.: Why eve and mallory love android. In: Proceedings of the 2012 ACM Conference on Computer and Communications Security - CCS 2012, p. 50. ACM Press, New York (2012)
4. Trusted Computing Group: TCG TPM specification version 1.2 revision 116 (2011). http://www.trustedcomputinggroup.org/resources/tpm_main_specification. Accessed 29 January 2013
5. The European Parliament and the Council of the European Union: Directive 1999/93/EC of the european parliament and of the council of 13 December 1999 on a community framework for electronic signatures. Offcial J. Eur. Commun. L **013**, 12–20 (2000)
6. van Hoboken, J.V.J., Arnbak, A., van Eijk, N.: Cloud computing in higher education and research institutions and the USA patriot act. SSRN Electron. J. (2012)
7. Naehrig, M., Lauter, K., Vaikuntanathan, V.: Can homomorphic encryption be practical?. In: Proceedings of the 3rd ACM Workshop on Cloud Computing Security Workshop - CCSW 2011, pp. 113–124. ACM Press (2011)
8. Bellare, M., Boldyreva, A., O'Neill, A.: Deterministic and efficiently searchable encryption. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 535–552. Springer, Heidelberg (2007)
9. Camenisch, J.L., Shoup, V.: Practical verifiable encryption and decryption of discrete logarithms. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 126–144. Springer, Heidelberg (2003)
10. Ateniese, G., Fu, K., Green, M., Hohenberger, S.: Improved proxy re-encryption schemes with applications to secure distributed storage. ACM Trans. Inf. Syst. Secur. (TISSEC) **9**(1), 1–30 (2006)
11. Hanser, C., Slamanig, D.: Blank digital signatures. In: Proceedings of the 8th ACM SIGSAC Symposium on Information, Computer and Communications Security - ASIA CCS 2013, p. 95. ACM Press, New York (2013)
12. Fujisaki, E., Okamoto, T., Pointcheval, D., Stern, J.: RSA-OAEP is secure under the RSA assumption. J. Cryptol. **17**, 81–104 (2004)
13. Breu, F., Guggenbichler, S., Wollmann, J.: PKCS #1: RSA encryption version 1.5. Vasa, pp. 1–19 (2008)
14. Barker, E., Johnson, D., Smid, M.: NIST Special Publication 800–56A Revision 2 - Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography. Nist Special Publication, New York (2013)
15. Apple: iOS Security - White Paper. Technical report (2014)

16. Kaliski, B.: PKCS #5: Password-based cryptography specification version 2.0 (2000)
17. Percival, C.: Stronger key derivation via sequential memory-hard functions. Self-published, 1–16 (2009)
18. Teufl, P., Zefferer, T., Stromberger, C., Hechenblaikner, C.: iOS encryption systems - deploying iOS devices in security-critical environments. In: International Conference on Security and Cryptography, pp. 170–182 (2013)
19. Housley, R.: Cryptographic message syntax (CMS). RFC **5652**, 1–57 (2009)
20. Turner, S.: Secure/multipurpose internet mail extensions. IEEE Internet Comput. **14**, 82–86 (2010)
21. Eastlake, D., Reagle, J., Solo, D., Hirsch, F., Roessler, T.: XML Signature Syntax and Processing, 2 edn., pp. 1–59. W3C Recommendation (2010)
22. Slamanig, D., Stranacher, K., Zwattendorfer, B.: User-centric identity as a service-architecture for eIDs with selective attribute disclosure. In: Proceedings of the 19th ACM Symposium on Access Control Models and Technologies - SACMAT 2014, pp. 153–164. ACM Press, New York (2014)
23. Leitold, H., Hollosi, A., Posch, R.: Security architecture of the Austrian citizen card concept. In: Proceedings of 18th Annual Computer Security Applications Conference (2002)
24. Orthacker, C., Centner, M.: Minimal-footprint middleware to leverage qualified electronic signatures. In: Filipe, J., Cordeiro, J. (eds.) WEBIST 2010. LNBIP, vol. 75, pp. 60–68. Springer, Heidelberg (2011)
25. Orthacker, C., Centner, M., Kittl, C.: Qualified mobile server signature. In: Rannenberg, K., Varadharajan, V., Weber, C. (eds.) SEC 2010. IFIP AICT, vol. 330, pp. 103–111. Springer, Heidelberg (2010)
26. Reimair, F., Teufl, P., Feichtner, J., Kollmann, C., Thaller, C.: MoCrySIL - carry your cryptographic keys in your pocket. In: Proceedings of the 12th International Conference on Security and Cryptography, pp. 285–292 (2015)
27. Reimair, F., Teufl, P., Prünster, B.: In Certificates We Trust - Revisited. In: Proceedings of the 14th IEEE International Conference on Trust, Security and Privacy in Computing and Communications, pp. 702–709 (2015)
28. Reimair, F., Feichtner, J., Teufl, P.: Attribute-based Encryption goes X.509. In: Proceedings of the 12th International Conference on e-Business Engineering, ICEBE (2015) (in press)

Society, e-Business and e-Government

Motivating Online Teams: An Investigation on Task Significance, Coordination and Incentive Mechanisms

Juliana de Melo Bezerra(✉), Lara Santos Diniz, Victor da Silva Montalvão, and Celso Massaki Hirata

Computer Science Department, ITA, Sao Jose Dos Campos, Brazil
{juliana,hirata}@ita.br, araldiniz@gmail.com, vsmontalvao@gmail.com

Abstract. Online teams can help to increase organizational effectiveness, due to their flexibility and facility to gather resources originally dispersed. In order to gather team participation, we need to address team motivation. Belief in task significance has positive effects on moving members to participate. Coordination is a central to a successful collaboration, since it guides members' participation. Incentive mechanisms can be used to foster online participation. We then conducted an experiment to analyze the effects on motivation of task significance, quality of team coordination, and usage of incentive mechanisms. We confirm that the three aspects influence motivation positively. We discuss characteristics that make them contribute to or interfere on online motivation. We also investigate the interplay among the aspects that can increase or reduce the aspects' effect.

Keywords: Online team · Motivation · Task significance · Coordination · Incentives

1 Introduction

Online teams are a strategy to global organizations keep competiveness. They are used in distinct kinds of organizations, including those based on research, product development, or service provision. Other contexts can take advantage of online teams, for example, the educational area, which uses online teams to prepare students to online demands of global organizations [23], and to promote international collaborative learning [10].

The main advantage of online teams is their virtual characteristic, driven by the usage of information and telecommunication technology through Internet. Individuals in online teams collaborate to accomplish tasks, by overcoming barriers related to difference in geography and time. Individuals can be assembled together in response to specific needs for an often short period of time [16, 30]. The reduction of time and expenses is an important benefit brought by online teams in contrast to traditional teams that require individuals to be collocated. Online teams then rely on flexibility to gather valuable resources and knowledge originally dispersed among people [22, 24].

The separation in space and time, faced by online teams, drives both structural and contextual issues that can in turn imposed challenges to participation. Examples of challenges include the lack of social context, the limitation of informal communication,

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 93–108, 2016.
DOI: 10.1007/978-3-319-30996-5_5

and the difficulty in providing shared context, visibility and knowledge transfer [8, 30]. Such issues can even negatively affect trust, cohesion and relationship building among members [33].

An online task to be accomplished requires participation of members. However, it is not easy to obtain participation, since it is mainly driven by motivation [19]. Our focus is then on motivation of online teams. Motivation was found to be positively affected by how members believe in the significance of the task being developed [32]. Coordination during the task development is seen as a serious issue that, if not well conducted, can damage team success [3, 8]. The use of online incentive mechanisms is often advocated as a way to influence motivation aiming to foster members' participation [26, 35].

In this paper, we conduct an exploratory experiment to investigate effects on motivation of three aspects: task significance, coordination, and incentive mechanisms. We investigate characteristics that make these aspects contribute or not to members' motivation and in turn to team success. Team success is analyzed through the obtained participation, by assessing team effectiveness with respect to team performance and team satisfaction. We also reason about the interplay among the three aspects, in a way to identify possible interferences that make one aspect ineffective in the presence of another.

The paper is organized as follows. In the next section, we discuss the background of our research regarding motivation in online teams. Later, we define our research questions. In Sect. 4, we explain how we designed the experiment. In Sects. 5, 6, 7 and 8, we present the experiment results and analyze them. In Sect. 9, we conclusions and future work are presented.

2 Motivation in Online Teams

Motivation can be defined as an impulse to act according to one's desires. If people are motivated, they choose to realize something, because it has a meaning for them [37]. Hersey et al. [18] propose a model, shown in Fig. 1, which is useful to understand the origins of the behavior of a person in offline organizations. Although Hersey et al.'s model was initially proposed to individuals acting on offline group or community, it also suits online contexts.

According to the motivation model (Fig. 1), *motives* are something inside the individual that moves him/her to act. *Objectives* can be understood as expected achievements

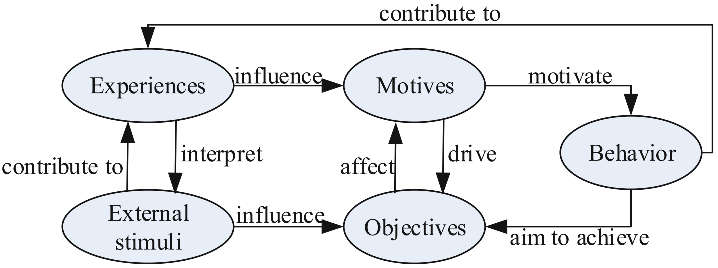


Fig. 1. Motivation model [4, 18].

that satisfy the motives. The *behavior* includes the tasks performed by the individual in order to reach the *objectives*. So, *behavior* comprises participation. The *motives* (or motivation itself) of an individual can be influenced by both his/her *experiences* directly and the *external stimuli* indirectly. *Experiences* are acquired during life and include personality, education, and values. In this way, *behavior* aims to achieve *objectives* and can also contribute to the composition of experiences.

External stimuli, in Fig. 1, refer to facilities or limitations that the environment imposes to the objectives' achievement. *External stimuli* may be ephemeral or eventual, i.e., they can be seen as opportunities or temporal restrictions that may not persist or may occur in the future. The main idea driven by the motivation model and used in our research is that the way to gather participation is through motivation, and motivation can be influenced by distinct aspects.

Aiming to understand the structure of online teams, we study definitions of online communities. An important definition is the one proposed by Preece [31]. Preece [31] defines virtual community as a group of people, who come together for a purpose online,

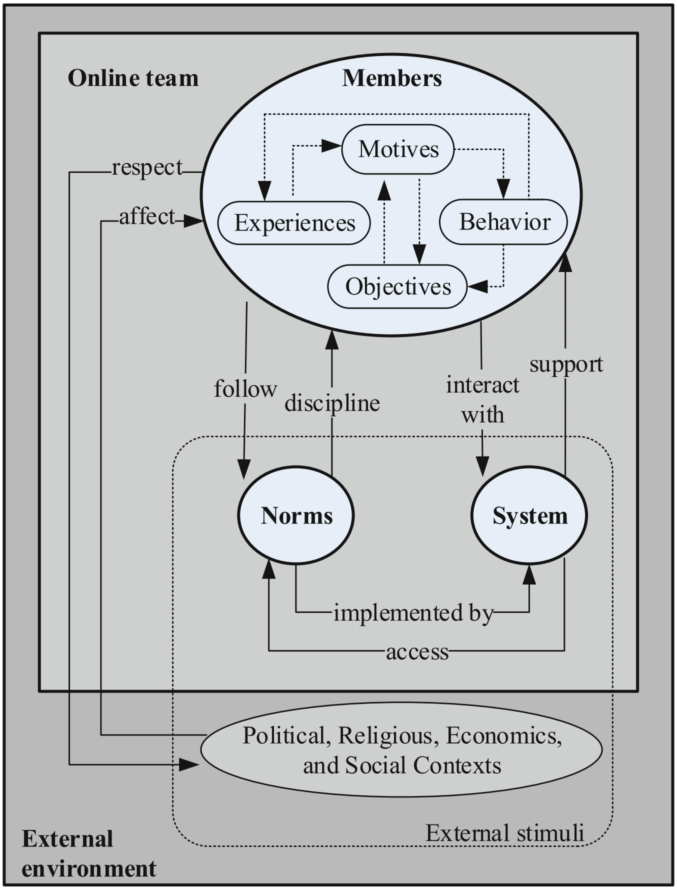


Fig. 2. Online team and motivation.

and who are governed by norms. Based on this definition, Melo Bezerra and Hirata [4] propose a model that represents a virtual community inside an external environment. The model also indicates the three main elements in a virtual community: *members*, *system*, and *norms*. Here we use such model to reason about online teams. We argue that the main elements of online teams are the same as in virtual communities. Important differences are regarding, for example, the quantity of members involved (in general, few members in online teams), and the period of time that members are assembled together (in general, online teams have short-term duration).

Figure 2 shows the integration between the online team model (adapted from [4]) and the motivation model (Fig. 1). *Members* are a group of people that belong to a team, and they should be aligned with the shared purpose of the team. *Members* in general respect the directives of the *external environment* in which they are immersed. In the other way, the external environment can affect the team *members*. *Norms* are specific to a social context, and they are generally defined in order to regulate the people relationships. *Norms* impose discipline to *members*, while *members* follow *norms*. *System* encompasses the means for the *members* to work in their activities and interact with other members. So, *members* interact with the *system*, whereas the *system* supports *members*. Each *member* in an online team characterizes an individual with his *experiences*, *motives*, *objectives*, and *behavior*. The *external stimuli* comprise *norms*, *system*, and *external environment*. The integrated model (Fig. 2) is useful to analyze the factors that influence motivation in online teams, as described in the next section.

3 Research Questions

In this section, we use the model in Fig. 2 to contextualize the importance of the three chosen motivating aspects investigated in this paper considering online teams: task significance, coordination, and incentive mechanisms.

In an online team, the participation (indicated as behavior in Fig. 2) of a member can be perceived through his/her performed tasks in the system. So, task is a relevant issue related to motivation. Characteristics regarding a task being developed by an online team can make the task more suitable to gather members' participation. For instance, decomposability is an attribute that makes possible to structure the task into sub-tasks and engender dependences between members [29]. Autonomy is also important, insofar as team should be self-sufficient to solve the problem regardless of external resources. The complexity of a task is also impacting: a task should be complex enough to require interaction of members, but always considering members' skills [8, 34]. These characteristics are important when defining a task to be performed by a team. A particular characteristic, which is task significance, is critical. According to Staples and Cameron [34], a task should be significant to members to make they believe in its importance and purpose. Regarding task significance, we are interesting to answer the following research questions:

Research Question 1 (RQ1): Is task significance an aspect that positively affects motivation in online teams?
Research Question 2 (RQ2): How to define a task with significance?

Figure 2 illustrates only one member, however in a team there are several members (with their particular characteristics) interacting through the system. Teamwork then presupposes collaboration, which is realized through communication and coordination supported by the system [12]. Collaboration follows the norms established for that team. Communication is required to make members exchange information, negotiate and take decisions. The virtual nature of communication makes online teams prone to problems, such as false interpretation of behavior in case of non-response of messages or incorrect use of text emphasis [5, 11]. Although communication issues can damage team motivation, they are more limiting to larger groups or virtual communities.

Coordination refers to the management of activities being carried out by members [15]. According Ellis et al. [12], coordination is the integration and harmonious adjustment of individual work efforts towards the accomplishment of a larger goal. Some studies indicate directives to enhance online coordination, for instance to specify intermediate deadlines and to promote training of members with the online environment previously [23]. Other studies have investigated organization types used by online teams, including fixed organization [29], self-organization [6], and shared leadership [32]. As coordination includes activities to organize members' actions and interactions to accommodate task execution respecting its schedule, it can be impeditive to team success [3, 8]. In this way, we intend to answer the following research questions:

Research Question 3 (RQ3): Is coordination an aspect that positively affects motivation in online teams?
Research Question 4 (RQ4): Which characteristics can influence coordination quality?

Incentive mechanisms are a strategy to influence individuals by addressing their motivations, aiming to improve participation. So, according to Fig. 2, incentives need firstly to address motivations of the individuals. Incentive mechanisms then act on members' motivation to induce a behavior. Research about motivation and incentives has been conducted considering both offline and online contexts of distinct teams and communities. In offline context, incentives are used, for example, in treatment of people with disorders [14, 38], and also to make employees reach better performance in organizations [28, 36]. In online context, incentives are applied, for instance, in areas as e-learning [17, 20], open-source software development [2, 13], and knowledge sharing [7, 9].

In case of online environments, incentive mechanisms are implemented in the system that supports members' collaboration. Popular online incentive mechanisms are those related to performance appraisal (e.g., to inform the value of one's participation) and social recognition (e.g., peer recognition, compliments, and praise) [21, 35]. Particular settings of online teams, such as those that belong to a company, can have intrinsic compensations (as monetary compensations or careers plans) that help moving members to participate. Teams based on volunteering face augmented challenges to promote participation [4, 27, 35]. In this paper, we aim to reason about the following research questions related to incentive mechanisms:

Research Question 5 (RQ5): Does the possibility of having incentive mechanisms positively affect motivation in online teams?
Research Question 6 (RQ6): Does the presence of incentive mechanisms positively affect motivation in online teams?

Related work investigates in a separated way the aspects of interest, including task significance, coordination and incentive mechanisms. Our objective is both to understand the relevance of these aspects to online motivation and consequently to team effectiveness, and to reason about the possible interference among such aspects. So we propose the additional research questions:

Research Question 7 (RQ7): How the aspects (task significance, coordination, and incentive mechanisms) are related to team effectiveness?
Research Question 8 (RQ8): Can the aspects (task significance, coordination, and incentive mechanisms) interfere in each other?

As we defined the research questions, the next section describe the experiment to explore such questions.

4 Experiment Design

In this section, we explain how the experiment was conducted using a qualitative approach. We describe the participants' characteristics, the type of online task to be developed, as well as the provided online platform.

We invited 32 students of an Engineering college to participate in our research. Their ages range from 18 to 25. They were divided randomly in four teams of eight people. Here we call the teams as A, B, C and D. Each team should work online in a distinct environment keeping team independence. The online task was to specify a project to be developed by future students of the Programming course. The project should be edited collaboratively, and all discussions should be held online. Anonymity was maintained inside each team in order to eliminate possibility of offline interactions.

The online environments were designed using MediaWiki as platform. A project should then be defined as a wiki page. For discussions, members should use the respective talk page. We installed LiquidThreads extension to empower the talk page with resources commonly found in forums, such as reply button, and automatic relation between question and answers. Teams A and B used this system. Teams C and D used this system with incentive mechanisms included, as described below.

In order to select incentive mechanisms, we used the foundation about online needs (known as the motives in Fig. 2). The classical Maslow's hierarchy of needs was adapted to the online context by Kim [25]. She explains each need as follows. *Physiological* need is related to system access, and the ability to participate online. *Safety* need, discussed together with the concept of security, refers basically to protection from hacking. *Belonging* is the need to be part of a group and to be accepted by it. *Esteem* refers to the need to be recognized by others due to participation. *Self-actualization* is the need to maximize own potential, by developing skills and opening up new opportunities. In the context of our experiment, *physiological* and *safety* needs are already satisfied with the designed system. *Belonging* needs are addressed since groups are defined and closed. We then focused on *esteem* and *self-actualization* needs.

We proposed three incentive mechanisms: "article feedback", "contribution scores", and "contribution appreciation". With these mechanisms, we aim to stimulate members' participation by allowing them, respectively, to receive feedback about their proposal, to be recognized by their contributions in the article, and to have their comments appreciated in discussions. Incentive mechanisms similar to these are commonly found in successful virtual communities, such as StackOverflow and Wikipedia [4]. They act mainly with motivations as prestige, visibility, reputation, recognition, competence, challenge seeking, and progress evaluation. The "article feedback" is a mechanism available as a MediaWiki extension. It allows readers to evaluate wiki articles using one to five stars. We invite other 10 students, different from team members, to act as readers and provide project feedback. The "contribution scores" mechanism is also a MediaWiki extension. It shows, at article footer, the names of members who contribute to article edition. We had to develop the "contribution appreciation" mechanism, which introduces "like" buttons in the questions and answers in a talk page with LiquidThreads extension.

The teams worked online during four weeks. At the end, each participant responded a questionnaire to evaluate his experience. Participants could also provide comments to explain their responses or to add new perspectives. In the next four sections, we detail the data collection and measures related to each investigated aspect. We also analyze the experiment results in order to respond the research questions.

5 Investigating Task Significance

Regarding the online task, we investigated the contribution of task significance to members' motivation, as well as what made a task attractive. Table 1 shows the questions, in the form of affirmatives to be evaluated by participants, the response options, and the associated results.

The respondents should evaluate the affirmative: "Task significance contributes to my motivation". The respondents should also evaluate if the following aspects contributed to task attractiveness: the "collaborative nature of the task", the "elaboration of a programming project", and the "possibility to use the project to future students". For the answer, we used the options: "strongly disagree", "disagree", "agree", and "strongly agree". The neutral option was removed to force respondents to make a decision, what is called the 'forced choice' method [1].

According to Table 1, a high percentage of participants (89 %) agreed or strongly agreed with the sentence "Task significance contributes to my motivation", which shows the relevance of the task definition. Among the aspects that made the task attractive, 81.5 % participants agreed or strongly agreed that the collaborative nature of the task contributed. To elaborate a programming project was considered relevant to task attractiveness for 85.2 % of the participants. The possible usage of the project to future students made the task attractive for 88.9 % of the participants. The feedback of participants was similar in teams. We did not observed relevant differences in teams about task as a motivation factor.

We observed that participants were really motivated with the task itself and its characteristics. It is important to understand that these characteristics were relevant for those participants in that context. Participants were students in a Programming course, so their activities were in general the development of programs. To specify a project was then

Table 1. Questions and results regarding task significance.

Topics	Response options	Results
"Task significance contributes to my motivation"	Strongly disagree	11 %
	Disagree	0 %
	Agree	45.5 %
	Strongly agree	43.5 %
The "collaborative nature of the task" contributes to task attractiveness	Strongly disagree	0 %
	Disagree	18.5 %
	Agree	60.3 %
	Strongly agree	21.2 %
The "elaboration of a programming project" contributes to task attractiveness	Strongly disagree	14.8 %
	Disagree	0 %
	Agree	50.5 %
	Strongly agree	34.7 %
The "possibility to use the project to future students" contributes to task attractiveness	Strongly disagree	7.4 %
	Disagree	3.7 %
	Agree	32.3 %
	Strongly agree	56.6 %

considered more appealing. Students of the chosen college live together in dorms, and they know each other. They found funny to design something to future colleagues to work on. It would be a mix of reception and retaliation to new students. Some participants reported that to participate in a research made them attracted to the task. Other participants commented that a positive aspect was the offline repercussion of their participation, since their roommates found the idea interesting and so they felt prestige.

Regarding *RQ1*, we found that task significance is, in fact, an aspect that positively affects motivation in online teams. So, in order to motivate members of a virtual team to perform an online task, the main directive is that the task should be attractive to them. Responding *RQ2*, to propose an attractive task is a challenge. We need first to understand the characteristics of the members and the context where they are settled. The analysis can include both online and offline attributes. Online attributes that made the task attractive in that context include the intrinsic collaborative characteristic of the proposed task, as well as the type of task (project elaboration instead of project execution). In the experiment, participants found the task interesting due to the possibility to employ the outcome to colleagues in future. Participants were also proud to participate in the initiative.

The repercussion of task results, and the reputation conquered for participating in a task are then examples of offline aspects that can also influence task significance.

6 Investigating Coordination

To reason about coordination, we asked about the contribution of the coordination satisfaction to members' motivation. Table 2 shows the questions, in the form of affirmatives to be evaluated by participants, the response options, and the associated results.

The respondents should evaluate the affirmative "Satisfaction with coordination contributes to my motivation", using the options: "strongly disagree", "disagree", "agree", and "strongly agree". We also asked participants to evaluate the following aspects:

Table 2. Questions and results regarding coordination.

Topics	Response options	Results
"Satisfaction with coordination contributes to my motivation"	Strongly disagree	0 %
	Disagree	26 %
	Agree	60 %
	Strongly agree	14 %
Evaluate "team commitment" in your team	Very poor	3.7 %
	Poor	22 %
	Normal	44 %
	Good	22 %
	Very good	8.3 %
Evaluate "deadlines' meeting" in your team	Very poor	0 %
	Poor	22.2 %
	Normal	26 %
	Good	26 %
	Very good	25.8 %
Evaluate "activities' division" in your team	Very poor	26 %
	Poor	52 %
	Normal	15 %
	Good	3.7 %
	Very good	3.3 %

"team commitment", "deadlines' meeting", and "activities' division". For these questions, the options of answer were: "very poor", "poor", "normal", "good", and "very good". Besides, we collected the amount of comments presented in forum in order to analyze participation in the communication process as result of the coordination activities.

As detailed in Table 2, participants, in 74 % of the cases, agreed or strongly agreed with the sentence "Satisfaction with coordination contributes to my motivation". The result shows that if people are satisfied with coordination, they can be more motivated and consequently perform better. Collaboration in teams revealed coordination problems related to "team commitment", "deadlines' meeting" and "activities' division". "Team commitment" to perform the task was considered "very poor" by 3.7 % of participants, "poor" by 22 %, "normal" by 44 %, "good" by 22 %, and "very good" by 8.3 %. Participants considered "deadlines' meeting" as "poor" in 22.2 % of the cases, "normal" in 26 %, "good" in 26 %, and "very good" in 25.8 %. The main problem was the division of activities among members. Participants considered "activities' division" "very poor" or "poor" in 78 % of the cases.

Teams explained how they organized themselves to perform the task collaboratively. The feedback was important to support the findings about problems related to task coordination. Team A commented that initially they discussed ideas of projects. After choosing a topic, one member elaborated a project and other members only complemented it. Before the end of the task, one member commented that the majority of the team stopped contributing. One member also assumed that he himself did not participate as desired. Similar problems were found in team B, where members discussed the initial ideas, but two members were mainly in charge of the project edition.

More aggravated problems were found in team C. In this team, one member gave the idea and practically elaborated the project alone. Other member mentioned that communication was difficult and suggested that to have few people in team could improve collaboration. One member commented that the online task demanded time to be accomplished and he was not available as expected. In team D, two members conducted the task by suggesting the theme and making the team develop the text. One member reported that he found grateful to collaborate online by exchanging experiences and observing others' algorithms. Other member commented that he contributed to the project by both elaborating and improving the text. According to one member, the team was very motivated and engaged. Other member said that the team was a little disorganized. Time and internet availability were reported as factors that limited the participation of one member.

We observed that problems related to coordination were presented in all teams but with distinct severity levels. Regarding communication, the quantity of comments in the forum was the following: 34 in team A, 42 in team B, 20 in team C, and 86 in team D. We observed that team C, which demonstrated more coordination problems, communicated less. Team D, with better task coordination, communicated more. It shows the importance of communication in the collaboration process, especially to achieve better coordination.

Answering *RQ3*, coordination, when well conducted, is an aspect that positively affects motivation in online teams. We noted, in the experiment, that the volume of online communication can reveal collaboration issues. Regarding *RQ4*, we investigated three coordination issues: "team commitment", "deadlines' meeting" and "activities' division". A relevant problem regarding coordination was the unfair division of activities in teams, mainly due to lack of engagement. A relevant issue is then to engage members in activities that they feel confident to perform, in order to gather contributions to fulfill the entire task.

7 Investigating Incentive Mechanisms

Regarding incentive mechanisms, we designed different questions for teams without incentives (A and B) and teams with incentives (C and D). For teams A and B, we made the following questions about each mechanism: "Could the incentive mechanism be useful?" and "Could the incentive mechanism motivate you?". For teams C and D, we asked: "Was the incentive mechanism useful?" and "Did the incentive mechanism motivate you?". The response options were only "yes" or "no". We also gathered comments about the systems that supported collaboration of the teams.

We analyzed the feedback of participants regarding the three developed incentive mechanisms: "article feedback", "contribution scores", and "contribution appreciation". In Table 3, we present the quantity of members that agree with the utility and motivation potential of the incentive mechanisms. For instance, regarding "contribution scores", in team D, 8 members said that it was useful and 6 members found it motivating. To better understand, we have to keep in mind that each team was composed by 8 members.

Table 3. Utility and motivation of incentive mechanisms.

Incentive mechanism	Characteristic	Quantity of members in teams			
		A	B	C	D
Article feedback	Useful	6	4	3	6
	Motivating	6	4	0	3
Contribution scores	Useful	5	8	5	8
	Motivating	5	4	0	6
Contribution appreciation	Useful	5	4	3	7
	Motivating	5	4	1	5

Teams A and B evaluated, in general, the incentive mechanisms as useful and they had a tendency to believe that mechanisms would be motivating. It shows that the incentive mechanisms were adequate to that context, and they really had a potential to motivate members. Comparing teams C and D, it is interesting to observe that both used the incentive mechanisms but they had very different experiences with them. Incentive mechanisms were more valuable for team D. We argue that coordination problems found in team C impacted, since members were not involved and did not use the mechanisms in fact. For instance, "contribution appreciation" mechanism is valuable if members contribute and they are able to evaluate others' contribution. According to the feedback of team D about the incentive mechanisms, the "article feedback" mechanism was less motivating. In this case, there were impediments that constrained the usage of the mechanism and consequently affected its motivating potential, for instance, the reduced number of external readers and the short-time characteristic of the task.

Participants also reported their experience with the systems that supported the online teams. One member in team A said that wiki features were not simple to use. Other member in team A added that wiki page is not easy to deal with. As a member in team D explained, the problem was mainly how to format text in the page that is a little complicated. We used wikis as MediaWiki platform provides. Although it seems to work well in successful communities like Wikipedia, it has limitations related to usability as reported by some participants. According to participants, a positive point of the system was the forum. In this case, the usability problem of talk pages in MediaWiki was overcome by the use of LiquidThreads extension. The extension makes transparent the need to add formatting to keep tracking between questions and related comments. Usability can be seen as hygienic factor to deal with. A hygienic factor [19] refers to a factor whose the presence is not stimulating, but the absence can reduce motivation.

Incentive mechanisms can be used to stimulate participation. To propose incentive mechanisms, it is required firstly the identification of motives that drive members. In the experiment, for example, we proposed incentives related to prestige, recognition, and reputation. Regarding *RQ5*, in the experiment, the feedback of teams A and B shows that the possibility of having the proposed incentive mechanisms positively affects motivation. However, answering *RQ6*, to have the incentives is not always stimulating. It may have interference of other factors not related to the incentives themselves, as we discuss below.

8 Investigating Aspects' Interplay and Team Effectiveness

We analyzed team effectiveness according both team performance and team satisfaction. Team performance is related to the delivery of a timely and high-quality product or service as result of online task. Team satisfaction is associated to the satisfaction of members after team interaction [22, 30]. In order to evaluate team performance, we asked two volunteers to act as evaluators by giving a grade (zero to ten) to team projects. Evaluators were students in the same college, but they were more experienced than the participants that performed the online task. They analyzed projects according to a defined criteria, which include: originality (if the project is different from the common programming activities), learning potential (if the project is able to improve programming learning in future students), attractiveness (if future students could be motivated to develop the project), and text quality (if the project is well written and easy to be understood). As team satisfaction is linked to the individual satisfaction of taking part of the team, we invited each participant to evaluate others' performance in the same team. We used the following options: 1 (very poor), 2 (poor), 3 (normal), 4 (good), and 5 (very good). The mean of grades assigned to a member can then indicate the quality of participation of that member.

The evaluators assigned the following grades to projects designed by online teams: 8.5 to team A, 9.8 to team B, 9.5 to team C, and 9.5 to team D. We observed that teams reached good performance. Problems related to coordination were not perceived looking only the quality of the team outcome. For instance, team C even with challenges reached a great result. We argue that it happened because the online task had an intrinsic characteristic of creativity. Teams could overcome their internal problems and elaborate a project with quality.

To reason about team satisfaction, we compute the mean of the performance evaluations of each member. The evaluations were made by co-workers with grades from 1 (very poor) to 5 (very good). Results are shown in Table 4. The data can reveal collaboration challenges found in teams. For instance, in Sect. 6, we commented that two members were mainly in charge of the task execution in team B.

Table 4. Members' assessment about team satisfaction.

Member	Team A	Team B	Team C	Team D
M1	4.1	3	2.6	4.3
M2	3.7	2.3	4.8	2.8
M3	2.7	2.8	3.2	4
M4	4.3	2.4	3.4	3.6
M5	3.8	5	3.4	2.3
M6	3	2.4	3.2	3.4
M7	4.2	2.7	3	4.9
M8	2.1	4.6	3.8	2.9

We can see in Table 2 that members M5 and M8 in team B were better evaluated than others. Other interesting case occurred in team D, where collaboration was reported to be more productive. We noted that three members (M1, M3 and M7) performed better, but others (as M4 and M6) also have their importance. The data regarding team satisfaction can also be misleading in case of problems with members' engagement. For example, in team C, we identified the member (M2) who participated more. The other grades are uniform and near to 3 (average performance). It may be interpreted as a regular participation. However, in fact, it indicates that members who did not participate were not able to assess others.

As task significance, coordination, and incentive mechanisms influence motivation, they are also important to team effectiveness, confirming *RQ7*. However, it is difficult to assess the influence to team effectiveness of each aspect separately. In order to measure team effectiveness in the experiment, we used team satisfaction and team performance. Team satisfaction is a good thermometer of online participation, since it can reveal coordination problems related to engagement. We measured team performance by assessing the quality of the developed task. As this measure only analyzes the outcome, it cannot explain possible participation issues during the process. Regarding *RQ8*, we observed the problems related to coordination especially in team C, during the experiment. A poor coordination negatively impacted members already motivated by the task. It also made incentive mechanisms lose force, for example, a mechanism to appreciate others' contributions is not useful if members do not contribute. We found that task significance, coordination, and incentive mechanisms can interfere in each other, in a way that a problem in one dimension can negatively affect others.

9 Conclusions

We used an experiment to investigate three aspects that influence motivation in online teams, including task significance, coordination, and incentive mechanisms. We also analyzed the impact of participation on team effectiveness, characterized by team performance and team satisfaction.

Task significance is essential to motivate members in online teams. It is the way to guarantee the initial attractiveness of members to participate online. Characteristics as task relevance and usefulness are then primordial. The quality of coordination is other aspect that influences online motivation. If the activity execution is well coordinated, the number of productive interactions increases, resulting in more motivated members. Disturbances in coordination can occur due to unfair division of activities among members, which ends up overloading some members. Problems with unequal participation among members can be revealed by team satisfaction, but obfuscated by analyzing only team performance.

Incentive mechanisms can be used to act on motivation and consequently improve participation. An effective identification of adequate tasks and design of effective incentive mechanisms have to consider members, their contexts, characteristics and intrinsic motives. Esteem and self-actualization needs constitute suitable categories of motivations to be addressed by online incentive mechanisms. Incentives mechanisms stimulate collaboration, when members are initially engaged, but if in the occurrence of problems, such as lack of coordination, the mechanisms are ineffective. So, incentive mechanisms work only if coordination and communication are properly assured.

The findings of our research are based on an experiment, so results and discussions should not be generalized, since they were drawn for a specific context. More experiments should be made in order to improve confidence on our initial results. We intend to expand our analysis by performing new experiments with more groups and distinct tasks. As future work, we plan to design features to promote online engagement in order to improve activities' division, by allowing members to define activities and identify responsibility. Incentive mechanisms will be used to support this new environment. As there is no formal guidance to design incentive mechanism, one future work is the definition of a process with directives to help designers to propose incentive mechanisms to members in online teams and virtual communities. Directives should include discussions about context, characteristics, and motivations of members.

References

1. Allen, E., Seaman, C.A.: Likert scales and data analyses. Qual. Prog. **40**, 64–65 (2007)
2. Bagozzi, R.P., Dholakia, U.M.: Open source software user communities: a study of participation in Linux user groups. Manag. Sci. **52**(7), 1099–1115 (2006)
3. Beise, C., et al.: A case study of project management practices in virtual settings. ACM SIGMIS Database **41**(4), 75–97 (2010)
4. de Melo Bezerra, J., Hirata, C.M.: Motivation and its mechanisms in virtual communities. In: Vivacqua, A.S., Gutwin, C., Borges, M.R. (eds.) CRIWG 2011. LNCS, vol. 6969, pp. 57–72. Springer, Heidelberg (2011)
5. de Melo Bezerra, J., Hirata, C.M.: Applying conflict management process to wiki communities. In: Zhang, R., Zhang, J., Zhang, Z., Filipe, J., Cordeiro, J. (eds.) ICEIS 2011. LNBIP, vol. 102, pp. 333–348. Springer, Heidelberg (2012)
6. Melo Bezerra, J., Hirata, C.M., Battagello, A.A.: Investigating collaboration and effectiveness of virtual teams with distinct organization types. In: Proceeding of the International Conference WWW/Internet (2012)
7. Bross, J., Sack, H., Meinel, C.: Encouraging participation in virtual communities: the "it-submit-blog" case. IADIS Int. J. WWW/Internet **2**, 113–129 (2007)
8. Casey, V., Richardson, I.: Uncovering the reality within virtual software teams. In: Proceedings of the International Workshop on Global Software Development for the Practitioner (GSD), pp. 66–72 (2006)
9. Chang, H.H., Chuang, S.-S.: Social capital and individual motivations on knowledge sharing: participant involvement as a moderator. Inf. Manage. **48**, 9–18 (2011)
10. Clear, T., Kassabova, D.: Motivational patterns in virtual team collaboration. In: Proceedings of the 7th Australasian Conference on Computing Education, vol. 42, pp. 51–58 (2005)
11. Crampton, C.: The mutual knowledge problem and its consequences for dispersed collaboration. Organ. Sci. **12**(3), 346–371 (2001)
12. Ellis, C., Gibbs, S.J., Rein, G.L.: Groupware: some issues and experiences. Commun. ACM **34**(1), 39–58 (1991)
13. Fang, Y., Neufeld, D.: Understanding sustained participation in open source software projects. J. Manage. Inf. Syst. **25**(4), 9–50 (2009)
14. Fitzer, A., Sturmey, P.: Autism spectrum disorders: applied behavior analysis, evidence, and practice. In: Ahearn, W.H., et al. (eds.) Behavior Analytic Teaching Procedures: Basic Principles, Empirically Derived Practices, pp. 31–72. Pro-Ed Inc., Austin (2007)
15. Fuks, H., Raposo, A., Gerosa, M.A., Pimentel, M., Lucena, C.J.P.: The 3C collaboration model. In: The Encyclopedia of E-Collaboration, pp. 637–644 (2007)
16. Grabowski, M., Roberts, K.H.: Risk mitigation in virtual organizations. J. Comput.-Mediat. Commun. (JCMC) **3**(4), 1–34 (1998)
17. Gutierrez, F., Baloian, N., Zurita, G.: Boosting participation in virtual communities. In: Vivacqua, A.S., Gutwin, C., Borges, M.R. (eds.) CRIWG 2011. LNCS, vol. 6969, pp. 14–29. Springer, Heidelberg (2011)
18. Hersey, P., Blanchard, K.H., Johnson, D.E.: Management of Organizational Behavior: Leading Human Resources. Prentice Hall, Upper Saddle River (2000)
19. Herzberg, F., Mausner, B., Snyderman, B.B.: The Motivation to Work. Wiley, New York (1959)
20. Jacob, S.M., Sam, H.S.: Analysis of interaction patterns and scaffolding practices in online discussion forums. In: Proceeding of the 4th International Conference on Distance Learning and Education (ICDLE). IEEE (2010)

21. Janzik, L., Herstatt, C.: Innovation communities: motivation and incentives for community members to contribute. In: Proceedings of the International Conference on Management of Innovation and Technology, vol. 4. IEEE (2008)
22. Johnston, K.A., Rosin, K.: Global virtual teams: how to manage them. In: Proceedings of the International Conference on Computer and Management (CAMAN), pp. 1–4 (2011)
23. Kaiser, P.R., Tullar, W.L., McKowen, D.: Student team projects by internet. Bus. Commun. Q. **63**, 75 (2000)
24. Karayaz, G.: Dealing with effectiveness on virtual team research. In: Proceedings of 25th Conference of American Society for Engineering Management (ASEM), pp. 242–247 (2004)
25. Kim, A.J.: Community Building on the Web: Secret Strategies for Successful Online Communities. Peachpit Press, Berkeley (2000)
26. Kraut, R.E., Resnick, P.: Encouraging contribution to online communities. In: Kraut, R.E., Resnick, P. (eds.) (Under Contract), Evidence-Based Social Design: Mining the Social Sciences to Build Successful Online Communities. MIT Press, Cambridge (2008)
27. Kuznetsov, S.: Motivations of contributors to Wikipedia. ACM SIGCAS Comput. Soc. **36**, 2 (2006)
28. Management study guide: motivation incentives – incentives to motivate employees (2012). http://www.managementstudyguide.com/motivation_incentives.htm
29. Piccoli, G., Powell, A., Ives, B.: Virtual teams: team control structure, work processes, and team effectiveness. Inf. Technol. People **17**(4), 359–379 (2004)
30. Powell, A., Piccoli, G., Ives, B.: Virtual teams: a review of current literature and directions for future research. ACM SIGMIS Database **35**(1), 6–36 (2004)
31. Preece, J.: Online Communities: Designing Usability, Supporting Sociability. Wiley, Chichester (2000)
32. Robert, L.P.: A multi-level analysis of the impact of shared leadership in diverse virtual teams. In: Proceedings of the Conference on Computer Supported Cooperative Work (CSCW 2013), pp. 363–374 (2013)
33. Robey, D., Khoo, H., Powers, C.: Situated learning in cross-functional virtual teams. IEEE Trans. Prof. Commun. **43**(1), 51–66 (2000)
34. Staples, D.S., Cameron, A.F.: Creating positive attitudes in virtual team members. In: Godar, S., Ferris, P. (eds.) Virtual & Collaborative Teams: Process, Technologies, & Practice, pp. 76–98. Idea Group Publishing, Hershey (2004)
35. Tedjamulia, S.J., Dean, D.L., Olsen, D.R., Albrecht, C.C.: Motivating content contributions to online communities: toward a more comprehensive theory. In: Proceedings of the 38th Annual Hawaii International Conference on System Sciences (2005)
36. The Incentive Research Foundation: Incentives, motivation and workplace performance: research & best practices (2002). http://www.loyaltyworks.com/incentive-program-research-articles/ispifullpdf.pdf
37. Werther, W.B., Davis, K.: Personnel Management and Human Resources. McGraw-Hill, New York (1981)
38. Wise, R.A.: Drive, incentive, and reinforcement: the antecedents and consequences of motivation. In: Dienstbier, R., et al. (eds.) Motivational Factors in the Etiology of Drug Abuse. The Nebraska Symposium on Motivation, vol. 50, pp. 159–195. University of Nebraska, Lincoln (2004)

Measuring Friendship Strength in Online Social Networks

Juliana de Melo Bezerra(✉), Gabriel Chagas Marques, and Celso Massaki Hirata

Computer Science Department, ITA, Sao Jose dos Campos, Brazil
{juliana,hirata}@ita.br, gabriel.chagas.marques@gmail.com

Abstract. Distinct friends can have different importance to an individual; and even the relevance of a given friendship can vary over time. To predict friendship strength can be useful to social, technical or commercial purposes. One interesting application of friendship strength is to help people to identify friends with whom they want to reestablish contact in order to keep a close relation. In this paper, we propose a strategy to define a tie strength metric to friendships in online social networks. We calculated the friendship metric in Facebook considering the public of young adults, and found 75 % of acceptance during experiments.

Keywords: Tie strength · Online friendship · Social network · Facebook

1 Introduction

The importance of social relationships is associated with individual physical and mental well-being, mainly due to the sense of being secure and supported [3]. The most frequent type of relationship is the friendship [1]. Not all friendships have the same meaning and impact to an individual. For instance, one can have close friends, casual friends or mere acquaintances. According to Granovetter [7], friendships with weak ties can help in generating ideas or finding jobs. Krackhardt [11] explains that friendships with strong ties can offer emotional support and trust in case of severe changes or uncertainty. Moreover, the salience of a friendship can vary over the life course [13].

In order to keep friendship closeness, maintenance strategies are essential [13]. Some examples include keeping in touch, offering emotional support, and participating in shared activities [4]. Communication technologies, like email and telephone, have an important role in friendship maintenance by providing easy and efficient means of interaction [17].

Online social networks provide an environment to rescue old friends and find new ones. They allow individuals to maintain friendships by using distinct mechanisms, such as exchange messages, and share comments, photos, and hobbies. To be able to predict tie strength in social media is a particular case of interest. Systems designers can use strength tie information to explore the link prediction problem [11], in order to study new associations between users or how such associations evolve. Tie strength can be useful to detect security frauds [14], to study answer quality for questions [15], and to improve privacy settings [12]. Besides, the knowledge about tie strength can have commercial impact. For example, products and services can be offered to individuals that trust in each other or have similar preferences, which may be common to close friends.

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 109–122, 2016.
DOI: 10.1007/978-3-319-30996-5_6

Our focus is friendship maintenance, so tie strength must reveal how a friendship is in a particular moment. The main motivation is to benefit users to keep friendship alive, a phenomenon similar to what occur in offline context [5]. So, a user, knowing about a weak friendship tie, can take action to reestablish contact with that friend.

In this paper, we propose a strategy to define a metric to quantify tie strength in online relationships. The metric is composed by variables related to existing features in a social network to support friendship maintenance. We use Analytic Hierarchy Process (AHP) as a decision method to find the relevance of the selected variables in the composition of the metric. We choose Facebook as the investigated social network, and find a friendship metric to the young adulthood. In order to evaluate the Facebook metric, we conduct experiments where users can search friends and assess information about tie strength.

The paper is organized as follow. In the next section, we explain the proposed approach to find a metric of tie strength in online relations. In Sect. 3, we develop a metric considering Facebook. In Sect. 4, we describe two experiments used to evaluate our proposal. In Sect. 5, we discuss benefits and limitations of our proposal. In Sect. 6, we compare our approach and results with related work. Conclusions and future work are presented in the last section.

2 Towards a Tie Strength Metric for Online Friendships

Given a relation between two friends in a social network, the metric M indicates how strong the tie strength is. We have that $0 \leq M \leq 1$. In terms of percentage, M varies from 0 to 100 %, where 0 % means no friendship maintenance, and 100 % indicates the existence of strong relationship maintenance. The proposed metric is basically the sum of variables v_i multiplied by their associated weights w_i, where $1 \leq i \leq N$ and N represents the quantity of variables considered to the metric, as follow:

$$M = \sum_{i=1}^{N} \left(CDF(v_i) * w_i \right) \tag{1}$$

Variables are the aspects in the social network that represent strategies of relationship maintenance. A variable can assume any value, for example, a relation can have 10 friends in common, while other relation has 100 friends in common. In order to be able to compare two variables, we use the cumulative distributed function (CDF) of each variable, so we have $0 \leq CDF(v_i) \leq 1$. Weights inform the relevance of the variables when composing the metric, so $0 \leq w_i \leq 1$ and $\sum w_i = 1$. We use multi-criteria decision analysis to find the weights.

2.1 Definition of Variables

We used the theory about strategies of relationship maintenance to support the identification of possible variables in the social network. Flanigan [5] and Metts et al. [13] provide useful discussion about different maintenance definitions and strategies that exist. They explain that friendships have distinct functions to individuals at different stages of life, and there are also differences in how individuals maintain friendships [13].

Dindia and Canary [4] use four strategies to define relational maintenance: continuity, stability, satisfaction, and repair. Individuals maintain their relation when they are continuing such relation and not terminating it. Stability refers to keep particular dimensions in a stable level, for instance, when individuals have interests or characteristics in common. The satisfaction concept explains how satisfied an individual is in keeping a relation. Repair is used to define relation maintenance as keeping a relationship in good condition by preventing decay.

Stafford and Canary [18] propose five relational maintenance strategies: positivity, openness, assurances, network, and sharing tasks. Positivity means being positive and enthusiastic about a relation. Openness is related to self-disclosure and being open to discuss a relation. Assurances include behaviors that show commitment to a relation. Networking means to have and keep friends in common. Sharing tasks is to share activities with your friend or to have activities in common.

Other theories exist to explain relationship, for example Granovetter [7] identifies four tie strength dimensions: time, intimacy, intensity, and reciprocal services. Time, for example, is an interesting aspect that can represent the amount of time spent together. Given the range of theories, there can have some overlap, for instance, positivity [18] can be understood as satisfaction [4], or continuity [4] can include time [7] aspects.

The initial set of variables to the metric can be found trough a brainstorming with users. Theories about relationship maintenance are useful to reason about maintenance strategies in online environments. For instance, 'time spent in chat together' is associated to satisfaction, whereas 'number of mutual friends' is related to network. Variables can even represent one or more dimension, for example, 'number of mutual friends' can be understood as both stability and network strategy.

Later, we focus on the five more relevant variables to measure the maintenance of a relationship. Thus we have number of variables N equals to 5. As AHP requires the pairwise analysis of variables to find variables' weights, we have the precaution to give to the next users a feasible evaluation to perform. It is difficult to estimate the limits of human information processing capacity. Halford et al. [8] made experiments breaking down problems into bite-size chunks to be solved by academics. The interactions among variables varied in complexity, considering two up to five variables. They found a significant decline in accuracy and speed of solution when problems got more complex. Performance on a five-way interaction was at chance level. They suggest that a structure defined on four variables is at the limit of human processing capacity. We decided to follow the directives of Halford et al. [8], and we select only five top variables to proceed with AHP.

The selection of the top five variables can be made by active users in the social network. By active users, we mean users that access the account at least one time a day and have more than 300 friends. The number of 300 friends is intentionally greater than the Dunbar's number. Dunbar's number (150) is an upper limit of relations that a person can maintain in offline social networks [9]. Relations exceeding the Dunbar's number are considered inactive or mere acquaintances. So, by selecting the active users, we wanted to get the perception from people that frequently use the social network and also may experience the problem of maintaining relations. Other important decision is to determine a period of time to consider time dependent variables.

Once we have identified the variables, we need to assign a standardized value within [0,1] to any given absolute number of each variable. We need to respect the following limits and constraints: (a) The maximum tie strength (equal 1) should be when all the absolute numbers assume its maximum value; (b) The minimum tie strength (equal 0) should be when all the absolute numbers are zero; (c) Any other combination of probabilities should generate a tie strength in [0,1]. The probability of random variable X being lower than a given absolute number x would fit it perfectly, considering zeros excluded. We then decided to base our standardization on the cumulative distribution function: $CDF(x) = \mathrm{P}(\mathrm{X} <= x)$. These directives are described in Eq. 2.

$$CDF(v_i) = \begin{cases} 0, & if\ v_i = 0 \\ CDFT(v_i), & otherwise \end{cases} \tag{2}$$

In order to build the *CDF* of each variable, we need to retrieve real data. We define $\max(v_i)$ as the highest value found to variable $\mathrm{v_i}$ in the collected data. Since, not all possible values between 1 and $\max(v_\mathrm{i})$ can be found in the *CDF* dataset, we decided to use trendlines (*CDFT*). We calculate the respective trendline of each *CDF* using polynomial approximation.

2.2 Definition of Weights

Analytic Hierarchy Process (AHP) is a multi-criteria decision analysis, which helps the analysis of complex problems [16]. Given a goal, possible alternatives, and established criteria, AHP provides numerical priorities to each alternative. Such priorities represent the ability of each alternative in achieving the goal. For example, the goal can be the purchase of a car; the alternatives are car A, car B, and car C; and the criteria can include aspects as price, quality and delivery date. Our objective is to determine the strength tie in a relation, so we would like to know the impact (weights) of variables (AHP alternatives) in the composition of the metric. AHP is then used to find the weights w_i of variables v_i, as stated in Eq. 1.

Here we do not detail the AHP calculations, but we present the main phases of the analysis:

(a) We define the goal of the problem, alternatives to reach the goal, and criteria to consider in the analysis.
(b) Decision makers indicate the relative significance of criteria, by comparing them in pairs. The objective is to find the decision matrix of criteria. We normalize the matrix and calculate the priority vector of criteria.
(c) Decision makers indicate the relative significance of alternatives, by comparing them in pairs considering each criterion separately. The objective is to find the decision matrix of alternatives to each criterion. It is necessary to normalize the matrix. Using it, we calculate the priority vector of alternatives given a criterion.
(d) Composing the priority vectors of alternatives in a matrix, and multiplying it to the priority vector of criteria, we find the priority of each alternative.

In the context of defining a friendship metric, decision makers are users of a given social networking. It is very subjective in a social network to define why individuals use some features or have some behavior, so we decided to have the users' perspectives as criteria and the social network variables as alternatives. A decision matrix of variables is created for each interviewed user and later normalized. Since we consider the judgment of each interviewed user equally important, the matrix of relative ranking of the criteria is filled with ones.

The comparison of alternatives in pairs is made based on a scale. So we elaborate a questionnaire to ask how important a variable is compared to others. As we have 5 variables, the questionnaire was composed by 10 questions in the form of "How important is v_i compared to v_j". We use the directive of Saaty scale to define answers' options. The answer could assume the following values: extreme importance (9), very strong importance (7), strong importance (5), moderate importance (3), equal importance (1), moderately less importance (1/3), strongly less important (1/5), very strongly less important (1/7), and extremely less important (1/9).

Each questionnaire is used to build a decision matrix of variables. The main diagonal is filled with one, meaning that one variable has the same importance when compared to itself. The questionnaire answers are the entries above the main diagonal, while their reciprocals are the entries below the main diagonal. As an example, a decision matrix of some existing questionnaire is shown in Table 1. The same procedure of building matrix is repeated to each questionnaire. Later we need to analyze the consistence of all matrixes, normalized them, and calculated the related priority vector. We then use these vectors to calculate the weights.

Table 1. Example of decision matrix of variables.

	v_1	v_2	v_3	v_4	v_5
v_1	1	1	1/7	1	1
v_2	1	1	1	1	1
v_3	7	1	1	1/3	3
v_4	1	1	3	1	1
v_5	1	1	1/3	1	1

3 The Friendship Strength Metric in Facebook

The usage of friendship maintenance strategies can vary according to individuals, for instance, in terms of age as young adults, middle age adults, and older adults [13]. Our work focuses on a specific public: the young adulthood. So, when there was a need to involve real users, we always selected different young adults, students of a college, with age varying from 18 to 26.

In the first brainstorming, we found 27 variables in Facebook, as follows: number of mutual friends (v_1), number of messages exchanged (v_2), number of pages in common that the friends liked (v_3), number of photos that the friends were tagged together (v_4), number of likes made in comments of a friend (v_5), if a friend is following the other, time online in common, number of apps in common, number of check-ins in common, age difference, number of events in common, number of groups in common, interests in common, family relationship, number of links liked in common, number of blocked pages in common, number of videos in common, work history in common, religion difference, politics difference, chat duration, chat frequency, event frequency, number of posts together, number of comments in common friend's posts, number of comments in common friend's photos, and number of comments in common friend's videos.

We submitted the list of variables, in a random order, to the appreciation of ten active Facebook users. One important decision was to determine a period of time as one month to consider time dependent variables. We then selected the following variables: number of mutual friends (v_1); number of messages exchanged in the last month (v_2); number of pages in common that the friends liked in the last month (v_3); number of photos that the friends were tagged together in the last month (v_4); and number of likes made in comments of a friend in the last month (v_5).

We invited ten Facebook users and, using an application, we collected v_i data of all their connections with friends. The quantity of connections assessed was 3855. From these relationships, we were able to collect 7244 nonzero data points that were used to plot the histograms. The histograms are shown in Figs. 1, 3, 5, 7, and 9. We also provide the CDF plots and *CDF* trendlines (*CDFT*) of all variables (Figs. 2, 4, 6, 8, and 10). The *CDF* trendlines (*CDFT*) are shown in Table 2. We checked if the polynomial approximation of each trendline was satisfactory by calculating the R-squared value. We found the following R-squared values from v_1 to v_5: 0.9909, 0.9505, 0.9947, 0.9961, and 0.9622. Our objective was to achieve R-squared value of at least 0.95 to each trendline, since a trendline is most reliable when its R-squared value is at or near 1.

Table 2. CDF trendlines of Facebook variables.

CDF trendline
$CDFT\left(v_1\right) = 4\mathrm{E}-08v_1^3 - 3\mathrm{E}-05v_1^2 + 0.0078v_1 + 0.1229$
$CDFT\left(v_2\right) = -0.0011v_2^4 + 0.0242v_2^3 - 0.1584v_2^2 + 0.3975v_2 - 0.213$
$CDFT\left(v_3\right) = 2\mathrm{E}-07v_3^5 - 2\mathrm{E}-05v_3^4 + 0.0008v_3^3 - 0.0182v_3^2 + 0.1915v_3 + 0.1632$
$CDFT\left(v_4\right) = -4\mathrm{E}-05v_4^4 + 0.0019v_4^3 - 0.0312v_4^2 + 0.226v_4 + 0.3581$
$CDFT\left(v_5\right) = 0.0001v_5^3 - 0.0057v_5^2 + 0.0916v_5 + 0.4847$

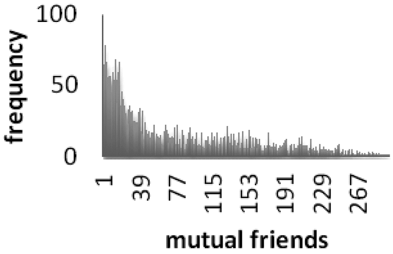


Fig. 1. Histogram of variable v_1.

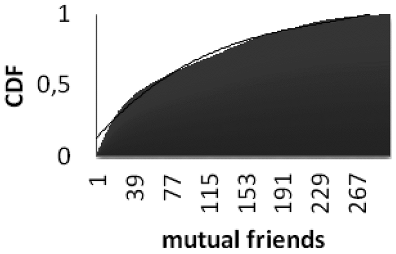


Fig. 2. CDF and trendline of variable v_1.

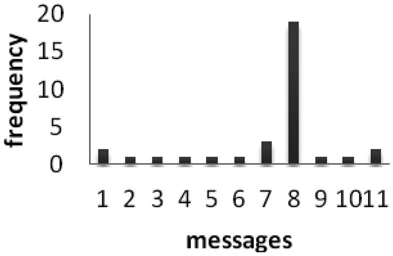


Fig. 3. Histogram of variable v_2.

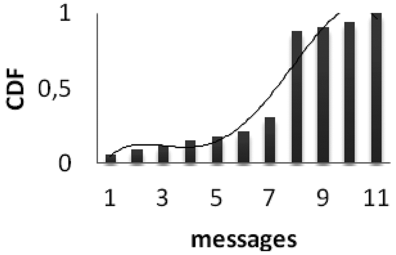


Fig. 4. CDF and trendline of variable v_2.

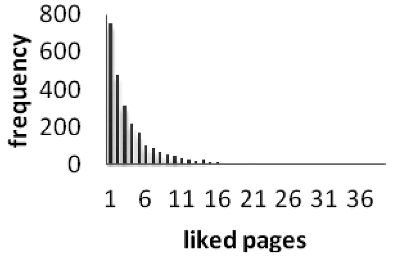


Fig. 5. Histogram of variable v_3.

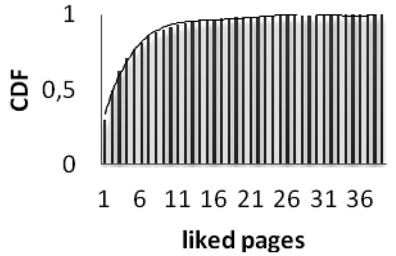


Fig. 6. CDF and trendline of variable v_3.

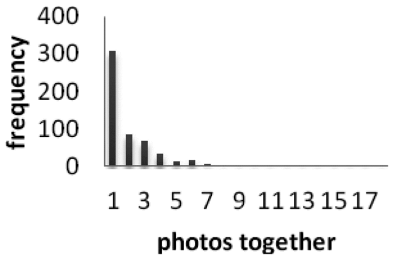


Fig. 7. Histogram of variable v_4.

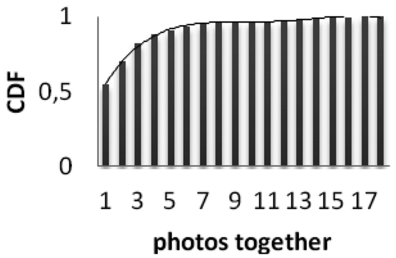


Fig. 8. CDF and trendline of variable v_4.

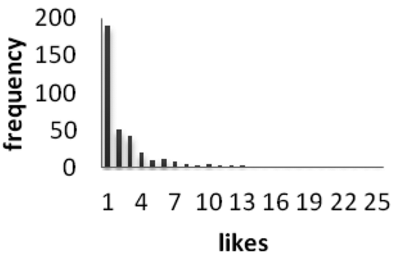


Fig. 9. Histogram of variable v_5.

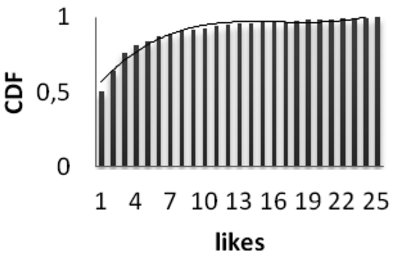


Fig. 10. CDF and trendline of variable v_5.

In order to define the weights of Facebook variables, we invited 30 Facebook users to act as decision makers individually. Following the directives in Sect. 2.2 to apply AHP, we found the weights shown in Table 3. We observe that ‘photos in common’ (v_4) is the most significant parameter of a friendship, since it means that friends were together. The next more important variable is ‘number of messages’ (v_2), meaning that friends are keeping in touch.

Table 3. Weights of Facebook variables.

Variable (v_i)	Weight (w_i)
v_1: number of mutual friends	0.121
v_2: number of messages exchanged in the last month	0.290
v_3: number of pages in common that the friends liked in the last month	0.105
v_4: number of photos that the friends were tagged together in the last month	0.332
v_5: number of likes made in comments of a friend in the last month	0.152

Using Eqs. 1 and 2 and the results shown in Tables 2 and 3, it is possible to calculate the tie strength of a Facebook friendship since you have the variables’ values that represent such relation.

4 Evaluating the Friendship Strength Metric in Facebook

We planned two experiments to evaluate the proposed metric already calibrated to young adults in Facebook, as described in the previous section. We invited 30 Facebook users, different from those that participated during the metric definition. In both experiments, applications were developed to shown information related to the metric. Applications to capture Facebook information have to deal with users’ permissions. So, before using the application, users had to accept that their private data would be used for the study. The applications also capture users’ answer of an evaluation question and their feedback to support the results’ analysis.

In the first experiment, we built an application where a user can search a friend and see the tie strength value of his friendship. In this situation, the user can analyze if the proposed metric is satisfactory. The user should evaluate the affirmative "I agree with the result" using the following 5-Likert scale: strongly agree, agree, neutral, disagree, and strongly disagree. The total of assessed relations was 96. The result was: 20 % strongly agree, 21 % agree, 56 % neutral, 0 % disagree, and 3 % strongly disagree. The metric was considered correct in 41 % of the cases. Neutral responses were, in general, users that could not judge if the metric value was adequate in absolute terms. We then conducted the second experiment to analyze the metric in relative terms.

In the second experiment, we developed other application, where a user can search two friends. The application calculates the metric of both relations, and it returns the name of the friend with higher metric. So, users can evaluate the metric in relative terms. Users answered the same question as in the first experiment. The amount of evaluated cases was 86. The result was: 35 % strongly agree, 40 % agree, 9 % neutral, 5 % disagree, and 12 % strongly disagree. The result was considered correct in 75 % of the cases. We observed that neutral responses dropped abruptly. According to users, it is easier to evaluate only the comparison instead of reasoning about the metric value itself.

Using users' feedback, we were able to understand the existence of disagreements with the result driven from the metric in the second experiment. One user said that one of the assessed friends was his brother in fact, and the result should have shown higher tie strength to his brother. We understand that the proposed metric correctly shows the maintenance level of the friendship and not the nature of that relation. Other user commented one case of disagreement, explaining that he always encounters his friend. It is a common misleading to evaluate a friendship using online tie when friends have strong offline interaction; however, we argue that the metric focuses on the strength of the online relation only.

A user reported that the application showed a higher tie with a friend, but he considers that both friends have the same importance, since the unique difference was to have only one more friend in common. It raises an interesting aspect about what we investigate in further comparative evaluations: the definition of a range to consider friendships as similar. Other case of disagreement was commented by a user who considers "likes" (v_5) are more important than "friends in common" (v_1). The metric uses the opposite, as show in Table 3. The metric was defined with solid foundation considering the opinion of distinct users. This user has a different impression, and we believe that it constitutes an outlier.

5 Discussions

The proposed metric of tie strength was proposed in a general way to fit friendships of any public in any social networking site. Different public can have different relational maintenance strategies; therefore they can use social network features in a different way. It can impact both trendline functions and weights. The five variables selected in the Facebook case are general and can be found in other social networks. The process using

AHP to define the friendship metric was described, and it can be repeated in further investigations that consider changes in social network or public.

Metric variables are time dependent. In the Facebook metric, variable v_1 regarding 'mutual friends' can change since individuals connect to others in a dynamic way by reconfiguring the network. The other variables (v_2 to v_4) have an explicit time range, in this case, a month. Time dependency is what makes the metric able to represent changes in relations. For example, an individual can interact more with a friend in a period, strengthening their relation. Later he can stay without contact, representing the absence of strategies to maintain the relation, so the tie strength reduces.

Different periods for data collection can be investigated to define the variables. We conjecture that long periods are not recommended since the metric may lose its momentum. Another issue is the effort to collect data. The bottleneck step is to capture the variables' value of a given relation. For instance, the variable v_4 about 'number of photos in common' requires an examination of each photo posted by a user. In this way, the application that uses the metric could not provide the metric value in a feasible time, which in turn generates usability issues.

We conducted a preliminary evaluation of the friendship metric using Facebook. A positive aspect was the online processing of relations' data. Two applications were built and used by real users, demonstrating the feasibility of the metric calculation. In the first experiment with absolute values of the metric, we found that it can be interesting to define labels to values, for instance 'low' and 'high'. It can facilitate users' judgment of the metric result. Other possibility is to remove 'neutral' option as answer, letting users to respond only positively or negatively, which is known as 'forced choice' method.

In the second experiment, we observed a major approval of the results, confirming that the metric was useful to compare tie strength of two friendships. According to users' feedback, one possible enhancement is the definition of a range to consider friends with the same importance. In both experiments, we argue that a higher testing sample could be beneficial to the evaluation. Other experiments can be designed considering friends not chosen by users, but friends selected randomly. Using this approach, we can even conduct the evaluation of friendships with high and low tie strength separately.

6 Related Work

Previous research has proposed different solutions to reason about tie strength in online social networks. Xiang et al. [19] propose a model to infer relationship strength from interaction activity (e.g., communication, tagging) and user similarity (e.g. common friends). Other important works are those proposed by Gilbert and Karahalios [6], Arnaboldi et al. [2], and Jones et al. [10]. Below we discuss these articles and compare their approach and results with ours.

Gilbert and Karahalios [6] investigate if social media data is able to predict tie strength of general relationships, in order to classify a relationship as weak or strong. They study the influence of the following dimensions (described here already in order of importance): intimacy, intensity, duration, social distance, services, emotional support, and structural. In order to build the prediction model, they considered data from the entire relationship, for example "wall words exchanged" variable counts every message since the relationship initiated in the social media. They used linear regression to determine the variables weights, and they added an extra term to the equation to take into account the network structure.

In our approach, we are interested in the online maintenance of relationships. We would like to know how a relation is in a given moment: if it is active or not. One possible benefit is to help people in keeping friendships alive. The background about relationship maintenance strategies helped us to identify potential variables to compose the tie strength metric. The maintenance strategies include, for instance, continuity, time, stability, and satisfaction. The maintenance strategies were used to investigate 27 Facebook features that are used to maintain friendships. The five selected variables are present in other social networks, which makes our approach feasible to be replicated in other environments. Gilbert and Karahalios [6] use variables as "wall intimacy words", which need content analysis, so that they focus on English language. For the Facebook metric, the selected variables do not rely on content analysis, so that it is possible to compare relations of a person with two friends using distinct idioms. While Gilbert and Karahalios [6] consider data during the entire relation, we specify a period of analysis. They retrieved all data and processed offline to calculate the dimensions' power. We did offline processing only to define our variables and weights, but later we use the metric in online experiments with real users. The experiments give confidence to the proposed metric, and they show that the metric was calculated in a feasible time: users selected friends, they waited the metric result and later they evaluated the result.

The focus of our paper is not just to find a metric to define tie strength, but also to provide an interpretation to how to maintain online friendships. Mathematically, linear regression and other types of regression make sense but they do not provide as much meaning to the equation they generate. Basically, the only meaning we can get from the equation is that it provides the best fit to the test data set. The key point is that regressions require the subjective evaluation on the result, i.e. users are asked to evaluate their relationship with other users and based on that the regression is calculated. On the other hand, our approach brings the subjective evaluation to the weights. The AHP allows us to bring meaning right away to what is important to users of social networks. If we find that $w_i = 3{*}w_j$, it literally means that most of people believe that v_i is more important to define an online friendship than v_j.

Arnaboldi et al. [2] use the same background as us, which includes the four tie strength dimensions (time, intimacy, intensity, and reciprocal services) proposed by Granovetter [7]. In fact, we complement our background with the dimensions suggested by Stafford and Canary [18] and Dindia and Canary [4]. Arnaboldi et al. [2] work with 11 quantitative relational variables. In the Facebook case, we initiate our investigation with 27 variables, which include the 11 variables used by Arnaboldi et al. [2], except from "number of days since first communication" and "number of days since last communication". Variables driven from user-filled fields (such as "educational difference") were not considered by Arnaboldi et al. [2] since the information depends on cultural aspects and can even not be provided by users. Instead of eliminate variables at the beginning, we decided to submit the 27 variables to the appreciation of users, in order to make them reason about the variables' importance. Five quantitative variables were selected, as follows: v_1 to v_5 (see Table 3). The variable v_1 was not used by Arnaboldi et al. [2]; however it is presented in other important works, such as Gilbert and Karahalios [6] and Xiang et al. [19].

Arnaboldi et al. [2] compare different models to predict tie strength, including models with uncorrelated variables and with correlated ones. They retrieved data from relations of 28 users, who also evaluated the strength of friendships (using a scale between 0 and 100). In our work, we asked users to evaluate the provided tie strength values in the first experiment, and we observed that it is a difficult task when using absolute values. So, we perform other experiment informing only the result of comparison between the tie strength values of two friends. Arnaboldi et al. [2] found a good performance of a 4-variables model, which includes: "number of days since last communication", "bidirectional frequency of contact", "number of days since first communication", and "frequency of incoming communication". It is interesting to observe that these variables are related to technological-mediated communication in general, and not exclusively to social networks. In our Facebook metric, variables v_1, v_3, v_4 and v_5 are typical of social networks. Arnaboldi et al. [2] reported an accuracy of approximately 80 %. It is close to our result of 75 %, although we are confident that this number can increase if more experiments are conducted. One similarity between our work and the one by Arnaboldi et al. [2] is that both respect the sociological background that considers tie strength as a linear combination of social factors. Other similarity is that the resulted models are composed by few variables, and consequently few data about relations, which make the more suitable to be used online in services and applications that explore tie strength prediction.

Jones et al. [10] investigate how to define if a Facebook user is a closest friend or a non-closest friend. They use classification methods, including logistic regression, SVM (support vector machines) and random forests. Our approach is different since we propose a tie strength metric, which makes possible to estimate the intensity of a relation between two users. Regarding the used variables, Jones et al. [10] selected variables among Facebook features, as they say, by hypothesizing those ones that would be diagnostic in categorizing dyads as closest-friends versus non-closest-friends. They consider both demographic variables (such as same gender and age difference) and interaction variables (such as comments, likes and photo tags). We rely on the background about relationships' maintenance strategies, whose dimensions lead us to 27 variables when considering Facebook case. Jones et al. [10] found that demographic variables contribute little to the prediction model, since the frequency of online interaction was diagnostic of strong ties. It corroborates with our approach, which has four interaction variables (v_2 to v_5 in Table 3). We also consider a network variable (v_1), which is cited as relevant in other works, such as Gilbert and Karahalios [6] and Xiang et al. [19]. Considering the results, Jones et al. [10] reported 82 % of accuracy when using logistic regression model to classify a friend as a closest-friend or not. We achieved 75 % of accuracy by comparing the tie strength values of two relations in an experiment with real users. As the works have different goals, it is not appropriate to compare these results directly, but it may give an indication about the works' potential.

7 Conclusions

We proposed a strategy to identify a metric to quantify the tie strength of friendships in an online social network. We worked with a public of young adults in Facebook, in order to identify a friendship metric considering such public and environment. Using the background about strategies of relational maintenance, we identified the metric variables, which are driven from features provided by the social network. The following variables were selected: mutual friends, exchanged messages, pages in common, photos together, and likes. The relative importance of variables in the metric composition was defined based on users' perspective retrieved using the Analytic Hierarchy Process (AHP).

Our preliminary evaluation showed that the metric was useful in detecting the closest friend when comparing the tie strength of two friendships in Facebook. Users considered that 75 % of the cases were satisfactory. It is an interesting result that shows the potential of the metric. Other evaluations need to be conducted with more users testing more relations, in a way to increase sample sizes. Other experiments can consider different ways to capture users' perception about the metric, for instance to present a label associated to the metric value, to select friends randomly, and to evaluate tie strength indirectly by testing friends' influence. New applications can also be designed to capture users' intention to rescue important friendships that are presenting weak ties.

Further investigations can be performed in Facebook with other public or even in other social networks. As the metric variables are related to features very common in social networks, they can be used without changes. Other variables can also be selected. Differences are expected to be presented in the data distribution as well as in the variables' weight, due to existence of distinct behaviors when changing people and environment.

References

1. Argyle, M.: The Psychology of Happiness. Routledge, London (1987)
2. Arnaboldi, V., Guazzini, A., Passarella, A.: Egocentric online social networks: analysis of key features and prediction of tie strength in Facebook. Comput. Commun. **36**(10–11), 1130–1144 (2013)
3. Baumeister, R.F., Leary, M.R.: The need to belong: desire for interpersonal attachments as a fundamental human motivation. Psychol. Bull. **117**(3), 497–529 (1995)
4. Dindia, K., Canary, D.J.: Definitions and theoretical perspectives on maintaining relationships. J. Soc. Pers. Relat. **10**, 163–173 (1993)
5. Flanigan, N.N.: Keeping friendships alive: self-monitoring and maintenance strategies. In: UNF Thesis and Dissertations, Paper 168 (2005)
6. Gilbert, E., Karahalios, K.: Predicting tie strength with social media. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2009), pp. 211–220 (2009)
7. Granovetter, M.S.: The strength of weak ties. Am. J. Sociol. **78**(6), 1360–1380 (1973)
8. Halford, G.S., Baker, R., McCredden, J.E., Bain, J.D.: How many variables can humans process? Psychol. Sci. **16**(1), 70–76 (2005)
9. Hill, R.A., Dunbar, R.I.M.: Social network size in humans. Hum. Nat. **14**, 53–72 (2003)
10. Jones, J.J., Settle, J.E., Bond, R.M., Fariss, C.J., Marlow, C., Fowler, J.H.: Inferring tie strength from online directed behavior. PLoS ONE **8**(1), e52168 (2003)

11. Krackhardt, D.: The strength of strong ties: the importance of philos in organizations. In: Nohria, N., Eccles, R. (eds.) Networks and Organizations: Structure, Form, and Action, pp. 216–239. Harvard Business School Press, Boston (1992)
12. Kauer, M.: Improving privacy settings for Facebook by using interpersonal distance as criterion. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2013), pp. 793–798 (2013)
13. Metts, S., Beverly, A., Asbury, B.: A comparison of maintenance strategies in the friendships of young adults, middle age adults, and older adults. In: Proceedings of The NCA 95th Annual Convention, Chicago Hilton & Towers, Chicago, IL (2009)
14. Neville, J. et al.: Using relational knowledge discovery to prevent securities fraud. In: Proceedings of the International Conference on Knowledge Discovery in Data Mining (KDD 2005), pp. 449–458 (2005)
15. Panovich, K. Miller, R., Karger, D.: Tie strength in question & answer on social network sites. In: Proceedings of the Conference on Computer Supported Cooperative Work (CSCW 2012), pp. 1057–1066 (2012)
16. Saaty, T.L.: Multicriteria Decision Making: The Analytic Hierarchy Process. RWS Publications, Pittsburgh (1991)
17. Shklovski, I., Kraut, R., Cummings, J.: Keeping in touch by technology: maintaining friendships after a residential move. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI 2008), pp. 807–816 (2008)
18. Stafford, L., Canary, D.J.: Maintenance strategies and romantic relationship type, gender, and relational characteristics. J. Soc. Pers. Relat. **8**, 217–242 (1991)
19. Xiang, R., Neville, J., Rogati, M.: Modeling relationship strength in online social networks. In: Proceeding of the International World Wide Web Conference (WWW 2010), pp. 981–990 (2010)

Enhancing the Modularity and Flexibility of Identity Management Architectures for National and Cross-Border eID Applications

Thomas Lenz(✉) and Bernd Zwattendorfer

e-Government Innovation Center, Inffeldgasse 16a, Graz, Austria
{thomas.lenz,bernd.zwattendorfer}@egiz.gv.at

Abstract. Identity-management systems play a key role in various areas of applications and e-Government processes where access to sensitive data needs to be protected. To protect this sensitive data, the identity-management system provides all necessary functionality to service providers to manage digital identities and to handle the identification and authentication process. Identity management per se is no new topic and hence several identity-management systems have evolved over time, which are deployed in almost all European countries. However, identity management is constantly evolving in terms of new technical or legal requirements, higher secure protocols, new identification and authentication mechanism, or new fields of applications. In particular, the need for exchanging or federating identities across domains or even borders requires new interoperable solutions and flexible identity management architectures. In this paper we present a flexible and modular identity management architecture which focuses on federation and interoperability capabilities based on plug-able components. Due to that, new arising requirements can be easily fulfilled by implementing appropriate plug-ins. Hence, our proposed architecture is especially applicable for high qualified identification systems such as national eIDs for e-Government applications and their federation across borders. We further illustrate the applicability of our architecture by implementing it to be used as an identity provider for Austrian eGovernment applications, on the one side being applicable for national authentications and, on the other side, in a cross-border context.

1 Introduction

Electronic identity (eID) is indispensable for a variety of Internet services and online applications. Once the identity of communication entities is established with a level of certainty matching the value associated with the service, the communication partners can gain the confidence and trust needed for mutual transactions. Such transactions can include social network interactions, but also more security-sensitive services such as a tax declaration or an eHealth application that protects personal medical data. In each case, besides using an electronic identity authentication is additionally required to prove a claimed identity to be

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 123–143, 2016.
DOI: 10.1007/978-3-319-30996-5_7

authentic. Consequently, this authentication step link the identity information to a person, which uses a PIN or password to proved that he or she is the owner of that identity information.

Actually, more and more transactions are preformed electronically, by using online applications ore Internet service, which processing sensitive data. Consequently, the importance for a high level of assurance by secure means of authentication linked to qualified identity is rising sharply. eGovernment is such an area, where high assurance in the citizen's identity is needed. Therefore, several countries have already developed and deployed electronic identity systems for the eGovernment infrastructure since the beginning of the 21st century. This deployed electronic identity systems are still in operation since more than a decade ago, it is not too hard to guess that requirements on identity management solutions have changed over time and new technologies have emerged. Such technologies and requirements are not only things like new authentication protocols, which support a higher level of security, or new identification and authentication mechanisms, but moreover are requirements targeting usability, interoperability, or identity-management federations.

Particularly, identity-management federations such as nationally federated eID solutions or even cross-boarder eID federations became more and more important in the last couple of years. In the case of cross-boarder eID, the European Commission has recently published the EU regulation on Internal Market electronic identification and trust services (eIDAS) [1], which builds the legal framework for cross-border eID acceptance within the EU. However, the eIDAS regulation is currently only the latest step towards the implementation of a pan-European eID federation. The aim on cross-border eID recognition dates already back to 2005, as the aim was mentioned in the Manchester Ministerial Declaration [2], followed by the EU Service Directive [3] in 2006 and the eID large scale pilot projects STORK[1] and STORK 2.0[2], which is still running.

Since national eID systems have been deployed nearly a decade, many requirements has been changed. In order to meet these new or changed requirements on e.g. cross-border federation, an improved and enhanced architecture for identity-management systems is inevitable for meeting those requirements. Therefore, we present an improved identity-management architecture in this paper, which will meet current requirements and which is open to future extensions.

This paper is structured as follows. In Sect. 2 general requirements for identity-management solutions are defined. In Sect. 3, we describe related work and discuss it with respect to the defined requirements of Sect. 2. In Sect. 4, we propose an enhanced architecture of an identity-management system which is capable of meeting all the requirements. Afterwards, in Sect. 5 we demonstrate the practical applicability of our proposed identity-management architecture by implementing an identity provider for Austrian eGovernment applications supporting three main identity-management use cases. Finally, conclusions are drawn in Sect. 6.

[1] https://www.eid-stork.eu/.

[2] https://www.eid-stork2.eu/.

2 Requirements

Identification and authentication are by far no new issues, thus several different identity-management systems have already evolved [4]. In most of these identity-management systems, user identification and authentication are handled by an identity provider, which finally transfers the user information and authentication data to the service provider. Based on these information and data, the service provider is able to decide whether to grant or deny access to its protected resources. Consequently, the identity provider constitutes a very important entity within an identity-management system. Especially if the service provider is a public-sector application providing eHealth or eGovernment services, the support of qualified citizen identification and secure authentication by the identity provider is essential. Hence, such an identity provider needs to fulfill certain requirements to meet the high level of assurance and security required by public-sector applications. For that reason, the following requirements should be fulfilled and kept in mind, if an identity provider for public-sector applications is designed.

- **Security:** An identity provider for public-sector applications is typically used in a security-sensitive area, which handles with highly personal date, like medical information. Public-sector applications require a highly secure identification and authentication process to protect these confidential and sensitive data against unauthorized access. Furthermore, a public-sector identity provider needs to be resistant against attacks that threaten to illegally influence the identification or authentication result.
- **Reliability and Testability:** Service providers that make use of the identity provider must be able to rely on the results of the identification and authentication processes carried out by the identity provider. In addition, it should be possible for the service provider to test and validate the authentication information to check if the information was provided from a trusted identity provider and not from a attacker.
- **Flexibility:** From a service provider's point of view, an identity provider should be able to provide different standardized interfaces for service-provider communication, to offer a wide range of possible connection scenarios. Therefore, flexibility with respect to service providers can reduce the deployment costs for them. From a citizen's point of view, an identity provider should provide different identification and authentication methods, in order to being able to support a large number of users and to enable a simple usage of different secure tokens.
- **Interoperability:** An identity provider should be work interoperable with other architectures, e.g. if the communication with other identity-management systems is necessary. The requirement of interoperability increases because the interconnection of heterogeneous identity management systems is important for identity federation. Especially, this requirement is important for cross-border acceptance of identity-management solutions and to interconnect national eID systems, like a pan-European eID federation.

- **Adaptability:** In many countries national legal requirements or eID solutions serving domestic needs exist, which an identity provider has to comply with. Such solutions – which cannot be implemented by generic standards – could be a special secure token or a proprietary national infrastructure. Therefore, an identity provider supporting public-sector applications needs to build on an adaptable framework to fulfill national characteristics and to support proprietary protocols or architectures.
- **Easy-to-Use Technology:** The usage of an secure identification and authentication process should not impede usability and accessibility for both citizens and service providers. Therefore, an identity provider should provide a recognizable user interface and enable a safe and known usage with this security-relevant application. Furthermore, this requirement covers several more aspects such as hiding complexity for service providers or platform independence to reduce deployment costs.
- **Modularity:** An identity provider should have a modular architecture, because modularity is in line with flexibility and interoperability. Therefore, a modular architecture facilitates the implementation of new functionalities to meet new requirements with respect to interoperability, standardized interfaces, or new identification or authentication methods.

There exists some other works, which handles with requirements for identity management systems [5,6]. Therefore, we use requirements of this related work in combination with our own experience to defined a non-exhaustive enumeration of requirements. This defined requirements are rather generic to be not bound to a special national identity-management system. In the next section, available identity-management systems are surveyed and their capabilities to meet the above defined requirements are assessed.

3 Related Work

Numerous identity-management initiatives and systems exist, therefore we will briefly introduce a couple of systems that gained importance either due to their broad use, or as they established relevant standards.

First systems used simple directory based solutions, like LDAP (Lightweight Directory Access Protocol), to perform identity management for single organisations. Since the borders between organisations decrease, interoperable identity-management becomes more and more important. In order to manage this, identity management has to be dynamic and adaptable in different and more complex situations to handle more then one specific context. This resulted in more adaptable solutions, like Kerberos [7], which is one of the earliest systems that allows secure authentication in unsecure TCP/IP networks.

With the increasing popularity of the World Wide Web, more sophisticated identity-management solutions, which allow secure authentication on application level, became popular. Therefore, within the Web new identity-management systems emerged, such as Shibboleth[3] or the Kantara initiative[4] (formerly the Liberty Alliance Project). Both projects influenced the development of the current version of the Security Assertion Markup Language (SAML 2.0) [8]. SAML has been developed by OASIS and defines one of the most important standards dealing with Single Sign-On or identity federation. A similar framework constitutes WS-Federation [9], being part of the WS-Security [10] framework. Another decentralized authentication system on the Web defines OpenID[5].

All above mentioned identity-management solutions could be used to perform a secure identification and authentication process, but most of them are limited to a single or few authentication protocols or standardized interfaces, which are used for service provider communication. Another issue is that they may not meet national legal requirements for qualified identification or authentication in security-sensitive areas of application as those identity-management systems have been designed generic. For instance, several countries use proprietary protocols or special eID infrastructures, like electronic mandate services for example, which can be used to add additional information to an authentication process. Furthermore, interoperability and federation with other eID solutions gains importance. While some of the previously described identity-management systems support federation, this is only possible when interconnecting systems with the same basic architecture or underlying protocol. However, currently used national eID systems have a heterogeneous structure, which means that different communication and variegated implementations are in use which hinder interoperability and identity federation.

In summary, there is currently no perfect solution available, which directly is able to fulfill all requirements stated in Sect. 2. To overcome this problem, we propose an enhanced and flexible architecture for identity-management systems using the example of an Austrian identity provider. The architectural design of the proposed solution is presented in the next section.

4 Architectural Design

The proposed solution of an advanced identity provider is based on a sophisticated modular architecture to satisfy the identified requirements. Figure 1 illustrates our proposal for a modular and adaptable architecture for an Austrian identity provider, which can be used in various ways for identification and authentication purposes. In case of an Austrian identity provider, our architectural solution facilitate authentication for public and private sector applications, electronic mandate services or cross-border authentication. Consequently, our architectural solution could not only be used in national eGovernment applications (public sector), but also are for a highly secure authentication on commercial applications (private sector), like a social network or an online shop, or to identity and authenticate foreign citizens.

[3] http://shibboleth.net/.

[4] http://kantarainitiative.org/.

[5] http://openid.net/.

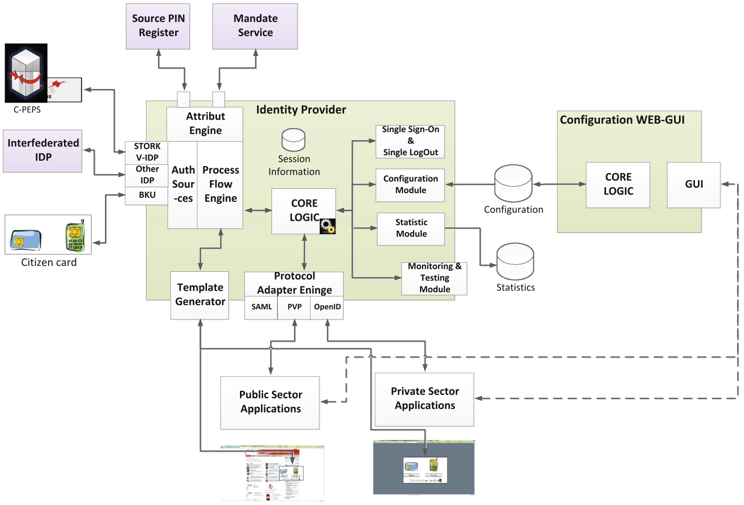


Fig. 1. Enhanced architecture of the Austrian public sector identity provider.

The *Core Logic* is the main item of our proposed solution. This main item coordinates the different steps of an identification and authentication process and handles the communication and interaction between all other modules and plug-ins, which can be used in our architectural design. This functionality is crucial, because an identification and authentication process mostly consists of different steps in which every step has a specific well-defined function. Consequently, it is imported to manage the divided different steps in a correct way, to fulfil the requirement of high secure identification and authentication of citizens. Therefore, our proposed architecture offer different features to support this different steps if an identification and authentication process in a modular way. To better illustrate this modular solution, we will describe it using the example of a generic identification and authentication process. This generic identification and authentication process describes the components and modules of our proposed solution on architecture level, but does not include every single communication step between the user's browser and the identity provider, service provider or other involved entities.

The authentication process is the first phase, if an citizen should be identified and authenticated. This phase is initiated by a communication between the Internet service and the identity provider via a well defined authentication protocol. In our architectural design, the *Protocol Adapter Engine* accomplish this communication task. To fulfil the requirements of flexibility and interoperability, we use a plug-in based to add and remove authentication protocols. For each supported authentication protocol, an appropriate *Protocol Plug-in* can be implemented. This modular Protocol Plug-in approach allows the implementation and usage of different protocols which are concerted to every single application with respect to protocol security and the required scope of operation. Such protocols could be SAML 2.0, which is widely in use, OpenID Connect [11], SAML 1.1 [12] or a national protocol, like the Austrian PVP 2.1 protocol [13], for example.

Our proposed architectural design allows the provision of different identification and authentication methods for users. According to this, a user could select the authentication method, which he or she wants to use, if the identity provider supports more the one identification and authentication solutions in the second phase. This step is carried out by a *Template Generator*, which generates a specific HTML Web interface providing a appropriate user interface to the user. Every Web interface is generated dynamically depending on all actually supported identification and authentication methods, which are implemented as Authentication Plug-ins, and application-specific information. This dynamically generated Web interface satisfies the requirement of an Easy-to-Use technology, because it provides a uniform interface to enable a safe and known usage of this security-relevant process step.

The third phase performs the technical identification and authentication operations. Our solution supports different high secure identification and authentication methods, which are collected and handled by an *Authentication Source Engine*. An identification or authentication step is realized as a single Plug-in. Such Plug-ins implement the communication with a secure token, like a smartcard, a hardware security-module (HSM), or the communication with another identity-management system, by using a well-defined interface, like STORK for example. A *Process Flow Engine* combines the single Plug-ins and these functionality to a well defined identification and authentication process flow. Every process flow, which is offered by the Process Flow Engine, is specified in a XML based configuration file by using an expression language. This expression language can be used to define single identification or authentication task, transactions between single tasks, and conditions for every transaction.

An additional *Attribute Engine* can be used in a fourth phase. This Attribute Engine manages *Attribute Provider Plug-ins*, which can be used to collect additional authentication attributes. Such attributes could be an electronic mandate in case of an authentication on behalf of somebody or other information collected from a national register, like the Austrian *Source-Pin Register* or the Austrian electronic *Mandate Service*. As example, the *Source-Pin Register*could be used to receive an additional unique identifier for this user an the electronic *Mandate Service* could be used to append mandate functionality to an eGovernment process.

At last, the collected identification and authentication information are processed to generate an authentication protocol specific authentication token, which is transmitted to the application by using a Protocol Plug-in. This modular approach allows the definition of various slightly different identification and authentication processes which satisfy the requirement of every application.

An additional feature of our architecture is a generic interface, which can be used to add new functionality to the Core Logic. The generic interface also uses Plug-ins to add new features to the core functionality. Such Plug-ins could implement features like Single Sign-On methods, monitoring and testing functionality, or a plug-in, which collects anonymised statistics information for quality assurance. To fulfil the requirement of an Easy-to-Use technology, a Web based management application, which provides a graphical interface to application administrators, can be used to configure the identity provider.

5 Implementation

The practical applicability of the proposed architectural design has been evaluated by realizing and implementing an identity provider in practice. To illustrate that, we have implemented an identity provider for Austrian eGovernment applications. Our implementation is based on Java, thus achieving platform independence and an easy deployment on heterogeneous server infrastructures. The next sub-sections discuss three practical use cases and their implementation by using our architecture in more detail.

5.1 Use Case 1: Austrian Citizen Authenticating at an Austrian Service Provider

In Austria, unique citizen identification and secure authentication is based on the technology-neutral concept of the Austria citizen card [14]. Currently, the Austrian citizen card is implemented as a client-side approach using smart cards and as a server-side approach involving the citizen's cell phone. Unique identification of a citizen is done by using a special XML data structure which is stored on the citizen card. Authentication is based by the creation of a qualified electronic signature. Since the Austrian citizen card is the official eID in Austria, a basic functional requirement of an Austrian identity provider is the support of the Austrian citizen card. Figure 2 illustrates the involved entities and their interactions in case of an identification and authentication process.

According to Fig. 2, the process of identification and authentication involves the following steps:

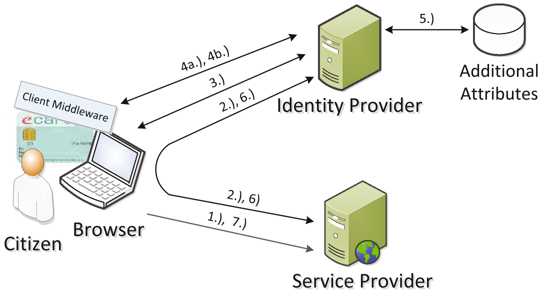


Fig. 2. Involved entities in an identification and authentication process in Austria.

1. A citizen wants to access a protected area on a service provider, which could be a eMail application, but also are a more sensitive application like an eHealth application, by using a HTTP GET or HTTP POST request. This protected area requires the identification and authentication of the citizen, by using the Austrian citizen card.
2. Therefore, the service provider starts an authentication process by triggering the identity provider. The identity provider is triggered by the service provider, which sending an authentication request via a specified authentication protocol. Most authentication protocols use HTTP POST or an HTTP Redirect with GET parameters to send an authentication request from service provider to identity provider via the user's browser. To fulfill the requirements of flexibility and interoperability and to support service providers, which use a diversified set of software implementations, our practical solution implements different authentication protocol plug-ins and hence is able to receive authentication request using different protocol formats. Actually, we implemented four protocol plug-ins to support SAML 2.0, OpenID Connect [11], the Austrian-specific PVP 2.1 S-Profile [13] and SAML 1.0[6] [12]
3. After the authentication request has been processed by the identity provider, the identity provider asks the citizen to select her preferred authentication method. Therefore, the Template Generator module, which is part of the identity provider, generates a web form to illustrate the different authentication solutions, which are supported by the identity provider. For Austria as example, a smart card based solution and a mobile phone based solution exists. After the citizen has selected the preferred solution, the identification and authentication process is started.
4. The proper identification and authentication process is performed by the Process Flow Engine in combination with the Authentication plug-ins. We have implemented different Authentication plug-ins to realize different processes for citizen identification and authentication. In the following two sub-steps, we describe the process, which uses the Austrian citizen card for this purpose, as an example. Therefore, a client middleware, which is just a piece of software (either installed on the citizen's PC or hosted on a server), facilitating access to the underlying citizen card implementation. In this example, a server hosted solution is used to deploy a JAVA Applet based client middleware in the citizens browser [15].
   (a) First, the identity provider identifies the citizen by using the XML data structure from the citizen card through the client middleware. This corresponds to the identification step. The corresponding plug-in implements the communication with the middleware and verification of the XML data structure, which comprises citizen identification information.
   (b) Second, the identity provider requests the citizen, via the client middleware, to create a qualified electronic signature for authentication. This task is also realized as a plug-in which implements the task specific communication and validation operations. Especially, validation is important

[6] In Austria, SAML 1.0 is widely used as legacy protocol by existing service providers.

to comply with the high security requirements for eGovnernment applications. Therefore, the electronic signature must be verified by the plug-in involving appropriate certificate revocation mechanisms, for example.

5. After identification and authentication are completed, the identity provider could use the Attribute Engine to collect additional authentication information of the citizen. Such additional information could be electronic mandates, for example, which are often used in Austria [16]. In our architecture, such additional information can be easily added to the authentication process by realization of an Attribute Engine plug-in. Therefore, we implement the communication with the Austrian electronic mandate service by using the Attribute Engine functionality.
6. If all authentication information is collected properly, the identity provider generates a protocol specific data structure. This data structure includes all authentication information that the service provider has requested and is transferred to the service provider.
7. Based on the received authentication information, the service provider is able to provide the protected resource to the citizen.

5.2 Use Case 2: Identity Federation

This scenario covers the case, where authentication information should be transferred from one identity provider to another identity provider. Such functionality brings considerable advantages to heterogeneous service models, in which service providers are linked to differed identity providers. Such advantages, for example are federated single sign-on (SSO) or interaction of identity providers which implements different identification and authentication methods. Figure 3 illustrates the actors and their relations in a federated service model. In this use case, every service provider is registered at a specific identity provider, similar to Use Case 1 described in Sect. 5.1, but in this use case there is the possibility of an

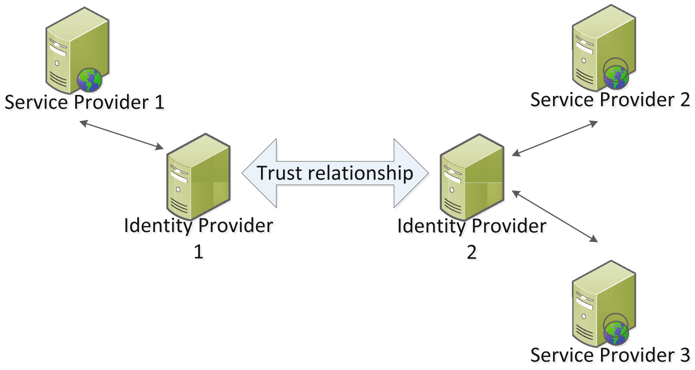


Fig. 3. Overview-Identity federation.

authentication data transfer between the individual identity providers. To transfer the authentication data between the concerned identity providers, a secure and trusted communication channel has to be established. We use the SAML 2.0 protocol to establish a trustworthy communication channel by using the SAML2 WebSSO Profile [17] and an exchange of SAML2 metadata [18]. An advantage of this solution is a high interoperability with other identity-management systems or identity provider implementations, because SAML2 is supported by almost all identity management solutions.

This functionality brings a lot of advantages for citizens and service providers. In the Austrian eGovernment, there actually exists practical applications for such an identity federation. We will present two of these applications next, one for citizens and one for employees of a public authority.

Federated Single Sign-on (SSO). eGovernment applications in Austria use a decentralized identity management approach, which means that service providers deploy there own identity provider for authentication locally in their service provider domain. This decentralized approach has advantages in case of availability and scalability but it is difficult to provide modern user-friendly functionality, like single sign-on for example. To overcome this disadvantage, we implement a single sign-on federation mechanism. Figure 4 illustrates such a federated single sign-on application scenario and the involved stockholders graphically.

The main stockholders in this application scenario are an *eGovernment Service Portal* with its dedicated *Identity Provider Service Portal* and an *eHealth Service* with is dedicated *Identity Provider eHealth*. According to Fig. 4, a citizen authenticates a single sign-on session on an *eGovernment Service Portal* by using the *Identity Provider Service Portal* to perform the identification and authentication process. Consequently, the single sign-on session is linked to the browser session between the *Identity Provider Service Portal* and the Web browser, which is used by the citizen. After this first identification and authentication, the citizen is authorized to enter the secure area of the *eGovernment Service Portal*. This secure

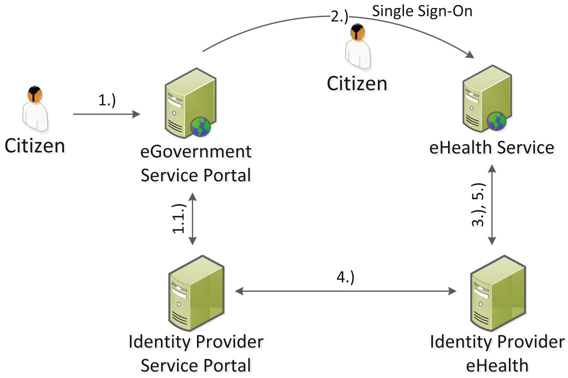

2.)
Single Sign-On
Citizen
1.)
Citizen
eGovernment Service Portal
eHealth Service
1.1.)
3.), 5.)
4.)
Identity Provider Service Portal
Identity Provider eHealth

Fig. 4. Federated Single Sign-On for citizens.

area of the *eGovernment Service Portal* could be a One-Stop-Shop for different other eGovernmant applications, like an *eHealth Service* for example.

After this, the citizen wants to use an *eHealth Service*, which operates as a self-contained web application. For this purpose, the citizen clicks a link in the *eGovernment Service Portal* which redirects to citizen to the *eHealth Service* and automatically starts an identification and authentication process for them. But in contrast to a traditional process, in which a full identification and authentication process similar to the process described in Sect. 5.1 must be performed on the *Identity Provider eHealth*, our solution could reuse the existing single sign-on session at the *Identity Provider Service Portal* to authenticate a transaction at the *Identity Provider eHealth*. Figure 5 illustrates the full sequence diagram of this identification and authentication information transfer between the two identity providers.

According to Fig. 5, the federated single sign-on identification and authentication process involves the following steps. This sequence description starts with step 2, shown in Fig. 4, in which the citizen is already authenticated at the *eGovernment Service Portal* and now wants to use an *eHealth Service*.

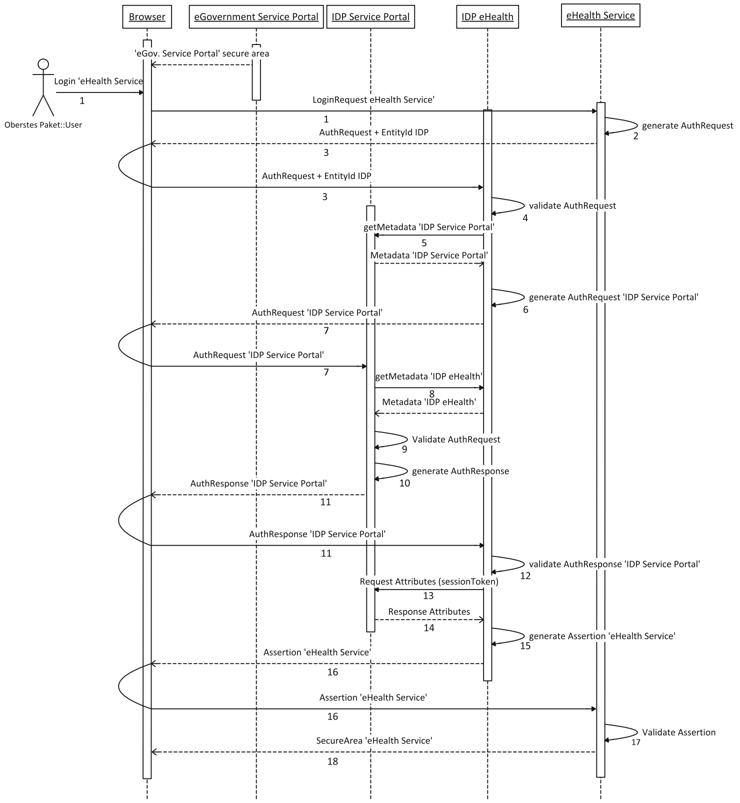

Browser
eGovernment Service Portal
IDP Service Portal
IDP eHealth
eHealth Service
'eGov. Service Portal' secure area
Login 'eHealth Service
1
Oberstes Paket::User
LoginRequest eHealth Service'
1
generate AuthRequest
2
AuthRequest + EntityId IDP
3
AuthRequest + EntityId IDP
3
validate AuthRequest
4
getMetadata 'IDP Service Portal'
5
Metadata 'IDP Service Portal'
generate AuthRequest 'IDP Service Portal'
6
AuthRequest 'IDP Service Portal'
7
AuthRequest 'IDP Service Portal'
7
getMetadata 'IDP eHealth'
8
Metadata 'IDP eHealth'
Validate AuthRequest
9
generate AuthResponse
10
AuthResponse 'IDP Service Portal'
11
AuthResponse 'IDP Service Portal'
11
validate AuthResponse 'IDP Service Portal'
12
Request Attributes (sessionToken)
13
Response Attributes
14
generate Assertion 'eHealth Service'
15
Assertion 'eHealth Service'
16
Assertion 'eHealth Service'
16
Validate Assertion
17
SecureArea 'eHealth Service'
18

Fig. 5. Sequence diagram of federated Single Sign-On.

1. The citizen wants to use an *eHealth Service*, which operates as a self-contained web application. For this purpose, the citizen clicks a link in the service portal, which starts an identification and authentication process on the *eHealth Service*. This link URL includes the information of a possible active single sign-on session on the service portal IDP as a HTTP GET parameter. This HTTP GET parameter contains a unique identifier of the service provider IDP. In our implementation, we use the SAML2 EntityID of the *Service Portal Identity Provider* as unique identifier.
2. The *eHealth Service* generates an authentication request, by using one of the authentication protocols which the eHealth identity provider offers. By using our implemented solution, the *eHealth Service* could use SAML1, PVP 2.1 or OpenID Connect as authentication protocol.
3. The *eHealth Service* requests authentication from its dedicated identity provider, but in contrast to Use Case Sect. 5.1 the information of an active SSO session at the *Service Portal Identity Provider* is provided, by using the SAML2 EntityID. We use the SAML2 EntityID of the Service Portal Identity Provider, because the EntityID could be easily used by the *eHealth Identity Provider* to determine all necessary information to communicate with the*Service Portal Identity Provider*.
4. The *eHealth Identity Provider* validates the authentication request from the *eHealth Service*. If the request is valid, the federated single sign-on process starts. This federated authentication process can be divided into several steps. In all steps, the SAML2 WebSSO Profile is used to transfer authentication data between the identity providers in an encrypted way. The encryption keys are shared by using the information in SAML2 metadata, which are provided from each IDP.
5. The *eHealth Identity Provider* use the SAML 2 EntityID, received from the *eHealth Service*, to load the SAML2 metadata from the *Identity Provider Service Portal*. Therefore, the SAML2 Well Known Location Method [18] is used to evaluate the SAML2 Metadata URL.
6. The *eHealth Identity Provider* use the information from the SAML2 metadata to generate a SAML2 authentication request for the *Identity Provider Service Portal*.
7. The *eHealth Identity Provider* sends the authentication request to the *Service Portal Identity Provider* by using SAML2 Redirect Binding [19]. The Redirect Binding endpoint URL is also automatically discovered from the *Service Portal Identity Provider* metadata information.
8. The *Service Portal Identity Provider* use the SAML2 EntityID, which is part of the authentication request to get the SAML2 metadata from the *Identity Provider eHealth*, by using the SAML2 Well Known Location Method.
9. The *Service Portal Identity Provider* validates the SAML2 AuthnRequest, by using the SAML2 metadata which was received one step before. If the authentication request is valid, the federated authentication process is continued.
10. The *Service Portal Identity Provider* checks if a valid single sign-on session exists for this citizen. If the single sign-on session is valid, the *Service Portal*

Identity Provider create a SAML2 Assertion, which should be returned to the *eHealth Identity Provider*. This SAML2 Assertion is encrypted, by using the encryption key from the SAML2 metadata and only contains a unique identifier for the citizen, which should be identified and authenticated.

11. The *Service Portal Identity Provider* sends the SAML2 Assertion to the eHealth identity provider by using SAML2 Redirect Binding [19]. The Redirect Binding endpoint URL is also automatically discovered from the *Identity Provider eHealth* metadata information.
12. The *eHealth Identity Provider* validates the SAML2 Assertion received from the *Service Portal Identity Provider*. If the *eHealth Service* requires more attributes as the unique identifier of the user, the *eHealth Identity Provider* could use a SAML2 AttributeQuery request to collect more detail information from the *Service Portal Identity Provider*.
13. Therefore, the *eHealth Identity Provider* creates an SAML2 AttributeQuery request, which contains the unique identifier of the citizen and all attributes which are required. After this, the SAML2 SOAP Binding [19] protocol to build up a direct communication channel between the *eHealth Identity Provider* and the *Service Portal Identity Provider*. This communication channel is used to request all additional attributes, which are necessary to identify and authenticate the user at the *eHealth Portal*.
14. The *eHealth Identity Provider* receives all requested identification and authentication information from the *Service Portal Identity Provider*
15. If all attributes are collected, the eHealth identity provider could generate an *eHealth Service* specific authentication protocol response.
16. This *eHealth Service* specific authentication protocol response is returned to the *eHealth Service* by using a authentication protocol specific communication binding.
17. At last, the *eHealth Service* uses this authentication response to authenticate the citizen and grant access to the secure area.

By using our federated solution, it is possible to combine the user-friendliness of single sign-on solutions with the availability of decentralized services. Additionally, this solution requires no service-provider modifications because all functionality can be implemented on identity provider side.

Public-Authority Network Gateway. eGovernment services are not only used by citizens, they are also used by public officials during there occupation in public administrations. Such public administrations are carried out from a private government network on public eGovernment services. However, such administrative operations often require extensive privileges or additional attributes for security reasons. Figure 6 shows this use case in a graphical example. In this example, a public official would use an eHealth Service as part of his work as a civil servant. Here, the public official could be identified and authenticated in the secure private network area and maybe some additional information attributes could be collected. After this, he could be authenticated as a civil servant at the eHealth service without full re-authentication on the eHealth identity provider,

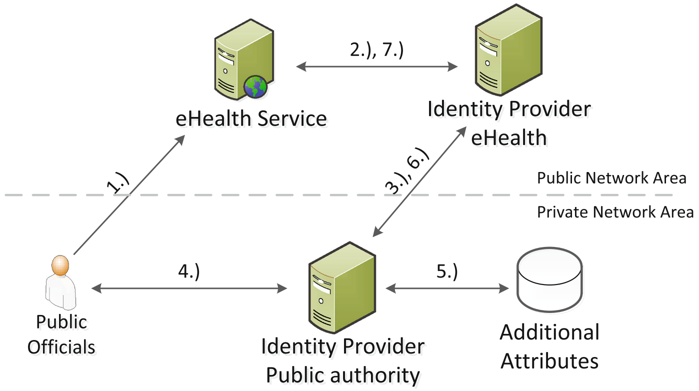


Fig. 6. Authentication of public officials on public eGovernment applications.

by using identity federation. An advantage of this solution is that there is no adjustment at the eHealth service necessary because the functionality for public officials is encapsulated in the identity provider functionality and can be also used for other services providers.

Both application scenarios can be implemented easily by using our architectural design and actually there is a trial period for establishment in Austrian eGovernement applications.

5.3 Use Case 3: European Citizen with European Service Provider

The third use case tackles the requirement of a secure and seamless cross-border electronic identification, which is part of the European eIDAS regulation or the STORK 2.0 large scale pilot [20]. Due the mobility of citizens, cross-border interoperability of national electronic identity systems in the European eID landscape has become more and more important in the last couple of years. Actually, every EU member state has implemented its own identity management service infrastructure. This circumstance leads to a heterogeneous environment when these individual solutions should be coupled to a cross-border electronic identification solution. The STORK large scale pilots treated with an interoperability framework, which can be used to couple different national eID solutions.

The STORK interoperability framework defines two different models, which can be used to build up an interoperability layer between national eID solutions. These models are the Pan European Proxy Service (PEPS) model, which is shown in Fig. 7 and the middleware (MW) model illustrated in Fig. 8 [21].

The PEPS model uses a proxy-based approach to encapsulate specifics of the national eID infrastructure. In this model, a PEPS is a national gateway and a single point of service for other countries, which implements the cross-border authentication functionality. In contrast to the PEPS model, in the middleware model citizens are directly authenticated at the service provider. Therefore, the service provider has to deploy a so-called V-IDP in the service provider

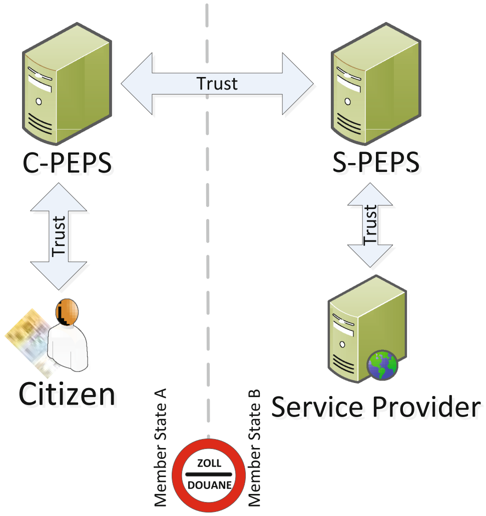


Fig. 7. STORK interoperability framework-PEPS model.

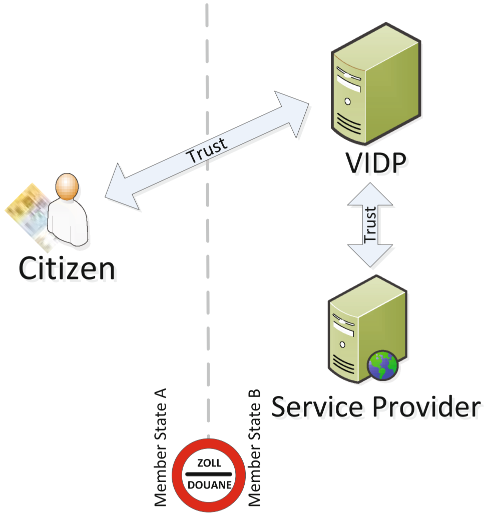


Fig. 8. STORK interoperability framework-Middleware model.

infrastructure. This V-IDP is the server-side middleware, which provides all necessary functionality for citizen identification and authentication. Actually, STORK implements both models and all possible combinations between them because there are advantages and drawbacks in both interoperability framework models. [21].

Therefore, we implement a solution for our Austrian identity provider, which can be used in both models in order to enable the widest possible utilisation.

From a national point of view, the implemented functionality can be separated into two process flows.

European eID to National Service Provider Flow. This process flow covers the case in which a European citizen, which does not have an Austrian eID, should be identified and authenticated to use an Austrian service provider. Therefore, we implement an authentication plug-in, which offers all functionality for PEPS communication to support the PEPS model, and functionality to identify and authenticate foreign citizens directly, which is identical to the middleware model. This direct identification and authentication is actually implemented for some European member states. Additionally, a mapping from European authentication information to national authentication information is required to fulfill Austrian legal requirements and to provide all necessary information to Austrian service providers [22].

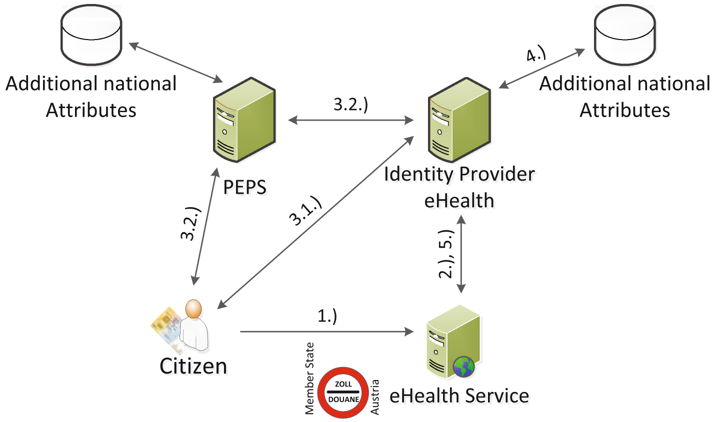

Additional national Attributes
PEPS
3.2.)
4.)
Additional national Attributes
Identity Provider eHealth
3.2.)
3.1.)
2.), 5.)
Citizen
1.)
Member State
ZOLL DOUANE
Austria
eHealth Service

Fig. 9. Process flow to authenticate an European citizen at an Austrian service provider.

Figure 9 illustrates this inbound process flow.

1. A citizen of a member state wants to access a protected area at an Austrian service provider.
2. The citizen is redirected to the identity provider and there the citizen has to select the his or her favourite identification and authentication model.
3. After selection, one of the following solutions is performed.
   (a) **Middleware Model:** In this case, the identification and authentication process is performed at the Austrian identity provider by using the citizen's secure token directly. Consequently, only information that can be provided by the secure token can be used for identification and authentication.

(b) **PEPS Model:** In this case, the citizen is redirected to the PEPS in the citizen's member state and there the identification and authentication process is performed. By using this model, some additional attributes could also be provided by using member state attribute infrastructure, which is connected to the PEPS. Afterwards, the authentication information is returned by using the STORK communication protocol.

4. To fulfill Austrian legal and technical requirements, the authentication data has to be processed by the Austrian identity provider. Therefore, we use the attribute plug-in functionality of our architecture to implement a register query plug-in, which uses the Austrian attribute mapping service [23] to fulfill these legal and technical requirements. This attribute mapping service uses the identification and authentication date received from STORK protocol to map this information to the Austrian proprietary datasets for identification information, which is a XML data structure. Additionally, this service also maps electronic mandate information, if an electronic mandate is used for identification and authentication by the foreign citizen.
5. At last, the authentication information is transmitted to the Austrian service provider and the citizen can access the protected resource.

National eID to European Service Provider Flow. The second process flow characterises the identification and authentication of an Austrian citizen to access protected resources at a European service provider. To perform this assignment, we implemented a new protocol plug-in, according to our architecture, which implements the STORK communication protocol for service provider communication. Therefore, this protocol plug-in can be used to authenticate an Austrian citizen by using his secure token.

If our solution is deployed as a single point of contact in Austria (C-PEPS) according to the PEPS model (see Fig. 7), then the member state service provider and the intermediate service provider PEPS (S-PEPS) can use the functionality of our identity provider just like an Austrian service provider can do. In this case all national legal requirements for additional attribute consuming, like the usage of electronic mandates, can be easily fulfilled.

The situation is different if the middleware model is used and our identity provider is deployed as a V-IDP which operates in the service provider infrastructure outside of Austria, because some national legal requirements cannot be achieved directly in this deployment situation. This circumstances affect mainly the attribute plug-ins, which are used to provide additional information after identification and authentication steps. In order to solve this problem, we benefit from our modular architecture design because the affected plug-ins can be easily replaced by a modified implementation, which are used in case of V-IDP deployment.

Figure 10 illustrates this deployment, in which a modified attribute plug-in for electronic mandate collection is used, as example. In contrast to the PEPS deployment, a request to the Austrian infrastructure is only necessary if requested authentication information cannot be provided by the V-IDP directly.

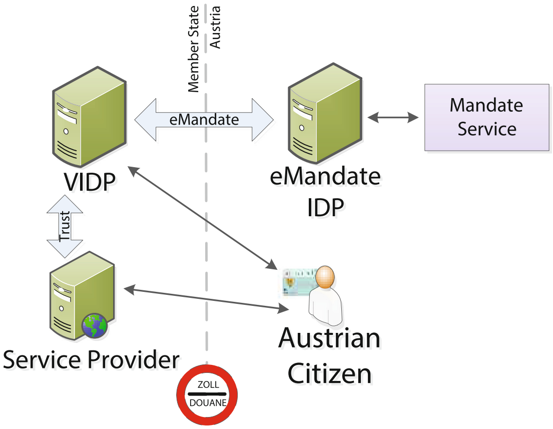


Fig. 10. Our IDP solution used as V-IDP with modified attribute plug-in.

The advantage of this solution is obtained by combining the benefits of the middleware model with the entire functionality of an Austrian identity provider.

By combining the inbound and outbound process flow, our solution can also be used to authenticate an European citizen to an European service provider. According to this, our implemented solution is also directly usable in other European states and not only in the Austrian national eID infrastructure.

6 Conclusions

Internet services and online applications are an integral component of our daily live. Such Internet services or online applications could be social network interactions and eMail applications, for example, but also are more security-sensitive services such as tax declarations or an eHealth application that protects personal medical data. The more transactions are performed by using online applications processing sensitive data, the higher is the importance for a high level of assurance into a qualified identity and a secure authentication of users. Consequently, identification and authentication of users is an integral component of general Internet services or eGovernment applications in particular. In this paper, we have presented a new architecture for identity-management systems, to provide a flexible, interoperable and easy-to-use identity provider for service provider identification and authentication. To facilitate future extensions or new requirements of identity-management systems, our solution relies on an adaptable and modular architecture. Although, the presented architecture has been implemented as an identity provider for the Austrian eID infrastructure which had to be fulfil special Austrian legal and technical requirements. But the general architectural design is also applicable to other contexts and the module implementation of the Austrian identity-provider implementation can be easily adapt to national technical or legal requirements.

We illustrate three use cases to demonstrated the practical applicability and flexibility of our implemented identity provider for the Austrian eGovernment infrastructure, which is based on our proposed architectural design. These use cases cover the use of the presented solution to identify and authenticate Austrian citizens and public officials in various ways and assure interoperability of our solution in a European context. All of this 3 use cases are implemented and practically used in different national or European online applications. In detail, the practical implementation of use case 1 is used for productive applications in the Austrian eGovernment. The implementation of the use cases 2 and 3 are actually evaluated in different national and European pilot programs. The realization of further use cases or additional functionality, like two-factor authentication in case of single sign-on, that make use of the presented architecture is regarded as future work.

References

1. European Union: Regulation (eu) no 910/2014 of the European parliament and of the council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing directive 1999/93/ec. European Union (2014)
2. European Union: Ministerial declaration, Manchester, United Kingdom, on 24 November 2005. European Union (2005)
3. European Union: Directive 2006/123/ec of the European parliament and of the council of 12 December 2006 on services in the internal market. European Union (2006)
4. Bauer, M., Meints, M., Hansen, M.: D3.1: Structured overview on prototypes and concepts of identity management systems (2005)
5. Kölsch, T., Zibuschka, J., Rannenberg, K.: Privacy and identity management requirements: an application prototype perspective. In: Camenisch, J., Leenes, R., Sommer, D. (eds.) Digital Privacy. Lecture Notes in Computer Science, vol. 6545, pp. 735–749. Springer, Berlin Heidelberg (2011)
6. Ferdous, M.S., Poet, R.: A comparative analysis of identity management systems. In: Smari, W.W., Zeljkovic, V. (eds.) HPCS, pp. 454–461. IEEE (2012)
7. Neuman, C., Yu, T., Hartman, S., Raeburn, K.: The Kerberos Network Authentication Service (V5) (2005)
8. Lockhart, H., Campbell, B.: Security Assertion Markup Language (SAML) V2.0 Technical Overview. Technical report (2008)
9. Kaler, C., McIntosh, M.: Web Services Federation Language (WS-Federation) Version 1.2 (2009)
10. Nadalin, A., Kaler, C., Monzillo, R., Hallam-Baker, P.: Web Services Security: SOAP Message Security 1.1. Technical report (2006)
11. Sakimura, N., Bradley, J., Jones, M., de Medeiros, B., Mortimore, C.: OpenID Connect Core 1.0 (2014)
12. Maler, E., Mishra, P., Philpott, R.: Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V1.1. Technical report (2003)
13. Rainer, H., Pfläging, P., Zwattendorfer, B., Pichler, P.: Portalverbundprotokoll Version 2 S-Profil (2014)

14. Leitold, H., Hollosi, A., Posch, R.: Security architecture of the Austrian citizen card concept. In: Proceedings of the 18th Annual Computer Security Applications Conference, pp. 391–400 (2002)
15. Orthacker, C., Zefferer, T.: Accessibility challenges in e-government: an Austrian experience. In: Cunningham, S., Grout, V., Houlden, N., Oram, D., Picking, R., (eds.) Proceedings of the Forth International Conference on Internet Technologies and Applications (ITA 2011), pp. 221–228 (2011)
16. Rössler, T., Hollosi, A., Liehmann, M., Schamberger, R.: Elektronische Vollmachten Spezifikation 1.0.0 (2006)
17. Hughes, J., Cantor, S., Hodges, J., Hirsch, F., Mishra, P., Philpott, R., Maler, E.: Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0. Technical report (2005)
18. Cantor, S., Moreh, J., Philpott, R., Maler, E.: Metadata for the OASIS Security Assertion Markup Language (SAML) V2.0. Technical report (2005)
19. Cantor, S., Hirsch, F., Kemp, J., Philpott, R., Maler, E.: Binding for the OASIS Security Assertion Markup Language (SAML) V2.0. Technical report (2005)
20. Leitold, H., Lioy, A., Ribeiro, C.: Stork 2.0: Breaking new grounds on eid and mandates. In: GmbH, M.M.F. (ed.) Proceedings of ID World International Congress, pp. 1–8 (2014)
21. Zwattendorfer, B., Sumelong, I., Leitold, H.: Middleware architecture for cross-border identification and authentication. J. Inf. Assur. Secur. **8**, 107–118 (2013)
22. Ivkovic, M., Stranacher, K.: Foreign identities in the Austrian e-government. In: de Leeuw, E., Fischer-Hübner, S., Fritsch, L. (eds.) IDMAN 2010. IFIP AICT, vol. 343, pp. 31–40. Springer, Heidelberg (2010)
23. Lenz, T.: A modular and flexible attribute mapping service to meet national requirements in cross-border eid federations. In: 13th International Conference on e-Society 2015, pp. 207–214 (2015)

Web Intelligence

A Personalized and Context-Aware News Offer for Mobile Devices

Toon De Pessemier(✉), Kris Vanhecke, and Luc Martens

Department of Information Technology, iMinds, Ghent University,
G. Crommenlaan 8/201, 9050 Ghent, Belgium
{toon.depessemier,kris.vanhecke,luc.martens}@intec.ugent.be

Abstract. For classical domains, such as movies, recommender systems have proven their usefulness. But recommending news is more challenging due to the short life span of news content and the demand for up-to-date recommendations. This paper presents a news recommendation service with a content-based algorithm that uses features of a search engine for content processing and indexing, and a collaborative filtering algorithm for serendipity. The extension towards a context-aware algorithm is made to assess the information value of context in a mobile environment through a user study. Analyzing interaction behavior and feedback of users on three recommendation approaches shows that interaction with the content is crucial input for user modeling. Context-aware recommendations using time and device type as context data outperform traditional recommendations with an accuracy gain dependent on the contextual situation. These findings demonstrate that the user experience of news services can be improved by a personalized context-aware news offer.

Keywords: Recommender system · Context-aware · Real-time · Mobile · News · User evaluation

1 Introduction

Recommender systems are software tools and techniques providing suggestions for items to be of interest to a user such as videos, songs, or news articles. The consumption of these audiovisual media and the accessing of information always happen in a certain context [31], i.e. conditions or circumstances that significantly affect the decision behavior. This gave rise to the development of context-aware recommender systems (CARS), which take this contextual information into account when providing recommendations.

For various application domains, the user context has gained an increased interest from researchers [3]. For context-aware music recommendations for example, the user's emotions can be used as input by using support vector machines as emotional state transition classifier [16]. In the application domain of tourism for example, various applications use the current location or activity of the user to personalize and adapt their content offer to the current user

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 147–168, 2016.
DOI: 10.1007/978-3-319-30996-5_8

needs [12,30]. Personal recommendations for points of interest can be provided based on the user's proximity to the venue [20].

In the domain of audiovisual media, more specifically news content, the influence of context on the consumption behavior and personal preferences is less obvious. However, research [42] has shown that the situation of the user (location, activity, time), as well as the device and network capabilities are important contextual parameters for context-aware media recommendations on smartphones.

The growth of the digital news industry and especially the development of mobile products is booming. Mobile has become, especially amongst younger media consumers, the first gateway to most online news brands. In a recent survey [29], conducted in 10 countries with high Internet penetration, one-fifth of the users now claim that their mobile phone is the primary access point for news. Despite this shift of news consumption to the mobile platform, the study of Weiss [40] highlights that a gap exists between what news consumers, particularly young adults, are doing and using on their smartphones and what news organizations are able to provide. In most cases, news organizations disregard contextual data or they are only using geo-location features in their mobile apps for traffic and weather; they do not anticipate the high use of location-based services by smartphone consumers.

In contrast to more traditional content domains of research on recommender systems, such as movies or books, news content are typically transient items. They are characterized by a short life span and quickly lose their information value over time. News items should therefore be recommended as soon as they are available in order to minimize delay between production and consumption of the content. For instance, a preview of a sports game has lost any information value after the game. Especially for online news, fast delivering and recommending of content is of utmost importance.

For content with a short life span, and for news content in particular, collaborative filtering (CF) systems have difficulties to generate recommendations because of the new item problem (cfr. cold start problem) [11]. CF requires a critical amount of consumptions (explicit or implicit feedback) before an item can be recommended. Once enough consumption data is available, the information value of the content might be degraded, making recommendations for the content useless. Therefore, content-based or hybrid approaches are considered as more suitable for news recommendation.

2 Related Work

In the domain of digital news services, various initiatives to personalize the offered news content have been proposed. One of the first recommender systems for personalizing news content was GroupLens [21]. GroupLens used collaborative filtering to generate recommendations for Usenet news and was evaluated by a public trial with users from over a dozen newsgroups. This research identified some important challenges involved in creating a news recommender system.

Another digital news service is SCENE [23]. It stands for a SCalable two-stage pErsonalized News rEcommendation system. The system considers characteristics such as news content, access patterns, named entities, popularity, and recency of news items when generating recommendations. The proposed news selection mechanism demonstrates the importance of a good balance between user interests, the novelty, and diversity of the recommendations.

The News@hand system [8] is a news recommender which applies semantic-based technologies to describe and relate news contents and user preferences in order to produce enhanced recommendations. This news system ensures multi-media source applicability and multi-domain portability. The resultant recommendations can be adapted to the current context of interest, thereby emphasizing the importance of contextualization in the domain of news. However, context is not the main focus of this study and the influence of context on the consumption behavior is not investigated.

The News Recommender Systems Challenge [32] focused on providing live recommendations for readers of German news media articles. This challenge highlighted why news recommendations have not been as analyzed as some of the other domains such as movies, books, or music. Reasons for this include the lack of data sets as well as the lack of open systems to deploy algorithms in. In the challenge, the deployed recommenders for generating news recommendations are: Recent Recommender (based only on the recency of the articles), Lucene Recommender (a text retrieval system built on top of Apache Lucene), Category-based Recommender (using the article's category), User Filter (filters out the articles previously observed by the current user), and Combined Recommender (a stack or cascade of two or more of the above recommenders).

The usefulness of retrieval algorithms for content-based recommendations has been demonstrated with experiments using a large data set of news content [6]. Binary and graded evaluation were compared and graded evaluation showed to be intrinsically better for news recommendations. This study emphasizes the potential of combining content-based approaches with collaborative filtering into a hybrid recommender system for news.

Although the various initiatives emphasize the importance of a personalized news offer, most of them focus on the recommendation algorithms and ignore the contextual information that is coupled with the information request, the user, and the device. In this study, the focus is not on improving state of the art recommendation algorithms, but rather on investigating the influence of context on the consumption of news content by means of a large-scale user study.

In many cases, the research on CARS remains conceptual [3], where a certain method has been developed, but testing is limited to an offline evaluation or a short-term user test with only a handful of people, often students or colleagues who are not representative for the population. In contrast, this research investigates the role of context for news recommendations by means of a large-scale empirical study. Users could utilize a real news service[1] that offers content of four

[1] http://www.iminds.be/en/projects/2014/04/17/stream-store.

major Flemish news companies on their own mobile devices, in their everyday environment, where and when they wanted, i.e., in a living lab environment.

Living lab experiments are an extension towards more natural and realistic research test environments [14]. Living labs allow to evaluate research hypotheses by users representative for the target population who satisfy their information need in a real context. Since users are following their own agenda, laboratory biases on their behavior can be neglected [18]. Although less transparent and predefined, living lab experiments aim to provide more natural settings for studying users' behavior and their experience [10].

Especially for context-aware applications, in which the user's environment has an influence on the way the application works and/or on the offered content, a realistic setting is essential for a reliable evaluation. Therefore, this paper investigates the influence of context and the benefit of context-aware recommendations for a real news service by means of a large-scale user panel, in a realistic environment, over a longer period of time. Since a user study can provide reliable explanations as to which recommendation method is the best, and why one method is better than the other [33], three alternative recommendation methods for the news service are compared through such a user study.

3 Search Engines and Recommender Systems

To generate recommendations, a content-based approach was chosen because of the availability of informative metadata about the content items, the sparsity of the data set, and the cold start problem associated with the start-up phase (Sect. 1). Content-based algorithms typically compare a representation of the user model with (the metadata of) the content, and deliver the best matching items as recommendations [24]. These algorithms often use relatively simple retrieval models such as keyword matching or the Vector Space Model (VSM) with basic Term Frequency - Inverse Document Frequency (TF-IDF) weighting [25]. As such, the matching process of content and user model in a content-based algorithm shows many resemblances with the content retrieval process of a search engine.

Before employing the VSM and TF-IDF weighting in a content-based algorithm, preprocessing of the content is often required. If the content consists of complete sentences, the text stream must be broken up into tokens: phrases, words, symbols or other meaningful elements. Tokens that belong together, e.g. United States of America or New York, deserve special attention, and can be handled by reasoning based on uppercase letters and n-gram models [7]. Before further processing of the content, the next operation is filtering out stop words, the most common words in a language that typically have a limited intrinsic value. Another important operation is stemming, the process for reducing inflected (or sometimes derived) words to their word stem, or root form. In our implementation, Snowball [28] is used, a powerful stemmer for the different languages. Again, a resemblance with content retrieval processes can be noticed, since these preprocessing operations are also performed during the indexing of web pages in search engines.

Based on these resemblances between the content recommendation and content retrieval problem, we opted to utilize a search engine as the component for processing and indexing the content in our news service. Utilizing a search engine to process and index the news content brings some additional advantages.

- *Short Response Time.* Search engines are strongly optimized to quickly identify and retrieve relevant content items. An inverted index [9] is used as a very efficient structuring of the content, enabling to handle massive amounts of documents.
- *Fast Processing of New Content.* New content items can be processed quickly by making additions to the index structure, thereby making these new content items available for recommendation almost immediately. In contrast, traditional collaborative filtering systems often require intensive calculations of similarities before a new item can be recommended.
- *Limited Storage Requirements.* The index structure of search engines is a very efficient storage way to retrieve documents.

4 Recommendation Architecture

Figure. 1 shows the architecture and content flow of the news recommender system. The five different phases will be discussed in more detail in this section.

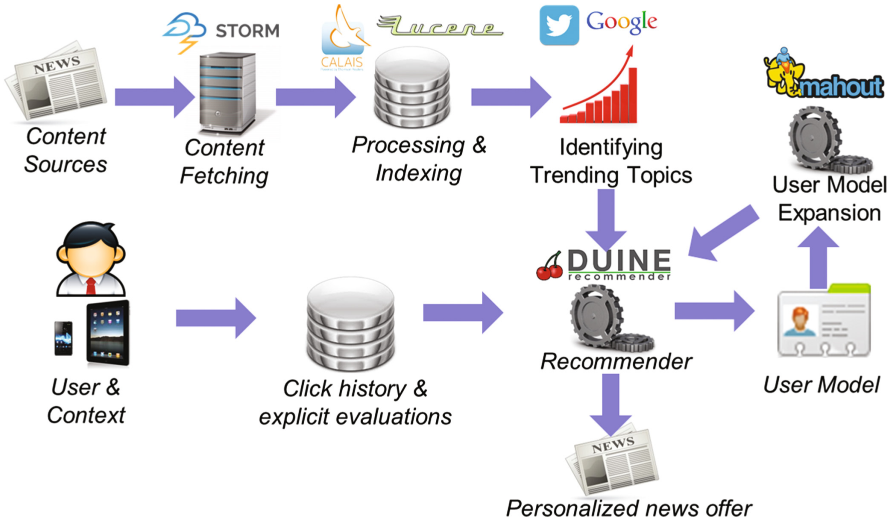


Fig. 1. The architecture and content flow of the news recommender system.

4.1 Content Fetching

The first phase of the recommendation process consists of *fetching the news content* periodically from different sources. When new items are available, their content is fetched and processed. Many online news services provide their content through RSS-feeds. To parse these feeds, the Rome project [41] is used since this is a robust parser. Besides RSS-feeds, other sources, such as blogs, can also be incorporated into the system by using a specific content parser.

In order to keep track of the most recent news content, news sources are checked regularly for new content. Different news sources have a different publishing frequency, ranging from one news item per day, to multiple news items per minute. Therefore, we used a simple mechanism to adapt the frequency of checking for new content to the publishing frequency of the content source. For each content source, a dynamic timer is used to determine when to check for new content. After a timeout, the content is fetched. If new content is available, the content item is passed to the search engine and the timeout is reduced by half. If no new content is available, the timeout is doubled. This simple mechanism showed to be sufficient as a convergence method for the timeout parameter.

In order to process the stream of incoming news articles of different sources continuously, Apache Storm [4] was used. Storm enables the processing of large streams of data in real time. As opposed to batch processing, Storm handles the news articles as soon as these are available. To use Storm, a topology composed of ‘Spouts’ and ‘Bolts’ has to be built, which describes how messages flow into the system and how they have to be processed. A Spout is a source of data streams. A Bolt consumes any number of data streams, does some processing, and can emit new data streams. Storm can make duplicates of these components, and even distribute these duplicates over multiple machines, in order to process large amounts of data. As a result, Storm makes the system scalable and distributed.

Figure 2 visualizes the Storm topology of our implementation. The Spouts input data into the system as URLs of RSS-feeds, blogs, or social network accounts. Storm will distribute the work load over different Bolts of the first type, which fetch the data from the feeds. In case new articles are available in the feed, the URLs of these articles are passed to the Bolts of the second type. These Bolts fetch the article content and remove non-topical information, such as advertisements, by identifying specific HTML tags in the source code of the web page. Subsequently, the Bolts pass the article content to Bolts of the third type. The task of Bolts of the third type is to analyze the content and obtain information such as the title, date, category, etc. Next, the article content is passed to the fourth type of Bolts, which input the news articles into the processing engine. After inputting the content into the processing engine, statistical information about the article content is stored by the fifth and last type of Bolts. E.g., the frequency of occurrence of a term at a specific moment in time is used to determine if a news topic is trending and important (Sect. 4.3).

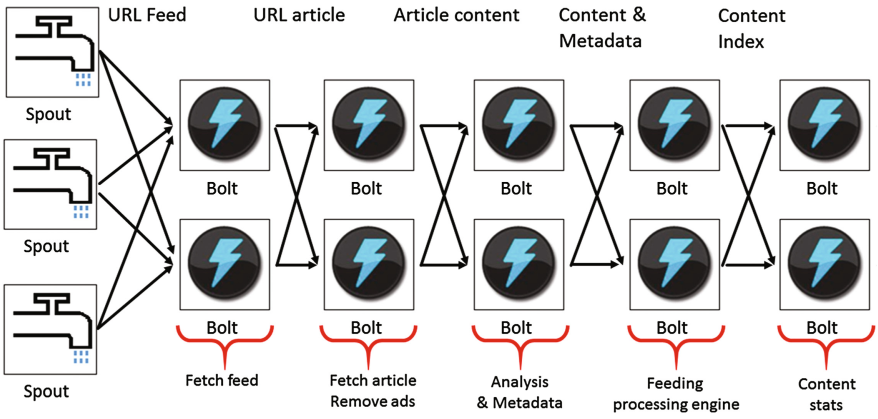


Fig. 2. The Storm topology of our system.

4.2 Processing Engine

In the second phase, the content is *processed and indexed*. Given the equivalence between recommender engines and search engines [34], we opted to use a search engine for tasks such as indexing, retrieving items, and identifying n-grams.

Apache Lucene [36] was chosen as search engine, a Java library that is typically used for services handling large amounts of data and offering search functionalities. Since Lucene's performance, simplicity, and ease-of-use have been investigated in related work [17], this research does not focus on the characteristics of Lucene, but rather on the combination of search engine and recommender system.

As alternative search engines, we considered Apache Solr [38] and ElasticSearch [13]. Solr is a ready-to-use, open source search engine based on Lucene. In comparison with Lucene, Solr provides more specific features such as a REST webinterface to index and search for documents. However, the disadvantage of Solr is that some of the specialized functionality (that is needed in the recommendation algorithm) is hidden and not directly usable. Besides, the overhead of the webinterface of Solr introduced some delay in comparison with Lucene in our experiments. Similar to Solr, ElasticSearch hides some of Lucene's functionality by using a simple web interface. Specific information about the content items, such as the term frequencies or statistics about the complete index, are not directly accessible using ElasticSearch. Therefore, Lucene was chosen to provide the functionality of the search engine. In case the processing load for the Lucene index becomes an issue, distribution over different machines is possible by solutions such as Katta [19], thereby making it scalable.

News items in our system are characterized by eight different categories (national, international, culture, economy, lifestyle, politics, sports, and interesting facts), which are used to elicit the content preferences of the users through a questionnaire (Sect. 6.1). In addition, more detailed info of the news items is extracted through keywords and named entities, i.e. names of persons, organizations, locations, expressions of times, quantities, monetary values, etc. These keywords and named entities are not predefined but are extracted from the text of the article using OpenCalais [39]. OpenCalais is a Web service that automatically creates rich semantic metadata for the content. It analyzes the news article and finds the entities within it, but it returns the facts and events hidden within the text as well. This way, the news article is tagged and analyzed with the aim of checking whether it contains information what the user cares about.

4.3 Identifying Trending Topics

News events with a high impact (e.g., a huge natural disaster in a remote part of the world) have to be detected and considered as a recommendation, even if the topic does not completely match the user's interests. In the third phase of the recommendation process, these *trending topics* are identified based on their frequency of occurrence in the index of the search engine. If the current frequency of occurrence is significantly higher than the frequency of occurrence in the past, the topic is considered as trending.

Besides, trending topics are discovered by checking trends on Google's search queries [15]. Every hour, Google publishes a short list with trending searches. A special Spout was implemented to fetch these trending topics hourly. Trending topics are used to create a query for the search engine, and the resulting news items are added to the user's recommendation list.

A final source of trending topics is Twitter. Research has shown that Twitter messages are a good reflection of topical news [27]. Therefore, another Spout was assigned specifically to query tweets regarding news topics using the Twitter API. Twitter accounts of specialized news services and newspapers were followed. The tweets originating from these accounts are focusing on recent news and characterized by a high quality. Retweets and Favorites give an indication of the popularity and impact of a tweet. Subsequently, Tweets are processed in the same manner as other news items by Bolts.

4.4 Generating Recommendations

In the fourth phase, personalized recommendations are generated. As content-based solution, the 'InterestLMS algorithm' of the Duine framework [35] was adopted. The InterestLMS algorithm is based on the VSM. It *builds a user model* by inferring personal preferences from the metadata describing the news items that are requested (implicit feedback) or evaluated (explicit feedback). As is common practice in the VSM [24], the user model is processed as a vector of terms (tags) together with a value specifying the user's interest in the term. These terms are words (or n-grams) in the article that are identified as relevant for the content (see Sect. 4.2).

Based on requests for reading news items (implicit feedback) or evaluations using the thumbs up/down functionality (explicit feedback), the user model is continuously *updated*. This feedback is transformed into a rating score.

Thumbs up is mapped to the maximum rating, whereas thumbs down is the minimum rating. For implicit feedback, the amount of time spent on a news item (reading time for text or watching time for pictures and videos) is translated into a rating.

These item ratings are normalized (*normRating*) and then used to create or update the terms of interest in the user model that correspond to the metadata fields of the content item (Eq. 1).

$$newTerm_i = Aging{*}currentTerm_i{+}updateModerator{*}normRating{*}Nfactor \tag{1}$$

Here, the updateModerator is a constant that specifies the rate in which interests in topics are updated in the user model. The $Nfactor$ corrects for the number of terms ($\#terms$) that describe a content item: $Nfactor = 1/\#terms$.

In this update model, articles from the past are considered as less representative for the user's preferences than recent articles. Therefore, the *Aging* factor, a constant smaller than 1, decreases the contribution of terms of older news items.

Subsequently, the algorithm determines the news items that best match the user model reflecting the user's preferences. The interest terms in the user model are used to infer a *recommendation score* for each unseen news item based on the terms describing the item (Eq. 2).

$$recommendationscore = \sum_i currentTerm_i * weight(Term) \tag{2}$$

Here, the sum iterates over all terms identified in the unseen news item. $CurrentTerm_i$ stands for the preference value of term i in the user model. The weights specify the relative importance of different terms (e.g., categories are considered as more important than keywords). By ordering and filtering the unseen news items according to their recommendation score, a collection of suitable recommendations is generated based on the user's personal preferences for news content of different categories and characterized by different keywords.

4.5 User Model Expansion

As explained in the introduction, straightforward collaborative filtering is not usable for news recommendations because of the new item problem. Unfortunately, content-based recommendations are often characterized by a low serendipity; recommendations are too obvious. To introduce serendipity, a hybrid approach was taken by adding a collaborative filtering aspect to the content-based recommender. A traditional nearest neighbor approach was used to calculate similarities between user-user pairs. Instead of recommending the items that the neighbors have consumed (as in a collaborative filtering approach), our implementation will recommend terms that are prominent in the user models of neighboring users. These terms of the neighboring user models are used to expand the user model created by the InterestLMS algorithm, thereby making it more diverse. Subsequently, this expanded user model is used to generate content-based recommendations as described in Sect. 4.4.

By expanding the user model with terms that are significant in the model of the user's neighbors, user models are broadened and diversified with related terms. These expanded user models will produce more diverse recommendations covering a broad range of topics. Since the additional terms are originating from neighbors' user model, the added terms will probably be in the area of interest of the user. The collaborative filtering component is based on the implementation of Apache Mahout [37]. Mahout ensures the scalability of this component of the system. Moreover, the user model expansion is not a time-critical component, and is therefore implemented as a batch process running periodically. Content-based recommendations are based on the current version of the user model, and as soon as the model expansion is finished, the user model is updated. This ensures that real-time recommendations can be generated at all time.

5 Experimental Setup

The news service that was used in this experiment, aggregates content of different premium content providers: newspapers, magazines, but also content of television as short video clips. Figure 3 shows a screenshot of the user interface offering the content of different providers. The aggregated content offers users a more complete and varied overview of the news than traditional services do. To anticipate the abundance of news content and the associated choices that people have to make, the news service offers personalized recommendations.

During the experiment, the device type (smartphone or tablet) and the time of the day (morning, noon, daytime, or evening) are studied as contextual influences of the news consumption. The news service is accessible through a mobile application, which is available on Android and iOS for tablets as well as smartphones. As a result, the type of device that is used for consuming the news content is an interesting contextual factor.

Compared to the well-established application domains of recommender systems, such as movies or books, news items have a shorter lifespan and are frequently updated. Consequently, consulting the news on a daily basis, or even multiple times a day, can be interesting, which makes the time aspect another important contextual factor. The time is closely related to the location of the user, as was also witnessed during the analysis of users' interactions and consumption behavior. A frequently recurring pattern was as follows: in the morning, users are at home; during daytime, they are at work; and during the evening, they are again at home. Therefore, and to prevent over-specification (Sect. 6.3), the location is not adopted as a separate contextual factor.

For the evaluation of this service and its recommendations, 120 test users were recruited by an experienced panel manager from iMinds-iLab.o[2] (i.e. a research division with a strong expertise in living lab research and panel management). These test users, all interested in news and owning a smartphone and/or tablet, belong to the target group of an online news service. The test users could install the mobile application of the news service on their smartphone and/or tablet

[2] http://www.openlivinglabs.eu/livinglab/iminds-ilabo.

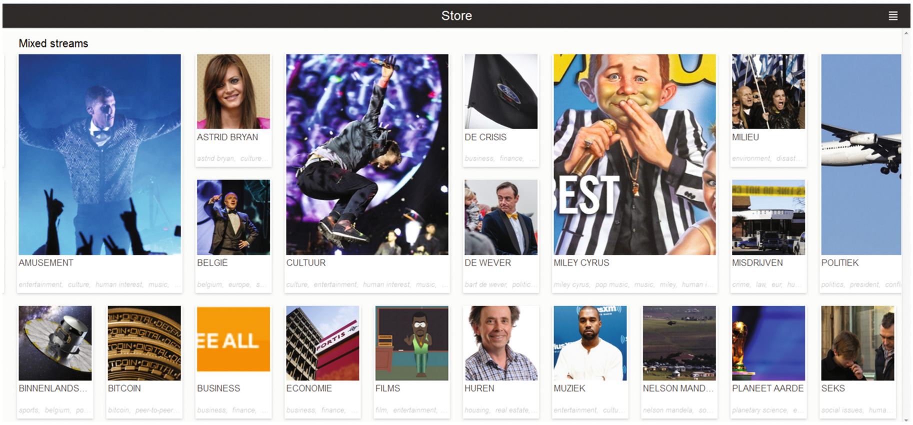


Fig. 3. Screenshot of the user interface of the news service.

and freely use the service during the evaluation period of around 5 weeks. These test users were divided into three groups, each receiving a different type of recommendations, as explained in Sect. 6.

The test users' interactions with the service were logged to analyze their consumption behavior and to get insight in the actual use of the news service and their overall experience: 10 test users did not install the app, or did not use the news service during the evaluation period. They are excluded from the analysis, so that the number of actual participants was reduced to 110.

6 Three News Recommendation Approaches

The experiment takes three different approaches to recommend interesting news. Each user received only one type of recommendations during the whole evaluation period. To avoid any bias, test users were not informed about the existence of multiple types of recommendations.

6.1 Recommendations Based on Explicit Static Preferences

Before the actual experiment, test users were asked about their preferences for the eight different categories of news content (Sect. 4.2) through an online questionnaire. Users could specify their interests on a 5-point rating scale for each category and refine this score for different times of the day (morning, noon, daytime, and evening). The answers on this questionnaire constitute the user model that is used for generating news recommendations (Sect. 4.4). During the experiment, these preferences are considered static; user models are not updated based on explicit or implicit feedback on the content, and the recommender is not learning from the user's behavior.

6.2 Content-Based Recommendations

The content-based recommendations are not based on a prior questionnaire but use the implicit and explicit feedback users provide during the evaluation period. A request to read one of the recommended news items is considered as positive implicit feedback. Evaluating the news recommendation by means of the 'thumbs up' and 'thumbs down' icons in the user interface provides explicit feedback.

This feedback is gradually collected during the usage of the service. As a result, the user models of the content-based recommender are dynamic and constantly change as users interact with the news service. As the user is utilizing the news service and provides feedback, the recommender is learning the user's preferences.

For the content-based recommendations, contextual aspects are not taken into account. So, contextual data, such as the device type and the time of the day, are ignored during the creation of the user model and the calculation of the recommendations. In Sect. 4.4, more details about the recommendation algorithm are provided.

6.3 Context-Aware Content-Based Recommendations

Just like the content-based recommendations, the context-aware content-based recommendations are not using a prior questionnaire but are self-learning based on the explicit and implicit feedback users provide during the experiment. For this type of recommendations, the content-based algorithm is extended to take into account the context of the user. Before generating the recommendations, the user feedback is processed by a contextual pre-filter [2]. Contextual information is used to determine the relevance of the feedback and filter these data based on the current situation. For instance, if a user wants to read news during the evening, an *exact pre-filter* [3] selects only feedback gathered during the evening to calculate the recommendations. Therefore, the day is partitioned into four non-overlapping intervals: morning from 6:00 to 11:00, daytime from 11:00 to 12:00 and from 13:00 to 18:00, noon from 12:00 to 13:00 and evening/night from 18:00 to 6:00.

One major advantage of the contextual pre-filtering approach is that it allows deployment of any of the traditional recommendation techniques [1]. This makes it possible to use the same underlying algorithm (Sect. 4.4) for the context-aware content-based recommendations as for the content-based recommendations, which enables the comparison of both types of recommendations and to investigate the influence of contextual information.

Different pre-filtering techniques have proven their efficacy in literature [5]. They all have to cope with the problem of context over-specification: focusing on the exact context is often a too narrow limitation. An overly specified context may not have enough training examples for accurately estimating the user's interests. For example, if a user rarely utilizes a tablet during noon to read news articles, the exact context (noon + tablet) may not provide enough data (feedback from the user) for an accurate user model, which gives rise to the 'sparsity' problem. As a result, insufficient feedback is available for generating reliable recommendations [26].

An appropriate solution for context over-specification is to use a more general context specification by applying context generalization [3]. Since certain aspects of the overly specific context may be less significant, the data filtering can be made more general in order to retain more data after the filtering for calculating recommendations.

In this experiment, context generalization is applied in two phases in case of insufficient feedback data. In a first phase, the time frame is broadened. For instance, if recommendations are needed for a user who is reading news on a tablet during noon, the time restriction "noon" is dropped first. The data gathered in that specific context is supplemented with the user's feedback gathered on a tablet during other time periods. If the amount of feedback is still insufficient after this first generalization, the context is further generalized. In a second phase, the device type is broadened. More specifically, the user's feedback provided on a specific type of device (e.g., a tablet) is supplemented with the user's feedback provided on other device types (e.g., a smartphone). We opted to apply the generalization first on the time aspect of the context, and in a second phase on the device type, since many users are utilizing the service during different time periods but typically prefer one type of device per time period.

7 Results

7.1 Usage Patterns Throughout the Day

Figure 4 shows the amount of user interaction with the service (i.e., selecting a news item to view), aggregated over all users, for each hour of the day and per device type. A clear pattern in the consumption behavior is visible throughout the whole day. The close relation between time and location (Sect. 5) may have strengthened this pattern.

As expected, the amount of activity with the service is limited during the night. In the morning, users are very interested in the news, which is reflected in a peak in the service usage. During the day, the amount of consumptions varies slightly per hour with a slight increase around noon on tablet. During the evening, users spend more time reading news, which is revealed in Fig. 4 by the increased user activity.

Comparing both device types (smartphone vs. tablet) reveals that tablets are commonly used for consulting the news during the morning and evening. In contrast, smartphones are the primary device for consulting news during daytime. This analysis confirms the assumption that the usage of a news service and the amount of news content that is read, is strongly linked to the time of the day and the device type.

7.2 Three Recommendation Approaches Evaluated

To quantify the added value of a dynamic user model and contextual information, the users' interactions (implicit and explicit feedback) with each of the three

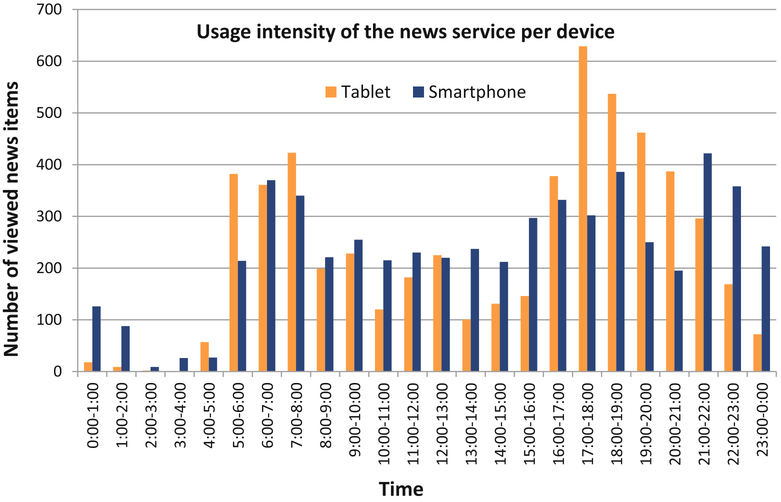


Fig. 4. The amount of user interaction with the news service, aggregated over all users and partitioned according to the hour of the day.

types of recommendations are analyzed. Figure 5 gives an overview of the user feedback on the news service obtained during the evaluation period. The chart distinguishes implicit feedback, i.e. requesting to 'view' a news item, and explicit feedback, i.e., evaluating a news item by providing a 'thumbs up' or 'thumbs down' rating.

In Fig. 5, this user feedback is aggregated over all users and partitioned by the type of recommendations that the users received. Since some test users dropped out just before the evaluation period, the different recommender types are not evaluated by the same number of test users. Table 1 shows the number of test users assigned to each type of recommendations, which has a direct influence on the total amount of feedback gathered for that type.

During the evaluation period, 2931 positive evaluations ('thumbs up') of a news recommendation were registered for all types of recommendations together. In contrast, only 1074 times a negative evaluation ('thumbs down') was provided by the users. These aggregated values ('thumbs up' 73.1 % - 'thumbs down'

Table 1. Comparison of the recommendation approaches

Recommendation Approach	Input Data	Number of Test Users	$\frac{\#Thumbs\ up}{\#(Thumbs\ up+down)}$
Explicit Static Preferences	Questionnaire	38	66.9 %
Content-Based	Feedback	37	73.7 %
Context-Aware Content-Based	Feedback + Context	35	79.3 %

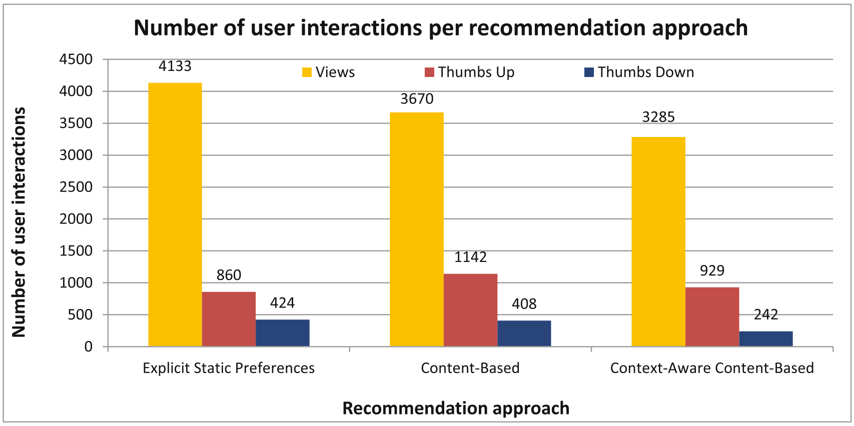


Fig. 5. The amount of user interaction with the news service for each type of recommendations, partition according to the type of interaction.

26.9 %) are an indication for the general satisfaction of the users with the news that they get recommended.

However, significant differences between the different types of recommendations can be witnessed. Compared to the recommendations that are based on the explicit static preferences of the users, less views (total number, but also number per user) are obtained for the content-based and context-aware content-based recommendations. This indicates that users get the interesting news more quickly using the more advanced algorithms, since they also spent more time per news item.

Comparing the different types of recommendations in terms of negative feedback ('thumbs down') demonstrates the added value of personal feedback and the context of the user for the recommender system. Recommendations based on explicit static preferences received 424 times 'thumbs down' from users who are not satisfied with the news content. The content-based recommender uses the implicit feedback (requests to view a news item) and explicit evaluations ('thumbs up & down') as personal feedback during the evaluation period. Compared to the recommendations based on explicit static preferences, less negative (408 times 'thumbs down') and more positive evaluations are provided for the content-based recommendations. The lowest number of negative evaluations (242 times 'thumbs down') was achieved with the context-aware content-based recommender, which suggests news items based on the personal feedback of the user and by taking into account the user's current context. A Wilcoxon rank-sum test showed these differences are significant ($p = 0.04 < 0.05$).

Table 1 shows the ratio of the number of positive evaluations (# thumbs up) and the number of explicit evaluations (# thumbs up + down) for the different types of recommendations. The results confirm the increase in accuracy by making the system dynamic (content-based recommendations) and taking into account the context (context-aware content-based recommendations).

The improvement obtained by dynamic profiles is further investigated in Sects. 7.3 and 7.4 discusses the influence of context on the recommendation accuracy.

7.3 Accuracy Improvement Through Dynamic Profiles

The accuracy increase obtained by making the user model dynamic is further investigated. Figure 6 shows the users' interests in news content of different categories. These interests can be obtained by explicitly asking the user through a questionnaire (as done in the recommendation approach based on explicit static preferences) or by deducing them from the actual user interactions with the service (as done in the content-based recommendation approach). These two methods are compared in Fig. 6, which shows a clear difference. In general, users express a higher degree of interest in all news categories through the questionnaire compared to their actual interaction behavior with the news content. In other words, their expressed interests are only partly reflected in their selection of news content. This discrepancy might be due to time constraints of the user or due to the fact that the user can consult other information sources for news content.

More importantly is the difference in relative importance of the various news categories. Through the questionnaire, users express most interest in categories such as national news, international news and politics, whereas lifestyle is the least interesting category for them. In contrast, analyzing the actual user behavior results in opposite conclusions. Lifestyle is the most popular news category in terms of the number of views during the evaluation period. A possible explanation for this contrast is social desirability, i.e. the tendency of survey respondents to answer questions in a manner that will be viewed favorably by others.

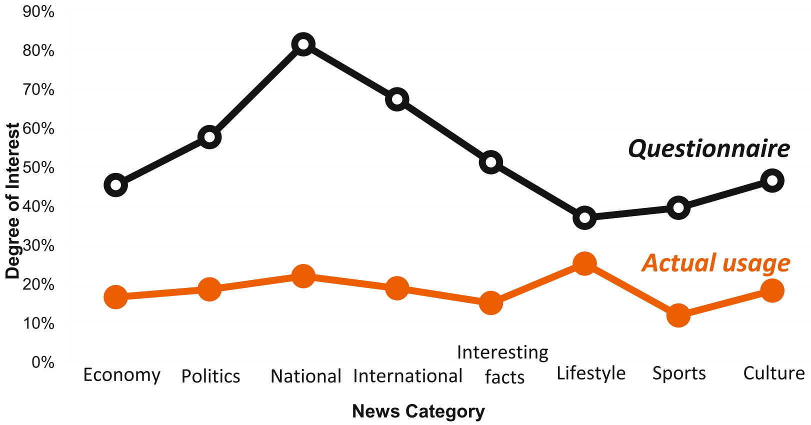


Fig. 6. The users' interests per news category: users opinion (questionnaire) vs. actual usage (mobile app).

These results show a significant difference between what users state that is interesting for them, and what users actually select and consume. As a result, making profiles dynamic by incorporating interaction behavior in the user model is of crucial importance to adjust the recommendations to the users' actual interests.

7.4 Accuracy Improvement Through Context

To investigate the influence of context, the accuracy of the recommendations is investigated in different contextual situations. Table 2 shows the ratio of the number of positive evaluations (# thumbs up) and the number of explicit evaluations (# thumbs up + down) for different device types (smartphone and tablet). This tables compares the recommendations based on explicit static preferences and the context-aware content-based recommendations in terms of received feedback from the test users.

For the recommendations based on explicit static preferences, the results show a lower accuracy for recommendations made on tablet (62.9 %) compared to smartphone (73.5 %). The accuracy of the context-aware content-based recommendations is higher in both context situations. But the accuracy difference between smartphone (79.6 %) and tablet (75.2 %) is reduced (from 10.6 % to 4.4 %) if the context is taken into account. This shows that the context-aware content-based recommender adjusts the news content to the contextual situation of the tablet. For example, news items that discuss specialized topics in large detail may be better suited for tablet devices than for smartphones. In addition, users may have specific preferences or habits regarding device type and content categories, such as reading sport news on smartphone and political news on tablet. The context-aware content-based recommender is able to automatically learn these preferences thereby recommending content adjusted to the device type.

Table 3 shows the ratio of the number of positive evaluations (# thumbs up) and the number of explicit evaluations (# thumbs up + down) at different time periods (morning, daytime, noon, evening). Again, the accuracy is compared for recommendation based on explicit static preferences and context-aware content-based recommendations.

For the recommendations based on explicit static preferences, the results show a rather stable accuracy over the different time periods. As shown in Sect. 7.1, the typical periods to consult the news are the morning and the evening. During noon and daytime, less time is spent on reading news articles. However, it is important to provide users an adjusted news offer during these time periods, such as an update of the morning news. Context-aware content-based recommendations take the time period into account, thereby achieving an accuracy gain of 1.6 % and 8.9 % for respectively morning and evening, i.e. the time periods most users consult the news. For the less typical time periods, the accuracy gain is much higher by exploiting the time-dependent needs of the users. The context-aware content-based algorithm improves the recommendation accuracy with 17.9 % and 18.7 % respectively during daytime and noon.

Table 2. Comparison of the recommendation results per device type

Recommendation Approach	Context	$\frac{\#Thumbs\ up}{\#(Thumbs\ up+down)}$
Explicit Static Preferences	Tablet	62.9 %
Context-Aware Content-Based	Tablet	75.2 %
Explicit Static Preferences	Smartphone	73.5 %
Context-Aware Content-Based	Smartphone	79.6 %

Table 3. Comparison of the recommendation results per time period

Recommendation Approach	Context	$\frac{\#Thumbs\ up}{\#(Thumbs\ up+down)}$
Explicit Static Preferences	Morning	65.4 %
Context-Aware Content-Based	Morning	67.0 %
Explicit Static Preferences	Daytime	65.4 %
Context-Aware Content-Based	Daytime	83.3 %
Explicit Static Preferences	Noon	70.3 %
Context-Aware Content-Based	Noon	89.0 %
Explicit Static Preferences	Evening	68.3 %
Context-Aware Content-Based	Evening	77.2 %

7.5 Discussion

While the context-aware content-based recommendations received the most positive evaluation in this experiment (highest ratio in Table 1), the gain in accuracy with respect to the static user model may become bigger over time.

In this experiment, the results are based on the evaluation period of approximately 5 weeks, and CARS require sufficient time to learn user preferences in different contextual situations. Because of the cold start problem (i.e. the issue that the system cannot draw any inferences for users or items about which it has not yet gathered sufficient information) and data fragmentation over the different contextual situations, we believe that there is still room for accuracy improvement of the context-aware content-based recommendations by gathering additional user feedback over a longer time period. The required number of ratings to overcome the cold start problem depends on various factors such as the algorithm parameters, the content domain, and the specific items that are rated. Studies have shown that, in general, more than 20 to 30 ratings are necessary for the system to recommend relevant items to the user [22]. In our user study, some of the users did not achieve enough ratings for each contextual situations. Additional data can help to learn patterns in the users' behavior and preference differences for various contextual situations, thereby further improving the accuracy of the context-aware content-based recommendations.

8 Conclusions

In this paper, a start-up news service offering personal recommendations is evaluated by an empirical user study. The typical characteristics of news content, such as the short-term life, and the limitations of a start-up service, such as a limited community of active users, make pure collaborative filtering techniques unusable. Therefore, the recommendation engine combines features of a content-based algorithm with a search engine and collaborative filtering using technologies such as Storm, Lucene, Duine, and Mahout. Storm enables the fast processing of large streams of news content. Lucene provides the functionality of a search engine and is used for processing and indexing the content. The Duine recommender framework is used to generate the content-based recommendations. The collaborative filter of Mahout is used to exchange terms of the user model among neighboring users. User models, as used in the content-based algorithm, are expanded with related terms interesting to read about. In the user experiment, three types of recommendations are tested: recommendations based on an explicit static user model, content-based recommendations using the actual user behavior but ignoring the context, and context-aware content-based recommendations incorporating user behavior as well as context.

The study aimed to assess the importance of context in the recommender of a real-life mobile news service by focusing on two contextual aspects: device type and time. The results confirm the added value of contextual information for personalized news recommendations by an increased recommendation accuracy. A more profound analysis showed that in specific contextual situations, a bigger accuracy gain can be obtained by using context-aware recommender systems, whereas in other situations the accuracy gain is limited. In this experiment, the context-aware algorithm obtained the best results on tablet devices and during time periods that are less typical for news consumption, such as during daytime and at noon. As future work, we consider to make a distinction between short-term interests and long-term interests of users. We also plan to focus more on entities mentioned in articles.

References

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **17**(6), 734–749 (2005)
2. Adomavicius, G., Tuzhilin, A.: Tutorial on context-aware recommender systems. In: Proceedings of the Second ACM Conference on Recommender Systems (RecSys 2008) (2008)
3. Adomavicius, G., Tuzhilin, A.: Context-aware recommender systems. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 217–253. Springer, US (2011). http://dx.doi.org/10.1007/978-0-387-85820-3_7
4. Apache Software Foundation: Apache storm (2015). http://storm.apache.org/

5. Baltrunas, L., Ricci, F.: Context-based splitting of item ratings in collaborative filtering. In: Proceedings of the Third ACM Conference on Recommender Systems. RecSys 2009, NY, USA, pp. 245–248 (2009). http://doi.acm.org/10.1145/1639714.1639759
6. Bogers, T., van den Bosch, A.: Comparing and evaluating information retrieval algorithms for news recommendation. In: Proceedings of the 2007 ACM Conference on Recommender Systems. RecSys 2007, NY, USA, pp. 141–144 (2007). http://doi.acm.org/10.1145/1297231.1297256
7. Brown, P.F., deSouza, P.V., Mercer, R.L., Pietra, V.J.D., Lai, J.C.: Class-based n-gram models of natural language. Comput. Linguist. **18**(4), 467–479 (1992). http://dl.acm.org/citation.cfm?id=176313.176316
8. Cantador, I., Bellogín, A., Castells, P.: News@hand: a semantic web approach to recommending news. In: Nejdl, W., Kay, J., Pu, P., Herder, E. (eds.) AH 2008. LNCS, vol. 5149, pp. 279–283. Springer, Heidelberg (2008). http://dx.doi.org/10.1007/978-3-540-70987-9_34
9. Cutting, D., Pedersen, J.: Optimization for dynamic inverted index maintenance. In: Proceedings of the 13th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR 1990, NY, USA, pp. 405–411 (1990). http://doi.acm.org/10.1145/96749.98245
10. De Pessemier, T., De Moor, K., Joseph, W., De Marez, L., Martens, L.: Quantifying subjective quality evaluations for mobile video watching in a semi-living lab context. IEEE Trans. Broadcast. **58**(4), 580–589 (2012)
11. De Pessemier, T., Coppens, S., Geebelen, K., Vleugels, C., Bannier, S., Mannens, E., Vanhecke, K., Martens, L.: Collaborative recommendations with content-based filters for cultural activities via a scalable event distribution platform. Multimedia Tools Appl. **58**(1), 167–213 (2012). http://dx.doi.org/10.1007/s11042-010-0715-8
12. De Pessemier, T., Dooms, S., Martens, L.: Context-aware recommendations through context and activity recognition in a mobile environment. Multimedia Tools Appl. **72**(3), 2925–2948 (2014). http://dx.doi.org/10.1007/s11042-013-1582-x
13. Elastic: Elasticsearch (2015). https://www.elastic.co/
14. Følstad, A.: Living labs for innovation and development of information and communication technology: A literature review. Electron. J. Organ. Virtualness **10**, 99–131 (2008)
15. Google: Google Hourly Trends (2015). http://www.google.com/trends/hottrends/atom/hourly
16. Han, B.J., Rho, S., Jun, S., Hwang, E.: Music emotion classification and context-based music recommendation. Multimedia Tools Appl. **47**(3), 433–460 (2010). http://dx.doi.org/10.1007/s11042-009-0332-6
17. Hatcher, E., Gospodnetic, O.: Lucene in action (in action series) (2004)
18. Hopfgartner, F., Kille, B., Lommatzsch, A., Plumbaum, T., Brodt, T., Heintz, T.: Benchmarking news recommendations in a living lab. In: Kanoulas, E., Lupu, M., Clough, P., Sanderson, M., Hall, M., Hanbury, A., Toms, E. (eds.) CLEF 2014. LNCS, vol. 8685, pp. 250–267. Springer, Heidelberg (2014). http://dx.doi.org/10.1007/978-3-319-11382-1_21
19. Katta: Lucune & more in the cloud (2015). http://katta.sourceforge.net/
20. Kenteris, M., Gavalas, D., Mpitziopoulos, A.: A mobile tourism recommender system. In: Proceedings of the IEEE Symposium on Computers and Communications. ISCC 2010, pp. 840–845. IEEE Computer Society, Washington, DC (2010)

21. Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gordon, L.R., Riedl, J.: Grouplens: applying collaborative filtering to usenet news. Commun. ACM **40**(3), 77–87 (1997). http://doi.acm.org/10.1145/245108.245126
22. Lee, H., Kim, J., Park, S.: Understanding collaborative filtering parameters for personalized recommendations in e-commerce. Electron. Commer. Res. **7**(3–4), 293–314 (2007). http://dx.doi.org/10.1007/s10660-007-9004-7
23. Li, L., Wang, D., Li, T., Knox, D., Padmanabhan, B.: Scene: a scalable two-stage personalized news recommendation system. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR 2011, NY, USA, pp. 125–134 (2011). http://doi.acm.org/10.1145/2009916.2009937
24. Lops, P., de Gemmis, M., Semeraro, G.: Content-based recommender systems: state of the art and trends. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 73–105. Springer, US (2011). http://dx.doi.org/10.1007/978-0-387-85820-3_3
25. Manning, C.D., Raghavan, P., Schütze, H., et al.: Introduction to Information Retrieval, vol. 1. Cambridge University Press, Cambridge (2008)
26. Papagelis, M., Plexousakis, D., Kutsuras, T.: Alleviating the sparsity problem of collaborative filtering using trust inferences. In: Herrmann, P., Issarny, V., Shiu, S.C.K. (eds.) iTrust 2005. LNCS, vol. 3477, pp. 224–239. Springer, Heidelberg (2005)
27. Phelan, O., McCarthy, K., Smyth, B.: Using twitter to recommend real-time topical news. In: Proceedings of the Third ACM Conference on Recommender Systems. RecSys 2009, NY, USA, pp. 385–388 (2009). http://doi.acm.org/10.1145/1639714.1639794
28. Porter, M.F.: Snowball: a language for stemming algorithms (2001). http://snowball.tartarus.org/
29. Reuters Institute for the Study of Journalism: Digital News Report (2014). http://www.digitalnewsreport.org/
30. Ricci, F.: Mobile recommender systems. Inf. Technol. Tourism **12**(3), 205–231 (2010)
31. Ricci, F.: Contextualizing recommendations. In: ACM RecSys Workshop on Context-Aware Recommender Systems (CARS 2012). In: Conjunction with the 6th ACM Conference on Recommender Systems (RecSys 2012). ACM, September 2012
32. Said, A., Bellogín, A., de Vries, A.: News recommendation in the wild: Cwi's recommendation algorithms in the NRS challenge. In: Proceedings of the 2013 International News Recommender Systems Workshop and Challenge. NRS, vol. 13 (2013)
33. Shani, G., Gunawardana, A.: Tutorial on application-oriented evaluation of recommendation systems. AI Commun. **26**(2), 225–236 (2013)
34. Shaphira, B., Rokach, L.: Recommender systems and search engines-two sides of the same coin? Slide Lecture (2012). http://medlib.tau.ac.il/teldan-2010/bracha.ppt
35. Telematica Instituut / Novay: Duine Framework (2009). http://duineframework.org/
36. The Apache Software Foundation: Apache Lucene (2015). https://lucene.apache.org/
37. The Apache Software Foundation: Apache Mahout (2015). http://mahout.apache.org/users/recommender/recommender-documentation.html
38. The Apache Software Foundation: Apache Solr (2015). http://lucene.apache.org/solr/

39. Reuters, T.: Open Calais (2008–2013). http://www.opencalais.com/
40. Weiss, A.S.: Exploring news apps and location-based services on the smartphone. Journalism Mass Commun. Q. **90**(3), 435–456 (2013)
41. Woodman, M.: Rome (2015). https://rometools.jira.com/wiki/display/ROME/Home
42. Yu, Z., Zhou, X., Zhang, D., Chin, C.Y., Wang, X., men, J.: Supporting context-aware media recommendations for smart phones. IEEE Pervasive Comput. **5**(3), 68–75 (2006)

Improving Predictions with an Ensemble of Linguistic Approach and Matrix Factorization

Manuela Angioni(✉), Maria Laura Clemente(✉), and Franco Tuveri(✉)

CRS4, Center of Advanced Studies, Research and Development in Sardinia,
Parco Scientifico e Tecnologico, Ed. 1, 09010 Pula, CA, Italy
{angioni,clem,tuveri}@crs4.it

Abstract. This paper extends a previous work done by the same authors [1] having the aim of improving the predictions coming from a matrix factorization based on latent factor models through an ensemble with the predictions obtained by an Opinion Mining methodology based on a linguistic approach. The experimental analysis was carried out on the Yelp business dataset, limited to the Restaurant category. An hypothesis of influence of the restaurant average rating on the number of stars given by the users is tested. An analysis of the meaning of some of the latent factors is shown.

Keywords: Opinion mining · Sentiment analysis · Text categorization · Collaborative filtering · Matrix factorization · Latent factors · Ensemble methods

1 Introduction

The availability of datasets, such as Yelp, being made of both textual reviews and star ratings, provide suitable input to the researchers in the field of recommender systems. Traditionally the predictions were obtained using above all the matrix of the ratings. Many researchers have enriched this input with some more information about the users or the items, but the improvement of the information with something written by the users which contains the actual opinion about an item represents an interesting challenge in order to obtain better results.

From the point of view of the Opinion Mining the most recent studies focus on detailing the user generated textual reviews in order to gain knowledge more closely reflecting the complexity of businesses, products and services contexts. While Recommendation Systems are currently mature technologies, the ones related to Opinion Mining are not yet able to provide reliable solutions beyond the research contexts.

In this paper, we propose an experimental analysis of a combination of Opinion Mining and Collaborative Filtering algorithms applied to the Yelp dataset of businesses. The analysis used this particular dataset in order to have both the textual reviews and the star ratings provided by the users. Opinion Mining was used to work on the textual reviews, while Collaborative Filtering worked on the star ratings.

As Pang and Lee [2] affirm at least one related set of studies claims that “the text of the reviews contains information that influences the behavior of the consumers, and that the numeric ratings alone cannot capture the information in the text” [3].

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 169–190, 2016.
DOI: 10.1007/978-3-319-30996-5_9

A fundamental aspect in the Yelp dataset is given by the fact that there is sometimes a discrepancy between the information written by a user in a textual review about a certain restaurant and the star rating.

This fact has been analyzed by means of a manual evaluation of the reviews, as described later in Sect. 4.3.

The rest of the paper is structured as follows: we provide related work about the combination of Opinion Mining and Collaborative Filtering in Sect. 2. Section 3 describes the Yelp dataset, the data extraction and the related issues. Section 4 describes the Opinion Mining analysis process while the prediction analysis methodology is described in Sect. 5. Section 6 describes the experimental setup and the results for the proposed approach. Lastly Sect. 7 reports conclusions and future works.

2 Related Works

Several studies have been written describing the combination of Opinion Mining and Collaborative Filtering.

Collaborative Filtering techniques aim to predict the preferences of users providing suggestions of further resources or entities that could be of interest.

The most popular commercial services on the web have demonstrated that user profiling is able to improve the revenues. For this reason the research in the field of user profiling and recommender systems have been developed at a very high speed in the last ten years. The most effective algorithms used by commercial services are defined as Collaborative Filtering, which can take as input a simple matrix of recommendations given by the users (the rows of the matrix) to the items (the columns). As stated in [4] the most important families of Collaborative Filtering algorithms are the neighborhood methods (deriving from k-Nearest Neighbour) and the latent factor models (which are based on the factorization of the matrix of recommendations).

Important results have been developed thanks to the 1 million dollars Netflix Prize, a competition started in 2006 by Netflix, the well-known dvd-rental company, for an algorithm able to increase by 10 % the accuracy of Cinematch, the algorithm used at the time by Netflix for movie recommendation.

The effect of this competition was to multiply the number of researchers involved in the topic, the number of related conferences, and most importantly the quality of the collaborative algorithms used by recommender systems. The million was won in 2009 by a combination of three different teams and their algorithms: item-based (a kind of kNN) [5], Restricted Boltzmann Machine (RBM) [6], and Biased Matrix Factorization [4].

While before the Netflix competition the item-based algorithms were considered the most effective for recommender systems, and in fact at the time they were used also by Amazon [7, 8], during the competition it has been demonstrated that the matrix factorization algorithms, working alone, were the most effective for this kind of problems [4, 9]. Although many types of algorithms can be used in the field of recommender systems, each of them has limitations, but these limitations change from one algorithm to another. It has been experimented that generally ensemble methodologies allow obtaining a blending prediction, which improve the ones coming from each of the algorithms singularly taken [10].

While algorithms based on user ratings produce interesting results, they do not consider qualitative information, like the actual opinion of a user about a resource and whether or not he/she actually would propose it to other users [11]. Moreover, explicitly given user ratings do not consider the different features of a resource and the weight that the users give to each of them, more or less unknowingly. On the other hand, feature-based Opinion Mining can be a very valuable resource to improve Collaborative Filtering performances, by adding qualitative information to explicit user ratings [12].

Levi et al. [13] proposed an interesting context-aware recommender system that uses Opinion Mining in order to analyze hotel reviews and to organize user tastes according to some users' preferences and to provide better recommendations in the cold-start phase. Another study related to a particular combination of Opinion Mining and Collaborative Filtering is [14], where the textual reviews are analyzed in order to be able to predict the level of interest of each user about different aspects of an item (representing a more detailed prediction than the single number of the predicted rating). An unusual combination of Collaborative Filtering and Opinion Mining is described in [15], where the output of an item-based collaborative filtering is further filtered by two different OM approaches.

A common problem to the user-generated reviews is usually related to the inconsistency in terms of length, content, treated aspects and usefulness because not every user writes about all the relevant aspects which characterize a business activity. For this reason relevant information would be disregarded, causing a lack of useful data in the input of the Opinion Mining algorithm.

3 Yelp Dataset

The dataset chosen for the presented activity is the one made available by the Yelp social network (http://www.yelp.com) for the RecSys Challenge 2013 "Yelp business rating prediction". An important feature of this particular data set is that it provides not only the star ratings (from 1 to 5 stars) assigned by the users to the business located in the Phoenix (AZ) metropolitan area, but also a textual review (along with many more information about users and business). This feature makes the Yelp dataset suitable for research in the fields of machine learning algorithms, which work on these two different types of information [16].

Only the training set for the competition was considered because in the test set the actual ratings were obviously missing.

The original training set was made of the following information:

- 229,907 reviews
- 43,873 users
- 11,537 business

For the aim of the presented activity the Restaurant category was chosen, which was the most represented in the original dataset. Restaurants constitute one of the most considered items in recommender systems [17–19] and also the Restaurant category in Yelp dataset [16, 20, 21]. It must be specified that more than one Yelp available dataset exists because there have been more than one competition providing each time a different version.

Actually each business in the data set has a list of categories, but in the one used for our activity, the word Restaurant is always present.

3.1 Data Extraction

We collected our data from the Yelp Dataset, considering only the users giving a number of reviews greater than 9, as more reliable. We target the most famous category in the set, Restaurants, and extracted 67,451 text reviews.

We did a spell check on the obtained reviews and then a transformation of the contracted forms of verbs in order to avoid introducing errors and to facilitate the syntactic parser activities. In fact, in the case of a sentence such as “We didn’t have a fridge in our room”, the parser was not able to correctly identify the contracted verb form didn’t. So, before parsing the text, some pre-processing steps related to the verbs were necessary, replacing the contracted forms into the long forms: *didn’t* became *did not*, *I’ve* became *I have*, *I’ll* became *I will*, and so on.

The reviews have been divided obtaining a number of 953,314 of sentences.

The phrase parser chunking process has been carried out by TreeTagger [22], annotating the sentences with part-of-speech tags and lemma information and identifying in each sentence its sub-constituents.

A Java class wraps the evaluation provided by TreeTagger and, analyzing the parts of speech, identifies the associations between nouns and their related information.

The “sentence analysis” (see Fig. 1) includes the result of the previous syntactic analysis, manages the feature extraction, and then uses the linguistic resources in order to calculate the polarity values of each sentence. As a result, the Sentence Analysis provides the categorization of each sentence of the reviews in order to distinguish between subjective and objective sentences, with or without orientation, and in particular in order to detect factual sentences having polarity value. In such a way, we consider only subjective sentences or factual sentences having polarity valence. The set of 953,314 sentences has been so reduced to about 394,000 subjective sentences bringing the entire set to the number of 50,705 reviews.

To achieve this task, we made use of SentiWordNet [23] a lexical resource that assigns to each synset of WordNet [24] three sentiment scores: positivity, negativity, objectivity, and we considered only the sentences containing adjectives with a polarity valence.

Opinion Mining analysis, as better described in Sect. 4, produced a set of rating predictions about the business activities to be compared with the Yelp ratings. Two researchers have manually evaluated a collection of 200 reviews to check the validity of the related ratings, using a common evaluation criterion. The comparison with the ratings manually assigned by the researchers, and described in detail in Sect. 4.3, allowed the evaluation of the performance of the Opinion Mining methodology.

Finally, the ratings coming from the Opinion Mining, combined with the user ratings (Yelp ratings), have been used by the ensemble algorithms.

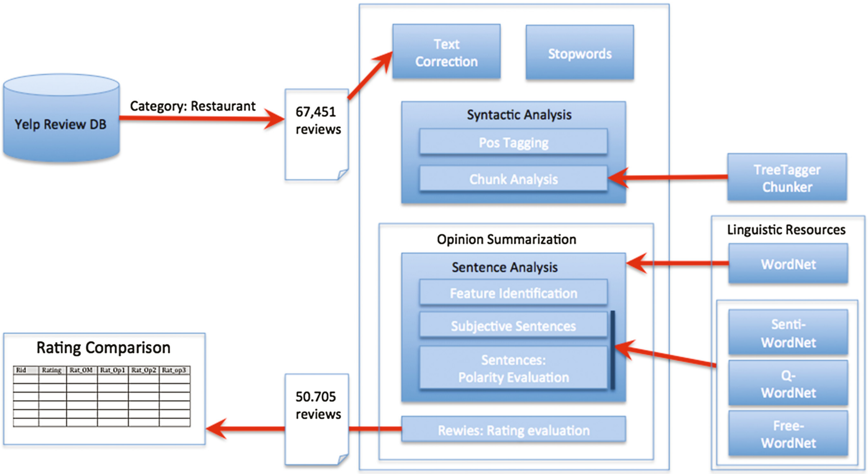


Fig. 1. The Opinion Mining analysis.

3.2 Issues Related to the Dataset

It is important to consider that since the textual reviews have been written without restrictions, they resulted affected by some limitations. Here after some examples of such limitations are explained.

The users could choose to talk about any of the aspects related to the restaurants (restaurant location, interior design, parking area, quality of service, quality/variety/amount of food, quality/variety of wine, prices, entertainment/live music, and intention to come back). This caused that some users talked about almost all of them, while many others limited their review to the quality of food.

Although Yelp applies an algorithm in order to filter out all the reviews posted by people related in some way to the business referenced by the review (such as the owner of the business, a relative of the owner or a person working there), it must be assumed that not all the reviews are spontaneous. This problem has caused also some lawsuits [25], and obviously Yelp will always be affected by phony reviews.

Jong [26] faces with the problem of the Yelp dataset in which the star ratings rarely provide the most objective or the fairest rating. In fact, most of the stars range from 3.5 to 4.5 stars with very few ratings below or above, resulting meaningless. In their study, some authors [27] put in evidence the differences of evaluation of distinct users (Michelle and Clif) who wrote about "Providence", a restaurant in LA area. Both users described their experience as very good using multiple positive words such as "perfection", "must go", "great treat", "tasted great", etc. However, the first user gave five stars to the restaurant whereas the second user gave only three stars.

Nevertheless most of the reviews can be considered reliable and this is the reason why Yelp has become so popular during the years.

4 Opinion Mining

Opinion Mining has been introduced in the presented activity to predict a business' rating based on textual reviews to be compared to the Yelp ratings.

As in [28], we propose a linguistic approach to Opinion Mining and, more in details, to the automatic extraction of feature terms by means of the syntactic and semantic analysis of textual resources. We focus on the analysis of the opinions through the processing of textual resources, the information extraction by means of the syntactic chunk analysis, and the evaluation of a semantic orientation.

The identification of adjectives and adverbs and the use of subjective lexical resources have a relevant role in this phase.

Many approaches to Opinion Mining are based on linguistic resources, lexicons or lists of words, used to express sentiments or opinions and are used for the identification of the polarity of words and their disambiguated meanings.

In the following Section the main tasks of the Opinion Mining process are described more in detail.

4.1 Feature Extraction

The feature extraction is a relevant task of the process. The term feature is used with the same sense given by [29] in their approach to Opinion Mining: given an object, that could be a service, a person, an event or an organization, the term feature is used to represent a component or an attribute describing that object.

We extracted the features by the textual reviews expressed by the users.

Considering that the domain is well known, the identification of the features for the Yelp reviews has been performed evaluating the nouns frequency in the text through a word counter. We first removed the stop words and then the cleaned text was tokenized obtaining as a result a collection of about 4000 words, including individual and compound words. We condensed this set by only considering words with a frequency greater than 100, in order to test the potential of the proposed approach, to be extended in a future work.

Finally, we identified the nouns as candidate features. The features were then manually validated and separated into six aspects: Food, Service, Staff, Ambience, Location and Price. As a result we obtained about 935 features. In a further development of this study we will also consider the verbs.

Although the reviews have been analyzed through the features they come with, we did not consider any criterion to evaluate them, putting at the same level each of the six aspects of the business. The evaluation of the reviews instead relies on the simple sum of the values of polarity associated with the terms they contain and on the identification of chunks, such as adverb + adjective, negations, and superlatives in their sentences.

For example, chunks can be considered as "not bad" or "very very good".

4.2 Feature Evaluation

Each sentence of the corpus of reviews was analysed and the association between features with adjectives and adverbs was found:

Table 1. Sample of feature, attribute and review relation.

Feature	reviewSid	Attribute	pos	Card
Staff	id112795s40	*Great*	JJ	2

In the above Table 1, the adjective *great* (JJ) is associated twice (cardinality = 2) to the feature *staff* belonging to the fortieth sentence of the review identified by the id 112795. The polarity of each attribute is calculated evaluating for all the synsets related to the term in WordNet the polarity associated to the synset in three different lexical resources: SentiWordNet, Q-WordNet and FreeWordNet.

SentiWordNet expands WordNet 2.0 and associates to each synset three numerical scores describing how much Objective, Positive and Negative are the terms related to that synset. This means that a synset may have nonzero scores for all the three categories.

Q-WordNet [30] is a lexical resource consisting of WordNet senses automatically classified by Positive and Negative polarity. Polarity evaluation has been used to decide whether a textual content is associated to a positive or negative connotation.

FreeWordNet [31] is another lexical database of synsets defined as extension of a subset of adjectives and adverbs of WordNet. Each synset has been enriched with a set of properties concerning the polarity and other properties according to a set of attributes identified by their association with nouns and verbs and chosen on the basis of their frequency of use in the language.

The Opinion Mining system has produced a set of rating predictions affected by the choice of the lexical resource. In some cases the three resources have produced discrepancies of polarity related to the same synset. For this reason we chose to consider the average of the three values obtained from the assessment made by the Opinion Mining system with the three resources.

4.3 Reviews Analysis and Algorithm Evaluation

Regarding the analysis of the reviews we faced with the issue related to the representation of the rating values given by the Opinion Mining system in order to compare them with the ratings of Yelp.

In fact, the values produced by the Opinion Mining system were not directly comparable with the ratings of Yelp, because they were distributed in a range between −25 and 36. Also the distribution of the ratings obtained was totally different if compared with the Yelp one. Several transformations were possible, here after we describe the one we chose.

The values have been initially linearly scaled on a rating system that ranges between 0 and 5. As a first step, we chose the transformation, which produced a distribution of the Opinion Mining values similar to the distribution of the Yelp ratings. The introduction of the thresholds, shown in Table 2, gave us the opportunity to assign to the rating classes a number of reviews similar to the Yelp ones. These values produced the distribution depicted in Fig. 2.

Table 2. The thresholds applied.

Thresholds	
Id	Range
T_0	x < 1.2
T_2	1.2 <= x < 2.2
T_3	2.2 <= x < 3.2
T_4	3.2 <= x < 4.2
T_5	x > 4.2

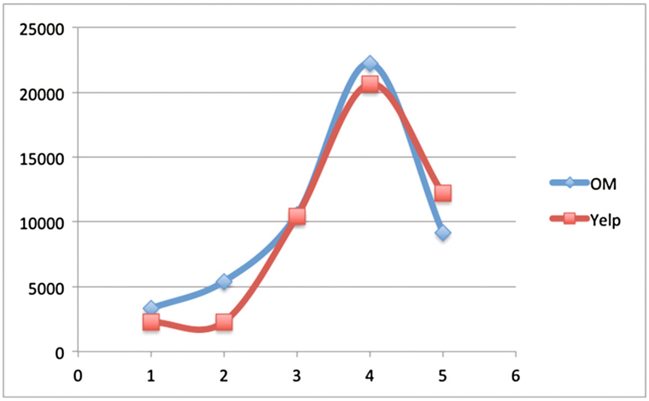


Fig. 2. Rating distribution.

As already mentioned in Sect. 1, in order to evaluate the performance of the Opinion Mining algorithm, two researchers (the Analysts) have manually evaluated a collection of 200 reviews. A preliminary tuning phase was carried out on a limited number of reviews in order to agree on a common evaluation criterion.

The choice was based on the length of the text, assuming that longer reviews contain more information.

The methodology of evaluation of the reviews was based on a set of sub-aspects of the original 6 aspects, previously introduced in Sect. 4.1, plus the "intention to come back" to the restaurant, as listed in Table 3. Each sub-aspect was independently evaluated by the Analysts with values ranging between 0 and 5, while the aspects disregarded by the author of the review (the Reviewer) were penalized by a value of 0.2. We wanted to penalize the sub-aspects not covered in the review because, although they were not negatively considered, their absence from the description meant that they had not positively impressed the customer anyway.

Table 3. The ratings as evaluated by the Analysts.

	Aspects	Res. 1	Res. 2	Avg. Rate
1	Food quality	3.0	1.0	2.0
2	Food quantity	4.0	4.0	4.0
3	Food variety	3.0	3.0	3.0
4	Food beverages	−0.2	−0.2	−0.2
5	Food desserts	−0.2	−0.2	−0.2
6	Service	2.4	2.8	2.6
7	Staff	2.5	2.0	2.25
8	Ambience	0.2	0.1	0.15
9	Location-bar	−0.2	−0.2	−0.2
10	Location-parking	−0.2	−0.2	−0.2
11	Price	−0.2	−0.2	−0.2
12	Come back	4.0	4.0	4.0
	Total	**2.58**	**2.27**	**2.43**

The rating of each review has been calculated as the sum of m partial ratings r_i, corresponding to the 12 aspects and sub-aspects present in the user's textual review. This rating was penalized by a constant value of −0.2 for each of the n features not present, and divided by m (see Eq. 1), where $m + n = 12$. In case the resulting total rating R was less than zero it was set equal to 0.

$$R = \frac{\sum_{i=1}^{m} ri - 0.2 * n}{m} \quad 0 < r_i \leq 5 \tag{1}$$

The average rating provided by the Analysts was finally used in order to evaluate the algorithm in terms of Precision (P), Recall (R) and F1-score.

Figure 3 illustrates the evaluation of the Opinion Mining system according to the values of threshold shown in Table 2 in terms of micro and macro averaging.

During the Opinion Mining algorithm evaluation, it was noticed that the star rating not always appeared in line with the content of the review. These inconsistencies were shown up throughout the manual analysis of the reviews, and evidenced by the discrepancies between the star ratings assigned by the Yelp users and the manual rating given by the researchers. The nature itself of the star rating does not depict a detailed experience or does not express emotions and feelings, which are instead described by the several aspects covered by textual reviews.

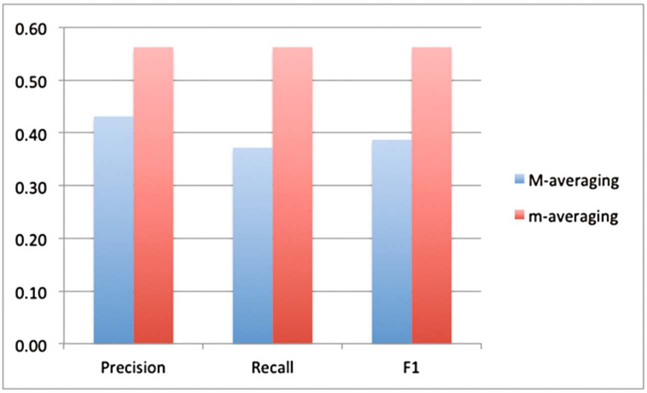


Fig. 3. Opinion Mining system evaluation.

Let us use an example to illustrate this concept considering a specific review having 4 as star rating, while the Analysts gave respectively a rating of 2.58 and 2.27, with an average value of 2.43 as shown in Table 3.

The following text has been extracted from the review and analyzed according to the aforementioned 11 aspects and sub-aspects: food-quality, food-quantity, food-variety, food-beverages, food-desserts, service, staff, ambience (atmosphere), location-parking, location-bar, price, plus the intention to come back.

Here after we discuss some of them in detail.

Food: Quality, Quantity, Variety

> *"I had the Chicken Tikka Masala and my friend had the Chicken Pot Pie - both were delicious! I was super impressed with the breadth of the pasty 'stuffings' and got very excited to see they had over 40 options! I want to go back and try them all. The yogurt served was perfect. the chicken was plentiful and well seasoned… their fillings are both inventive, somewhat unique and really entice me to come back."*

- *"I appreciated that their pastys are a good size…their pastys were yummy"*
- *"I found a piece of red thick rubber-band in my soup."*

Service

- *"Yes, this was not a short lunch… but it was also not a long lunch. Food took a bit long but we also were expecting 'custom-baked' pasty."*
- *"their service was excellent"*

Staff

- *"Our server was quick enough to bring us drinks and my soup order, etc., to hold us over."*
- *"Our server guy was outstanding - friendly in a genuine way, prompt, kept checking on us, and had very good customer service skills."*

- *"I did note one of the food preppers talking on his cell phone while he was handling food. I thought that was gross..."*
- *(about the "red thick rubber-band in my soup") "Yeah, not excited about that find, but I did appreciate that our server immediately apologized... fixing a problem immediately and correctly - kudos for him for a great response and showing good customer service."*
- *"Can understand that 'shit happens' sometimes but it's the aftermath of how you treat the customer that found the rubber band in their soup that matters"*

Ambience (Atmosphere)

- *"Smells: Yes, but we deduced it was the cabbage."*
- *"It needs a good cleaning ... there is a distinctly strong smell"*
- *"Those pew cushions were nasty and the wax should be scraped off and menus cleaned up"*
- *"The pews had dirty cushions on them...the candle wax was all over the table/menus, the menus had other grime on there...this place REALLY needs to quit the slacking and clean this place up."*

Intention to Come Back

- *"I'll be back!"*

Not Mentioned

Location-Bar; Location-Parking; Price; Food-Beverages; Food-Desserts;

The obtained ratings are the algebraic sums of each sub-aspect divided by the number of mentioned sub-aspects. The average rating is used, as said, in order to evaluate the algorithm.

You can notice that, even if the Reviewer said that he would go back there, the lack of cleanness, the strong smell, the slow service, the presence of a rubber in the food, are so negative elements that it is incredible that he/she could assign a high rating.

As said before, there is a big discrepancy between the ratings given by the Reviewer, 4 stars, due to the very good evaluation given to the Food aspects, while the Analysts gave an average value of 2.43.

Initially we considered this discrepancy in the evaluation as an evident error in the assignment of the rating by the Reviewer. But we are now more likely to think that the difference between the two evaluations is caused by different gastronomical habits and, more in general, by different profiles.

This fact highlights the critical issues related to the definition of an objective criterion of evaluation of the reviews as described below.

4.4 Review Analysis Issues

During the described analysis of the reviews we found some discrepancies between the evaluation of the reviews provided by the two Analysts compared with the corresponding values provided by the Yelp Reviewers and those obtained by the Opinion Mining system. This is due, in our opinion, to the differences of "taste" between the two Italian

Analysts and the American Reviewers, i.e. differences in food culture, "culture" in the broader sense, economic opportunities, and competences.

These differences definitely affect the assessments of products and services. The various online rating systems often seem to not take them into account.

We are planning to adopt a user profiling system, applied to the user-related data, able to avoid the differences expressed by the Analysts and the Reviewers in the evaluation of the reviews and in the rating expressed. It is thus more likely that Analysts and Reviewers having similar profiles express similar ratings about the same reviews.

The introduction of 12 weighted aspects implemented in the code, as presented in the paper, while certainly help to define a calibrated and consistent method of analysis of the reviews, does not solve alone the cited discrepancies because does not take into account the priorities and the preferences expressed by the Reviewers according to their profiles.

The crux of the problem is given by the identification of the users' profiles. The dataset provided by Yelp contains information about the users registered as Reviewers, such as the number of the reviews, the average stars, the vote type and the number of the friends. It could be possible doing a search by hand of the users having tastes similar to the Analysts, but it is definitely better to evaluate the preferences of both Reviewers and Analysts through a prediction system. The system by identifying the set of users having the same preferences of the Analysts can also identify and suggest the set of businesses to be analyzed.

5 Prediction Analysis

Predictions for the test set were computed by means of three different algorithms singularly run:

1. The Baseline algorithm made of average ratings, described in Sect. 5.2
2. Opinion Mining, described in Sect. 4
3. Biased Matrix Factorization, described in Sect. 5.3

Then a RMSE was calculated for each different set of predictions singularly taken and compared with an ensemble of algorithms 2 and 3, as described in Sect. 5.4.

5.1 Thresholds

As already mentioned, during the experimental activity, it was evident that the output in terms of predictions coming from the Opinion Mining algorithm were not always aligned with the star ratings. In particular while working at the activity related to the manual check of the Opinion Mining predictions (described in Sect. 4.3) against the actual ratings, most of the times the star rating overestimated what the same user expressed in words.

This inconsistency between the textual review and the star rating appeared to be an interesting behavior and a possible explanation brought to think that maybe many users were influenced by the average rating of the business (which in the Yelp web site is obviously well shown).

In order to test this hypothesis, an experimental analysis was carried out applying some coefficients to the predictions obtained by the Opinion Mining; during this activity the thresholds already used during the calculation of the predictions were applied (see Table 2): T0 = 1.2, T1 = 2.2, T2 = 3.2, and T3 = 4.2. In particular, the predictions have been multiplied by a coefficient, but only under the condition that the average business rating (BRT) was greater than a certain value depending on the threshold.

A schema of the thresholds is shown in Fig. 6.

5.2 Baseline Algorithm

The baseline algorithm chosen for the activity was run using a 5 fold cross-validation method and based on the following averages calculated for each user and business related to a rating:

- average rating of user *i* ($\mathrm{avg\_u_i}$), the average of all the ratings in the training set given by user *i* ($\mathrm{u_i}$)
- business average ($\mathrm{avg\_b_j}$), the average of all the ratings in the training set received by the business *j* ($\mathrm{b_j}$)
- global average (global_avg), the average of all the ratings in the training set (3.6891).

In particular, when both the user and the business were present in the training set, the prediction $\mathrm{p(u_i, b_j)}$ of the star rating that the user *i* ($\mathrm{u_i}$) could give to a business *j* ($\mathrm{b_j}$) was calculated as the following weighted average:

$$\mathrm{p}\left(\mathrm{u_i}, \mathrm{b_j}\right) = \left(\mathrm{avg\_u_i} * \mathrm{w_1} + \mathrm{avg\_b_j} * \mathrm{w_2}\right)/2 \tag{2}$$

where $\mathrm{w_1}$ and $\mathrm{w_2}$ are the weights, described in Sect. 6.1.

When only the user was known in the training set the prediction was calculated as:

$$\mathrm{p}\left(\mathrm{u_i}, \mathrm{b_j}\right) = \left(\mathrm{avg\_u_i} * \mathrm{w_1} + \mathrm{global\_avg} * \mathrm{w_2}\right)/2 \tag{3}$$

When only the item was known in the training set:

$$\mathrm{p}\left(\mathrm{u_i}, \mathrm{b_j}\right) = \left(\mathrm{avg\_b_j} * \mathrm{w_1} + \mathrm{global\_avg} * \mathrm{w_2}\right)/2 \tag{4}$$

And lastly, when both the user and the business were not present in the training data (the famous cold start problem), the prediction was set equal to the global average.

5.3 Biased Matrix Factorization

As already recalled in the Introduction, an effective latent factor model is represented by the biased matrix factorization, which is based on the fact that each review and each rate is influenced by a certain number of latent factors not known. These factors can be inferred by the algorithm, and some hypothesis about the actual meaning of some of them is proposed later on, in Sect. 6.5.

A summary of the algorithm applied for the presented activity is given here after (more details can be found in [4]). The basis of the algorithm is the decomposition of the original matrix of the R ratings given by M users to N items (in this case, the restaurants) into two different matrices: P of dimension M × K, and Q of dimensions N × K, where K is the number of features which must be decided beforehand (see Fig. 4). In particular, in the P matrix related to the items, for each item (for each restaurant) we have a vector made of the K latent factors; each of these K values represents the measure of the relation of the item i to each of the K latent factor. The same is valid for the users.

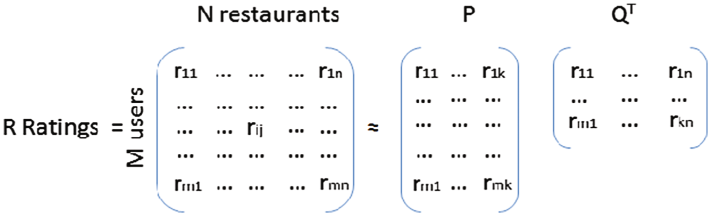


Fig. 4. The schema of matrix factorization.

By multiplying $Q^T * P$ it is possible to approximate the ratings in the original matrix R as far as the known rates are concerned and moreover, all the other unknown values can be predicted, so an unknown rate r_{ij} that the user *i* would give to a restaurant *j* can be predicted as:

$$r_{ij} = q_j^T . p_i \tag{5}$$

From this equation it is possible to obtain the error which is given by the difference of the known ratings (which in the training set are known) from the predicted ones. A regularization constant β is used to avoid the over fitting (when a model which is able to produce predicted values very closed to the known ratings, actually is not able to generalize in order to obtain good predictions of the unknown values). The equation to be minimized is:

$$e^2 = \sum\nolimits_{k=1}^{K} (r_{ij} - q_j^T p)^2 + \beta\left(\left|\left|q_j\right|\right|^2 + ||p_i||^2\right) \tag{6}$$

In the presented activity the learning algorithm used to minimize this error is the Stochastic Gradient Descent: at each iteration a step proportional to a learning rate α is made toward the actual rate (which in the training set is known).

In order to take consideration of the different personality of the users in giving the ratings and also some inexplicable reasons of an item (in this case a restaurant *r*) popularity, the user and item biased are included in the basic formula of Matrix Factorization. A very simple expression of the biases is obtained by a correction of the dataset average rating μ with the user bias b_u and the restaurant bias b_r:

$$b_{ur} = \mu + b_u + b_r \tag{7}$$

This quantity is intended to be added to the Eq. 6 of the error.

So while the original matrix was typically sparse (not all the users rated all the restaurants), once the model has been trained, it is possible to have a prediction for any user in relation to any restaurant.

During the presented experimental activity the Mahout Taste library [32, 33] was used; in particular, the Stochastic Gradient Descent Factorizer (used as learning algorithm) is known to be an implementation of the Biased Matrix Factorization algorithm described in [4, 34]. The Singular Value Decomposition was used as Recommender.

The research activity considered different values of the following parameters through a 5-fold cross validation: number of features, number of iterations, learning rate, regularization constant, random noise, and learning rate decay.

The cases of unknown restaurants or unknown users were dealt with averages analogue to the ones used in the baseline algorithm already described in Sect. 5.2.

5.4 Ensemble Methods

As already mentioned in Sect. 2, ensemble methodologies allow the improvement of the results coming by multiple algorithms because typically they are weak in different ways and their combinations produce a more robust and generalizable solution.

For this reason, in the presented activity some ensemble methods were applied in order to combine the output in terms of predictions coming from the Opinion Mining and the Biased Matrix Factorization algorithms.

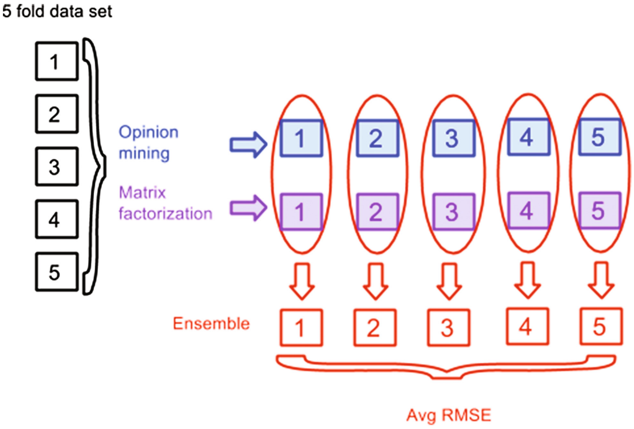


Fig. 5. Scheme of cross validation and ensemble.

The regressions used during the activity are the followings:

- Linear Regression, which finds the best fitting line minimizing the sum of the squared errors of the predictions;
- Ridge regression, which differs from a linear regression because it applies a ‘ridge’ penalty to reduce the variance of the values;

- Gradient Boosting Regression Trees (GBRT), which uses decision trees as weak learners and is extensively used also for its predictive power.

In Fig. 5 the scheme of the ensemble methodology is shown. The training dataset has been split into 5 folds to be used as input of both Opinion Mining and Collaborative Filtering algorithms. At each step, one of the folds is kept apart as test set, while the others are used together as training set.

The ensemble methods were used to merge these 5 output and so 5 different RMSE values have been obtained. Then, the average of these 5 values has been considered as the final result.

6 Results in Terms of RMSE

The Root Mean Squared Error (RMSE) is a very common way to evaluate the quality of predictions for recommender systems and in fact it is greatly used in competitions and related leader boards (such as Netflix prize, available at www.netflixprize.com/leaderboard, RecSys2013 available at www.kaggle.com/c/yelp-recsys-2013/leaderboard, etc.). It amplifies large errors and provides the advantage of concentrating the result in a single parameter.

Since the errors are squared, negative and positive errors do not cancel each other. Smaller RMSE values correspond to better results.

$$RMSE = \sqrt[2]{\frac{\sum_{i=1}^{N}(\mathrm{P}_i - r_i)^2}{N}} \tag{8}$$

In the above formula, P_i is the prediction for each of the N reviews in the data used as test set, while r_i is the actual rating (since the data set used was a training set, the actual ratings were known).

In the presented study the RMSE was used to evaluate the quality of the predictions coming from Baseline, Opinion Mining, and Biased Matrix Factorization algorithms. The same evaluation was used to analyze the ensemble of Opinion Mining with Biased Matrix Factorization as well.

6.1 Baseline

The algorithm used as Baseline, described in Sect. 5.2, was initially run giving the same weights to the user average ($\mathrm{avg\_u_i}$), the business average ($\mathrm{avg\_b_j}$) and the global average (global_avg). Further experimental analysis brought to the choice of penalizing the user average contribution, and in the end the best values were $\mathrm{w}_1 = 0.4$ and $\mathrm{w}_2 = 0.6$.

Since most of the actual ratings of the dataset are included in the range between 3.5 and 4.5 stars, the baseline could produce a RMSE value of 1.0259, which was hard to be outperformed for this particular dataset and for this reason represented a good reference point for the study.

6.2 Opinion Mining

In terms of RMSE the predictions, which were originally output of the Opinion Mining methodology, did not outperform the Baseline predictions, giving a value of 1.25011. But as already stated, a more attentive analysis of this result induced to work with the set of thresholds in order to apply some coefficients to take into account the influence under which most users had expressed the star rating, due to its aforementioned inconsistency with the content of the textual reviews. The set of thresholds applied and their values have been schemed in Fig. 6.

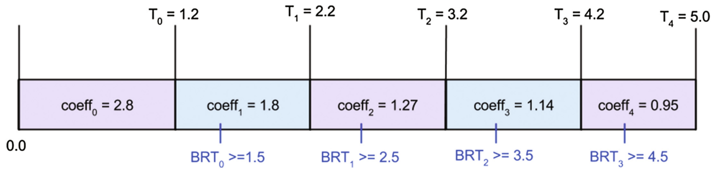


Fig. 6. The thresholds used on OM predictions.

The application of these thresholds to the Opinion Mining (OM+T) caused a change in almost the 50 % of the original predictions. The new value of RMSE was 1.00548, which greatly outperformed the baseline, but unexpectedly did slightly better than the BMF algorithm as well.

6.3 Biased Matrix Factorization

As already explained in Sect. 5.3, the experimental analysis on the Biased Matrix Factorization was carried out through a 5-fold cross validation and involved many different configurations depending on the values given to the parameters taken as input by the factorizer (the RatingSGDFactorizer).

An analysis of the first four latent factors related to the users and to the restaurants has been carried out, as described in Sect. 6.5.

The parameters which provided best results in terms of RMSE are the followings:

- Number of features = 14
- Number of iterations = 75
- Learning rate = 0.0025
- Regularization constant = 0.02
- Random noise = 0.01 (the default value)
- Learning rate decay = 1.0 (the default value)

With this configuration the resulting RMSE was 1.00859.

6.4 Ensemble

The Gradient Boosting Regression Tree, the Linear Regression, and the Ridge Regression produced better RMSEs than each of the predictive algorithms, singularly taken.

In particular, the best value of RMSE with the different ensemble algorithms was obtained by the GBRT that, as expected, was also the better result of all the presented experimental analysis, as summarized in Table 4.

Table 4. Summary of the best RMSEs obtained.

Alg.	Baseline	BMF	OM	OM+T	Ens. GBRT
RMSE	1.02593	1.00859	1.25011	1.00548	0.98874

6.5 Analysis of Some Latent Factors

This section describes an analysis of the first four latent factors related to the users and to the restaurants. A development of this analysis could be carried out as future work.

The values of restaurant latent factors and user latent factors are output of the Stochastic Gradient Descent Factorizer, as explained in Sect. 5.3. The aim was to try to understand the actual meaning of these values. In order to do this the first two latent factors (related to both the users and the restaurants) were mapped in two dimensions (see Fig. 7); the same was done with the values of the third and fourth latent factors (see Fig. 8).

First of all the number of users was reduced by including only the 53 who wrote at least 80 reviews, while the number of restaurants was limited to the 75 which had received more than 200 reviews (the greater the number of reviews, the better the evaluation of the main features of each restaurant).

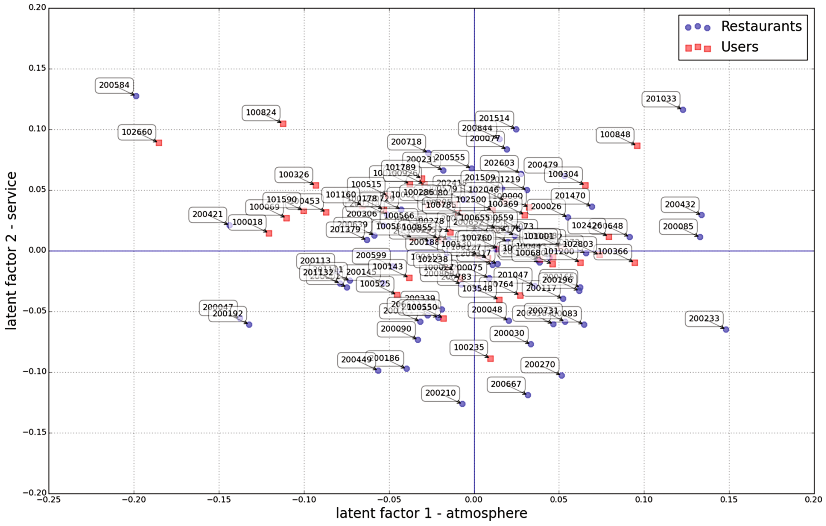


Fig. 7. Mapping of the first 2 latent factors for users and restaurants.

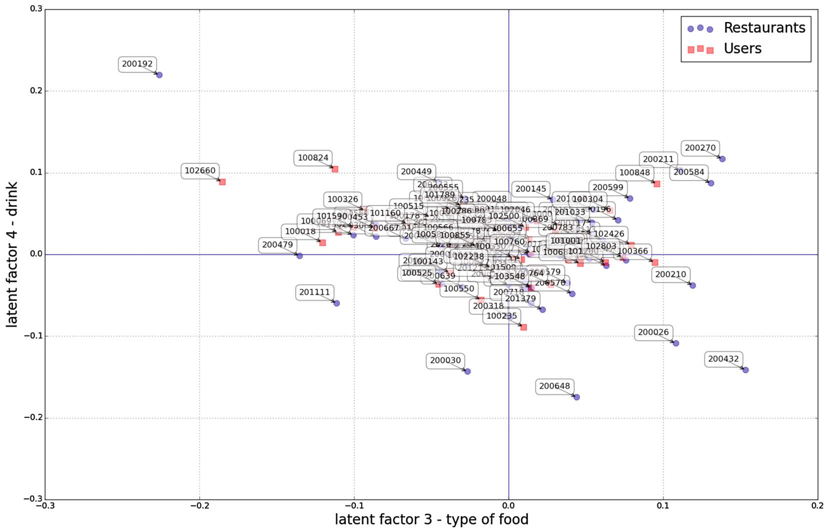


Fig. 8. Mapping of the third and fourth latent factors for users and restaurants.

Then a manual analysis of the reviews was performed, keeping into account the same aspects explained in Sect. 4.3. Being based on manual observations of a limited number of elements, it must be said that the output of this analysis must be considered as a possible explanation of the meaning of the latent factors, but we think it is a valid one.

In relation to the first two latent factors, from the analysis of the reviews it was possible to notice that on the bottom-left resulted many restaurants characterized by rude staff and/or exceedingly long waiting time (until three hours in one case), although the quality of food is often good. Opposite to this, on the far right, resulted some places with an amazing atmosphere given by very friendly staff, excellent service, and often also music (live or not). Along the way from left to right, there are restaurants with an increasing amount of these aspects, bringing to suppose that the first latent factor related to restaurants expresses the overall atmosphere influenced also by the music and by the mood of the other customers. While the second latent factor (the vertical direction) could be related to the service, including in it the parking and the level of the restrooms (although this is more difficult to be proved because strangely enough very few people talked about restrooms). The analysis has also shown some interesting groups of similar restaurants; for example three steakhouse restaurants resulted mapped very closed to each other, on the bottom-right part of the drawing; all of them characterized by the presence of a bar, good food, and good service, along with a modern and elegant interior. Opposite to them, a group of not very attractive restaurants which do not exceed in any aspect.

As far as users are concerned, their position in the same plotting seem to support the considerations about the restaurants, and few examples are reported here to sum up this interpretation: in the top right quadrant of the map, there is a user who wrote many words related to the friendliness of the service, along with quality of food, the drinks. In the same quadrant, but much more close to the origin of the coordinates, there is a person who talks a lot about food quality, about drinks (presence of the bar), and also about atmosphere, music, but does not mind about the service. In the bottom-right quadrant, there is a person who talks a lot about music, atmosphere and drinks, and also about service, although not very much about quality of food. In the bottom-left quadrant, almost at the same ordinate of the previous person, there is a guy who talks a lot about quality of food and drink, but seems not to consider all the other aspects, including the service (which in fact, in this part of the diagram is worse than everywhere else).

The mapping of the third and fourth latent factors in two dimensions (see Fig. 8) brought to some further considerations. In the top-right quadrant there are restaurants characterized by spicy/peppery, or rich in chili food; on the central bottom part of the diagram, there are more steakhouses (including the aforementioned three which were close to each other in the map of the first two latent factors) and barbeque restaurants. On the top-left quadrant there are mostly restaurants with peculiar features related to drink: a pizzeria which is actually more famous for the bar and its variety of wines, a Latin-American restaurant famous for its sangria, a restaurant offering a great variety of beers, and so on. So, the analysis of Fig. 7 brings to associate the different types of food to the abscissas and the drinks to the ordinate axis.

7 Conclusions and Future Work

Most existing Recommender Systems are based only on users' overall ratings about items, but do not consider and do not work on the opinions expressed by the users about the different aspects of an item. As a result, the rate does not wholly summarize the opinion of the users, maybe ignoring important information.

In order to overcome this problem a research activity about possible combinations of Opinion Mining and Collaborative Filtering has been carried out.

The encouraging results obtained in terms of RMSE seem to confirm the hypothesized influence of the average business rates on the users in the choice of the number of stars to set as rate.

We would like to further develop this study in many ways: regarding the evaluation of the textual reviews, we would like to apply an algorithm able to calculate the ratings through the estimation of the aspects described in Sect. 4.3; regarding the Opinion Mining, we would like to improve the syntactic and semantic analysis; in relation to the Collaborative Filtering, we are interested to deepen the analysis of the latent factors in order to find their correlations with the aspects and sub-aspects characterizing the businesses.

Acknowledgements. This study is part of a POR FESR 2007-2013 project co-funded by the Autonomous Region of Sardinia: Comunimatica (P.I.A. n. 205 co-funded according to the DGR 39/3 of 10/11/2012).

References

1. Angioni, M., Clemente, M.L., Tuveri, F.: Combining opinion mining with collaborative filtering. In: Proceedings of WEBIST 2015, 11th International Conference on Web Information Systems and Technologies, Lisbon (2015)
2. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Found. Trends Inf. Retrieval **2**(1–2), 1–135 (2008). doi:10.1561/1500000011
3. Ghose, A., Ipeirotis, P.G.: Designing novel review ranking systems: predicting usefulness and impact of reviews. In: International Conference on Electronic Commerce (ICEC) (2007)
4. Koren, Y., Bell, R., Volinsky, C.: Matrix factorization techniques for recommender systems. Computer **42**(8), 42–49 (2009). IEEE Computer Society
5. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Item-based collaborative filtering recommendation algorithms. In: Proceedings of IEEE Internet Computing, 10th International World Wide Web Conference (2001)
6. Hinton, G.E.: A practical guide to training restricted boltzmann machines. In: Montavon, G., Orr, G.B., Müller, K.-R. (eds.) Neural Networks: Tricks of the Trade. LNCS, vol. 7700, 2nd edn, pp. 599–619. Springer, Heidelberg (2012)
7. Linden, G., Smith, B., York, J.: Amazon.com recommendations. In: IEEE Internet Computing, vol. 07, no. 1, pp. 76–80 (2003)
8. Clemente, M.L.: Experimental results on item-based algorithms for independent domain collaborative filtering. In: Proceedings of AXMEDIS 2008, pp. 87–92. IEEE Computer Society (2008)
9. Tosher, A., Jahrer, M., Bell, R.M.: The BigChaos solution to the Netflix grand prize, Netflix Prize Documentation (2009)
10. Jahrer, M., Töscher, A., Legenstein, R.: Combining Predictions for Accurate Recommender Systems. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 693–702. ACM (2010)
11. Koukourikos, A., Stoisis, G., Karampiperis, P.: Sentiment analysis: a tool for rating attribution to content in recommender systems. In: 2nd Workshop on Recommender Systems for Technology Enhances Learning (RecSysTEL 2012). Saarbrucken, Germany (2012)
12. Quadrana, M.: E-tourism recommender systems (2013). http://hdl.handle.net/10589/84901
13. Levi, A., Mokryn, O., Diot, C., Taft, N.: Finding a needle in a haystack of reviews: cold start context-based hotel recommender system. In: Proceedings of the Sixth ACM Conference on Recommender Systems, pp. 115–122. ACM (2012)
14. Wu, Y., Ester, M.: FLAME: a probabilistic model combining aspect based opinion mining and collaborative filtering. In: WSDM 2015, Shanghai, China (2015)
15. Singh, V.K., Mukherjee, M., Mehta, G.K.: Combining collaborative filtering and sentiment classification for improved movie recommendations. In: Sombattheera, C., Agarwal, A., Udgata, S.K., Lavangnananda, K. (eds.) MIWAI 2011. LNCS, vol. 7080, pp. 38–50. Springer, Heidelberg (2011)
16. Huang, J., Rogers, S., Joo, E.: Improving restaurants by extracting subtopics from yelp reviews. SOCIAL MEDIA EXPO (2014). https://www.ideals.illinois.edu/bitstream/handle/2142/48832/Huang-iConference2014-SocialMediaExpo.pdf
17. Burke, R.: Hybrid recommender systems: survey and experiments. User Model. User-Adap. Inter. **12**(3), 331–370 (2002)
18. Ganu, G., Kakodkar, Y., Marian, A.: Improving the quality of predictions using textual information in online user reviews. Inf. Syst. (2012). doi:10.1016/j.is.2012.03.001

19. Ganu, G., Elhadad, N., Marian, A.: Beyond the stars: improving rating predictions using review text content. In: Twelfth International Workshop on the Web and Databases (WebDB 2009), Providence, Rhode Island, USA (2009)
20. Trevisiol, M., Chiarandini, L., Baeza-Yates, R.: Buon Appetito - Recommending Personalized menus (2014)
21. Govindarajan, M.: Sentiment analysis of restaurant reviews using hybrid classification method. Int. J. Soft Comput. Artif. Intell. **2**, 17–23 (2014)
22. Schmid, H.: Probabilistic part-of-speech tagging using decision trees. In: Proceedings of the International Conference on New Methods in Language Processing, pp. 44–49 (1994)
23. Baccianella, S., Esuli, A., Sebastiani, F.: SentiWordNet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. In: LREC 2010, 7th International Conference on Language Resources and Evaluation, Malta, pp. 2200–2204 (2010)
24. Miller, G.: WordNet: an Electronic Lexical Database. Bradford Books, Cambridge (1998)
25. Clark, P.: Yelp's Newest Weapon Against Fake Reviews: Lawsuits (2013). http://www.businessweek.com/articles/2013-09-09/yelps-newest-weapon-against-fake-reviews-lawsuits
26. Jong, J.: Predicting Rating with Sentiment Analysis (2011). http://cs229.stanford.edu/proj2011/Jong-%20PredictingRatingwithSentimentAnalysis.pdf
27. Mingming, F., Khademi, M.: Predicting a Business Star in Yelp from Its Reviews Text Alone. ArXiv e-prints: arXiv:1401.0864 (2014)
28. Benamara, F., Cesarano, C., Picariello, A., Reforgiato, D., Subrahmanian, V.S.: Sentiment analysis: adjectives and adverbs are better than adjectives alone. In: Proceedings of ICWSM 2007, International Conference on Weblogs and Social Media, pp. 203–206 (2007)
29. Ding, X., Liu, B., Yu, P.S.: A holistic lexicon-based approach to opinion mining. In: International Conference on Web Search and Web Data Mining, ACM, NY, USA (2008)
30. Agerri, R., Garcia-Serrano, A.: Q-WordNet: extracting polarity from WordNet senses. In: 7th International Conference on Language Resources and Evaluation (LREC2010), Malta (2010)
31. Tuveri, F., Angioni, M.: A linguistic approach to feature extraction based on a lexical database of the properties of adjectives and adverbs. In: Global WordNet Conference (GWC 2012), Matsue, Japan (2012)
32. Owen, S., Anil, R., Dunning, T., Friedman, E.: Mahout in Action. Manning Publications Co., Shelter Island (2011). ISBN 9781935182689
33. Shelter, S., Owen, S.: Collaborative Filtering with Apache Mahout. In: RecSys Challenge (2012)
34. Paterek, A.: Improving regularized singular value decomposition for collaborative filtering. In: Proceedings of KDDCup and Workshop, pp. 39–42. ACM Press (2007)

When Users with Preferences Different from Others Get Inaccurate Recommendations

Benjamin Gras(✉), Armelle Brun, and Anne Boyer

LORIA - Université de Lorraine, Vandoeuvre-lès-nancy 54506, France
{benjamin.gras,armelle.brun,anne.boyer}@loria.fr
http://www.loria.fr

Abstract. The social approach in recommender systems relies on the hypothesis that preferences are coherent between users. To recommend a user u some resources, this approach exploits the preferences of other users who have preferences similar to those of u. Although this approach has shown to produce on average high quality recommendations, which makes it the most commonly used approach, some users are not satisfied: they get low quality recommendations. Being able to anticipate if a recommender will provide a given user with inaccurate recommendations, would be a major advantage. Nevertheless, little attention has been paid in the literature to studying this particular point. In this work, we assume that some of the users who are not satisfied do not respect the assumption made by the social approach of recommendation: their preferences are not coherent with those of others; we consider they have atypical preferences. We propose measures to identify these users, upstream of the recommendation process. These measures only exploit the users profile. The experiments conducted on a state of the art corpus and three social recommendation techniques show that the proposed measures allow to identify reliably a subset of users with atypical preferences, who will actually get inaccurate recommendations with a social approach. One of these measures is the most accurate, whatever is the recommendation technique.

Keywords: Atypical preferences · Atypical users · Recommender systems · Collaborative filtering · Accuracy of recommendations

1 Introduction

The continuous increase of the amount of data available on the Internet makes the task of accessing targeted information more and more complex for the users. This is the reason why many services now offer to assist their users during their search, by selecting for them the most relevant information or data. Several types of such services are proposed, among which recommender systems (RSs) [1]. Through a recommendation process, a RS aims to guide a user, called the *active user*: the user the system aims to provide with recommendations, towards resources relevant for him/her. A resource can be a book, a movie, a

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 191–210, 2016.
DOI: 10.1007/978-3-319-30996-5_10

web page, etc. To make such a recommendation possible, the system uses the knowledge it has collected about this active user.

RSs have been studied for more than twenty years [1]. The two most common approaches are content-based filtering [2] and collaborative filtering (CF) [3,4]. Content-based filtering exploits the content of the resources (as well as indexes, keywords, title, type of the resource, etc.) to select those that match the active user's preferences. Conversely, CF (also referred to as social filtering) does not require the exploitation of the content of the resources. It relies on the assumption that users' preferences are consistent among users, which allows to infer the active user's preferences from those of other users. In both approaches, users' preferences are generally represented by ratings on the resources. As CF is the most popular approach, it will be the focus of this work.

Providing users with high quality recommendations is of the highest importance. Indeed, in the context of e-commerce it increases customer retention, in e-learning it improves learners' learning process, in digital libraries it allows users to save time, etc. The quality of the recommendations provided by CF is now considered as acceptable on average [5]. However, some users do not receive accurate recommendations, which results in serious consequences: unsatisfied users, customer attrition, failure among learners, time wasted, etc.

If we are unable to provide each user with accurate recommendations, we are convinced it is essential that a given recommender can anticipate, upstream of the recommandation process, the users it will provide with inaccurate recommendations. Once these users are identified, the system can decide to not provide them with recommendations at all, or decide to use another approach specifically dedicated to these users. The literature has emphasized that one reason why some users are not satisfied is the small number of preferences the system collected about them. This problem is referred to as the cold-start problem [6]. However, some users with a significant number of preferences still get inaccurate recommendations. This can also be explained by the quality of the preferences collected about these users [7] or by the inconsistency of the users when expressing their preferences [8]. Recent works have noticed that some specific users tend to rate resources differently than other users [9,24]. We will refer their preferences to as atypical preferences. Remind that collaborative filtering assumes that preferences (ratings) are consistent between users. As these users do not match this requirement (their preferences are not consistent with those of others), this may explain why some of them get inaccurate recommendations.

The work conducted in this paper is in line with these latter works. We aim at identifying reliably users with atypical preferences (ratings) and who will receive inaccurate recommendations. From now on, we will refer these users to as atypical users. Their identification will be performed prior to any recommendation computation. To reach this goal, we introduce several measures that reflect the atypicity of preferences of a user.

Section 2 presents a short overview of recommender systems and the way atypical users are identified and managed in social recommendation. Section 3 introduces the three measures we propose to identify atypical users. Then, in

Sect. 4 the experiments we conducted to evaluate those measures are presented. Finally, we conclude and discuss our work in the last section.

2 Related Works

2.1 Social Recommender Systems

To provide a user, referred to as the *active user*, with some personalized recommendations, the social recommendation, also denoted by collaborative filtering (CF) [3,4], relies on the knowledge of other users preferences (generally some ratings) on resources. When the ratings are not available, preferences can be inferred from the traces of activity left by the users [10].

There are two main techniques in social recommendation: the memory-based technique and the model-based technique [11]. The memory-based technique (also referred to as instance-based learning) exploits directly users' preferences, without pre-processing. The most commonly used technique, the K Nearest Neighbors (KNN) user-based paradigm [3], exploits neighbor users of the active user. First, it computes the similarities of preferences between the active user and each other user. There are many ways to compute the similarities, the most popular is the Pearson correlation coefficient presented in Eq. (1). Second, it identifies the k nearest neighbors of u who have rated r(those with the highest similarity value). Last, it computes an estimation of the active user's rating using the ratings of his/her K nearest neighbors, using the weighted mean average (see Eq. (2)).

$$Pearson(u,v) = \frac{\sum_{r \in R_{uv}} (n_{u,r} - \overline{n}_u)(n_{v,r} - \overline{n}_v)}{\sqrt{\sum_{r \in R_{uv}} (n_{u,r} - \overline{n}_u)^2} \sqrt{\sum_{r \in R_{uv}} (n_{v,r} - \overline{n}_v)^2}} \tag{1}$$

where $n_{u,r}$ is the rating of the user u on the resource r, R_{uv} is the set of co-rated resources by users u and v and $\overline{n}_u$ is the average rating of u.

$$n^*_{u,r} = \overline{n}_u + \frac{\sum_{v \in V_{u,r}} (n_{v,r} - \overline{n}_v) * sim(u,v)}{\sum_{v \in V_{u,r}} |sim(u,v)|} \tag{2}$$

where $n^*_{u,r}$ is the estimated rating of user u on resource r, $V_{u,r}$ represents the k nearest neighbors of u, who rated the resource r and *sim(u,v)* is the similarity calculated between u an his/her neighbor v. The similarity can be instantiated by the Pearson correlation coefficient (see Eq. (1)). r Another well-known memory-based paradigm is the item-based paradigm which computes the similarities between items (resources) to deduce preferences of users. This paradigm relies on the hypothesis that if a user like a resource r, he/she will like the most similar resources to r. Once more the Pearson correlation coefficient (presented in Eq. (1)) is the most popular similarity measure used in the item-based paradigm.

The memory-based technique is simple to implement, provides high quality recommendations and takes into account each new preference dynamically in the recommendation process. However, it does not scale, due to the computation cost of the high number of similarities.

The model-based technique learns, as its name suggests, a model that describes the data (preferences). This model is used to estimate unknown preferences, so to provide the active user with recommendations. This approach does not suffer so much from the scalability problem. However, it does not easily allow dynamic changes in the model, especially if it has to be updated each time a new preference is provided by a user.

The model-based matrix factorization technique [12] is now the most commonly used technique, due to the quality of recommendations it provides. The matrix of users' preferences is factorized into two sub-matrices, one representing users, the other representing the resources, both in a common sub-space where dimensions correspond to latent features. Then, to compute a estimation of the rating of the user u on the resource r, the recommender multiply the vector of latent features associated to u by the vector of latent features associated to r. There are several matrix factorization techniques, including the singular value decomposition (SVD) [13] and alternating least squares (ALS) [14].

One limit, common to all CF techniques (whether memory-based or model-based) is the cold-start problem [6], which is related to the lack of data on new resources or new users.

2.2 Identifying Atypical Users in Recommender Systems

In the literature, several terms are used to make reference to atypical users. They are deviant users [9], abnormal users [15], grey sheeps [16], etc. T Most of the techniques used to perform their identification are issued from data analysis. The abnormality measure [9,15] is the most commonly used one. It has actually several names such as abnormality or deviance. Those names reflect the tendency of a user to rate differently from others. This measure exploits the difference between the ratings assigned by a user on some resources and the average rating on these resources. It is defined by Eq. (3).

$$Abnormality(u) = \frac{\sum_{r \in R_u} |n_{u,r} - \overline{n_r}|}{||R_u||} \tag{3}$$

where $n_{u,r}$ represents the rating that user u assigned to resource r, $\overline{n_r}$ is the average rating of r among all users, R_u is the set of resources rated by u and $||R_u||$ is their number. The higher a user rates resources differently than the average user, the higher his/her abnormality value. Users with a high abnormality value are considered as atypical users. The main advantage of this measure is its low complexity. However, although it is the reference measure in the literature to identify users with atypical preferences, from our point of view it suffers from several limitations. First, the resources about which users' preferences are not unanimous (the ratings between users is very different) will unfairly increase the abnormality of the users who rate these resources. Second, this measure does not take into account the individual behavior of each user. For example, a user more strict than the average user may be labeled as abnormal, while he/she has similar preferences to others; he/she only differs in his/her way of rating resources. This measure will thus probably identify some users as atypical, whereas they will get accurate recommendations.

Some studies identify atypical users with the aim to explain the fluctuations of performance of RS [15,17–19]. To reach this goal, they study users' characteristics: number of ratings, number of neighbors, etc. For example, a link between the small number of ratings of a user and a high recommendation error may be identified (cold-start problem). In [15], the authors form clusters of users, based on their preferences and aim at interpreting the resulting clusters. Among the set of clusters, a cluster made up of atypical users is identified: users with a high recommendation error (RMSE) as well as a high abnormality (Eq. (3)) as well. However, we are convinced that in the general case, clustering fails to build a cluster of users with atypical preferences and who will get inaccurate recommendations. Indeed, an atypical user, in the sense of the social recommendation, has preferences that are not close to those of other users. Thus, if a user belongs to a cluster, it means that his/her preferences are similar to those of users in the same cluster. So, he/she is not an atypical user. The work presented in [16] also relies on clustering of users, and is in line with our conviction: it proposes to consider users who are far from the center of their cluster as atypical users.

[17] defines a clarity indicator, that represents how much a user is non-ambiguous in his/her ratings. This indicator is based on the entropy measure: a user is considered as ambiguous (small value of clarity) if his/her ratings are not stable across resources. Authors show that there is a link between the ambiguity of the ratings of a user and the quality of recommendations he/she gets. Users with a small clarity value are considered as noise and are discarded from the system; they do not receive any recommendations. We believe that this approach quickly appears constrained. Indeed, various ratings (preferences) of a user can be explained by several factors such as the evolution of his/her preferences through time, his/her varying preferences across domains, etc. Therefore, a social approach may anyway provide this user with high quality recommendations. Notice that, at the opposite of previous approaches, the clarity indicator does not reflect the coherence of a user's preferences with respect to other users, it reflects the coherence he/she has with him/herself. It can thus be exploited in an approach other than the social one. Clarity can also be linked the magic barrier concept [20] and to recent works about user inconsistency and natural variability [21], which aim at estimating an upper bound on the rating prediction accuracy.

The impact of users identified as atypical on the overall quality of recommendations has been studied. The comparison of the results presented is difficult as atypical users are not selected on the basis of the same criteria. However, they do all conclude that removing atypical users in the learning phase of the recommender improves the overall quality of the recommendations.

Notice that the identification of atypical users may be associated with the identification of outliers or anomalies. According to [22], an outlier is "an observation that deviates so much from other observations as to arouse suspicion that is was generated by a different mechanism". In the context of recommender systems, an outlier is a user whose preferences appear to have been generated by a different preference expression mechanism. Criterion based, statistical approaches and clustering are also widely used in the field of outliers detection [23].

2.3 Managing Atypical Users in Recommender Systems

Once atypical users have been identified, one question that can be addressed is related to their management. In the context of recommender systems, new recommendation approaches dedicated to these specific profiles have been proposed, with the aim to provide them with better recommendations.

In [9], which refers atypical users to as deviant users, the authors divide the set of users into two subsets: deviant and non-deviant users, using the abnormality measure (Eq. (3)). These two subsets are considered independently when training recommendation models (two models are formed), as well as during the recommendation process. Only deviant users are taken into account when the active user is identified as deviant. Conversely, only non-deviant users are considered when the active user is non-deviant. This approach has shown to improve the quality of recommendations provided to non-deviant users. However, it has no impact on the quality of the recommendations provided to deviant users. This confirms our intuition that atypical users do not share preferences with any other user. In addition, we find this result not surprising as the recommendation approach has not been adapted to these specific users.

We previously reported how [16] identify atypical users through clustering. To address these atypical users, they use a specific cluster-based CF algorithm (model-based approach) to better reflect the preferences of these users and to offer them more accurate recommendations. Authors assume that these users only have partial agreement with the rest of the community (i.e. CF will fail on these users) and propose to rely on the content of resources to generate recommendations.

Finally, J. Bobadilla [24] has proposed a more general solution to take into account the specificities of atypical users, through a new similarity measure. This new measure is based on the singularity of ratings. A rating on a resource is considered as singular if it does not correspond to the majority rating on this resource. Authors assume that atypical users tend to assign singular ratings to resources. The singularity is used when computing the similarity between users: the more a rating is singular, the greater is its importance. The similarity between users is then used as in a classical KNN user-based recommendation approach. It has shown to provide high quality recommendations to users with specific preferences.

3 New Atypical Users Identification Measures

In this section, we introduce new measures for identifying atypical users, *i.e.* users with preferences that differ from those of the population of users.We consider that an atypical user receives inaccurate recommendations. These identification measures are designed to be used prior to the recommendation process, so they only rely on the users' profiles (preferences on resources). We want to propose measures that wont select any user receiving accurate recommendations, to not have a negative effect on him/her.

3.1 CorrKMax

The first measure we propose is dedicated to the user-based *KNN* technique. We are convinced that the user-based approach, which exploits the K most similar users to the active user, fails in the case the active user does not have enough highly similar users. We thus define $CorrKMax$ to highlight the link between the similarity of the most similar users of a user u and the quality of the recommendations he/she gets. $CorrKMax(u)$ (Eq. (4)) represents the average similarity between the active user u and his/her K most similar users.

$$CorrKMax(u) = \frac{\sum_{v \in Neigh(u)} Pearson(u,v)}{||Neigh(u)||} \tag{4}$$

where $Pearson(u,v)$ is the Pearson correlation between the preferences of users u and v (see Eq. (1)). $Neigh(u)$ represents the k most similar users to u, in the limit their correlation with u is positive. We believe that the users associated with a low value of $CorrKMax(u)$ receive inaccurate recommendations.

The two following measures are an extension of the *Abnormality* measure from the state of the art, which has shown good atypical users identification capabilities (see Sect. 2.2). To overcome the limitations that we have mentioned and presented in the previous section, we propose a first improvement.

3.2 AbnormalityCR

The *AbnormalityCR* (Abnormality with Controversy on Resources) measure assumes that the meaning of the discrepancy between a rating on a resource and the average rating on this resource differs according to the resource. Indeed, a large discrepancy on a controversial resource has not the same meaning as a large discrepancy on a consensual resource. The abnormality measure of the state of the art considers these differences as equal, which has the effect of increasing the abnormality of users who express their preferences on controversial resources. We therefore propose to reduce the impact of the ratings on controversial resources, by weighting them with the degree of controversy of the resources they refer to.

This degree of controversy of a resource is based on the standard deviation of the ratings on this resource. The *AbnormalityCR* of a user u is computed as shown in Eq. (5).

$$Abnormality_{CR}(u) = \frac{\sum_{r \in R_u} ((n_{u,r} - \overline{n_r}) * contr(r))^2}{\|R_u\|} \tag{5}$$

where $contr(r)$ represents the controversy associated with resource r. It is based on the normalized standard deviation of ratings on r and is computed according to Eq. (6).

$$contr(r) = 1 - \frac{\sigma_r - \sigma_{min}}{\sigma_{max} - \sigma_{min}} \tag{6}$$

where σ_r is the standard deviation of the ratings associated with the resource r. σ_{min} and σ_{max} are respectively the smallest and the largest possible standard deviation values, among resources. The computation complexity of *AbnormalityCR* is comparable to that of the abnormality of the state of the art. It can therefore be computed frequently and thus take into account new preferences.

3.3 AbnormalityCRU

The *AbnormalityCRU* (Abnormality with Controversy on Resources and Users) measure is a second improvement of the *Abnormality* measure. It starts from the observation that neither *Abnormality*(u) nor *AbnormalityCR*(u) reflect the general behavior of the user u. Thus, with these measures, a user who is strict in his/her way to rate resources may be considered as atypical, even if his/her preferences are actually not. In addition, this user will probably receive high quality recommendations. To avoid this bias, we propose to center the ratings of each user around his/her average rating. This way to reflect the user's behavior is also the one used in the Pearson correlation coefficient. Furthermore, the average rating on a resource is computed on the centered ratings, as well as the controversy. The abnormality of a user u, denoted by *AbnormalityCRU*(u), is computed using Eq. (7).

$$Abnormality_{CRU}(u) = \frac{\sum_{r \in R_u} [(|n_{u,r} - \overline{n_u} - \overline{n_{C_r}}|) * contr_C(r)]^2}{\|R_u\|} \tag{7}$$

where $\overline{n_{C_r}}$ represents the average centered rating on the resource r, $contr_C(r)$ represents the controversy associated with resource r, computed from the standard deviation of the ratings on u, centered with respect to users. The computation of *AbnormalityCRU*(u) is more complex than *AbnormalityCR*(u), but should allow a more accurate identification of atypical users.

Note that these last two measures are independent of the recommendation technique used, whether it is KNN or matrix factorization, contrary to the $CorrKMax$ measure, dedicated to the user-based KNN technique.

4 Experiments

The experiments we conduct in this section are intended to assess the quality of the atypical users identification measures we propose ($CorrKMax$, *AbnormalityCR* and *AbnormalityCRU*) in comparison with the measure from the state of the art (*Abnormality*). The assessment is based on the quality of the recommendations, more precisely the errors, provided to users identified as atypical.

4.1 Errors Measures

The quality of recommendations is evaluated through two standard measures: the Root Mean Square Error (RMSE) and the Precision. The former exploits the discrepancy between the rating provided by a user on a resource and the rating estimated by the recommender, the latter corresponds to the proportion of accurate predictions. The lower the RMSE, the higher the accuracy of recommendations provided to users. On the contrary, the higher the Precision, the higher the accuracy of the recommender system. In this work, we will specifically exploit per-user RMSE ($RMSE(u)$) and Precision ($Precision(u)$), computed respectively by Eqs. (8) and (9).

$$RMSE(u) = \sqrt{\frac{\sum_{r \in R_u} (n_{u,r} - n^*_{u,r})^2}{||R_u||}} \tag{8}$$

$$Precision(u) = \frac{||(|n_{u,r} - n^*_{u,r}| < 0.5)||}{||R_u||} \tag{9}$$

where $n^*_{u,r}$ is the estimated rating of user u on resource r.

4.2 Dataset and System Settings

Experiments are conducted on the MovieLens100K[1] dataset from the state of the art. MovieLens100K is made up of 100,000 ratings from 943 users on 1,682 movies (resources). The ratings range from 1 to 5, on integer values. We divide the dataset into two sub-sets made up of 80 % (for learning) and 20 % (for test) of the ratings of each user.

As presented in the beginning of this paper, our goal is to identify the users who will be provided with inaccurate recommendations, due to their atypical preferences. The literature emphasizes that users about who the system has collected few preferences get inaccurate recommendations (cold-start problem). To not bias our evaluation, we decide to discard these users from the dataset. We consider that a user who has less than 20 ratings in the training set is associated to cold-start [25]. The set of users is then reduced to 821 users (97 k ratings).

To compute the per-user errors, we implement three different commonly used CF techniques: a user-based technique, an item-based technique and a matrix factorization technique (see Sect. 2.1). Evaluating the atypical users identification measures on various techniques will allow us to determine which measure fits which technique or if these measures are generic: they are accurate whatever is the technique used. We set up the mostly used settings in the state of the art for each of these three techniques.

The user-based technique defines the similarity of two users as the Pearson correlation coefficient (see Eq. (1)) between their two rating vectors. The rating

[1] http://grouplens.org/datasets/movielens/.

estimation for a user is based on the ratings of his K nearest neighbors, using a weighted average of their ratings (see Eq. (2)). We fix K = 20 for this dataset.

The item-based technique defines the similarity between two items as the Pearson correlation coefficient (see Eq. (1)) between their rating vectors. The rating estimation for a user u on an resource r is based on the most similar items to r rated by u, we used a weighted average of the ratings. Such as in the user-based recommender, we fix the number of most similar items to K = 20.

We use the ALS factorization technique to compute the matrix factorization with 5 latent features. The ALS factorization is the most accurate technique to manage sparse matrices.

In order to give us a first overview of the link between those two elements, in the following section, we focus on the correlation between errors calculated with those techniques and our identification measures.

4.3 Correlations Between Identification Measures and Recommendation Error

Four measures are studied in this section: *Abnormality* from the state of the art and the three measures we propose: $AbnormalityCR$, $AbnormalityCRU$ and $CorrKMax$ (with $K = 20$). Based on these correlations, we can determine which measures are good indicators of the quality of recommendations that will be proposed to users.

The correlations on the user-based recommender are presented in Table 1.

Table 1. Correlations between identification measures and RMSE/Precision of a user-based technique.

	RMSE	Precision
Abnormality	0.453	−0.274
$AbnormalityCR$	0.504	−0.305
$AbnormalityCRU$	0.546	−0.364
$CorrKMax$	−0.22	0.07

Let us first focus on the *Abnormality* measure from the state of the art. Its correlation with RMSE is 0.453. This correlation is significant and confirms the existence of a link between the *Abnormality* of a user and the accuracy of the recommendation he/she gets: the higher the *Abnormality* of a user, the higher the error made on the rating estimation, so the lower the accuracy of the recommendations he/she receives. At the opposite, the lower the *Abnormality*, the higher the accuracy. Recall that a user with a high *Abnormality* value is considered as atypical. The correlation between *Abnormality* and Precision is less significant (−0.274) but does not negate the previous conclusions.

When considering $AbnormalityCR$, the correlation with RMSE reaches 0.504, which corresponds to an improvement of 11 % of the correlation compared to *Abnormality*. In parallel, the correlation of the *AbnormalityCR* with the Precision is also improved by 11 % compared to *Abnormality*. We can deduce that integrating the controversy associated with the resources in the computation of the Abnormality improves the estimation of the accuracy of the recommendations provided to users.

With *AbnormalityCRU*, the correlation with RMSE is equal to 0.546, which corresponds to a further improvement of 8 % (20 % with respect to *Abnormality*) and the correlation with Precision is equal to 0.364, which correspond to a further improvement of 19 % (32 % with respect to *Abnormality*). So, taking into account users' rating peculiarities (users' average rating) further improves the estimation of the accuracy of recommendations.

The correlation between *CorrKMax* and RMSE (−0.22) or Precision (0.07) indicates that, contrary to our intuition, the quality of a user's neighborhood is not correlated with the quality of the recommendations provided to him/her, with a *KNN* recommendation technique. This result is surprising as the *KNN* technique assumes that the more a user is correlated with the active user, the more he/she is reliable, and thus the more important he/she is in the computation of recommendations for this active user.

Table 2 presents the correlations on the item-based recommender.

Table 2. Correlations between identification measures and RMSE/Precision of a item-based technique.

	RMSE	Precision
Abnormality	0.398	−0.225
AbnormalityCR	0.421	−0.252
AbnormalityCRU	0.480	−0.363
CorrKMax	−0.09	0.03

With the item-based technique, the correlations are all weaker than with the user-based technique. Nevertheless, most of those correlations are still significant such as the correlation between *Abnormality* and RMSE with a value of 0.398. The *AbnormalityCR* measure increases this correlation to 0.421 (+6 %) and the *AbnormalityCRU* measure increases it to 0.480 (+20 %). Similar improvements can be measured on the Precision. The correlation between *Abnormality* and Precision is equal to −0.225. The correlation increases by 12 % with *AbnormalityCR* (−0.252) and increases by 61 % with *AbnormalityCRU* (−0.363).

We can then conclude about memory-based approaches that *AbnormalityCR* and *AbnormalityCRU* add some important and useful information to the state of the art *Abnormality*. The *AbnormalityCRU* measure is once more the more correlated with the errors. The *CorrKMax* is absolutely not tied to the errors of the item-based technique, even less than with the user-based technique.

The correlations on the matrix factorization technique are presented in Table 3.

Table 3. Correlations between identification measures and RMSE/Precision of a matrix factorization technique.

	RMSE	Precision
$Abnormality$	0.432	−0.297
$AbnormalityCR$	0.409	−0.285
$AbnormalityCRU$	0.488	−0.398
$CorrKMax$	−0.20	0.15

The correlation between the RMSE and the *Abnormality* measure is equal to 0.432 and the correlation between RMSE and *AbnormalityCR* is only equal to 0.409. This means that *Abnormality* is more related to the RMSE of a matrix factorization recommender than *AbnormalityCR*. This could indicate that the matrix factorization process can reduce the impact of the controversy of resources on errors. Furthermore, as on the memory-based techniques, the *AbnormalityCR* measure is less correlated with the Precision on matrix factorization errors. The *AbnormalityCRU* measure is once again the more correlated measure with both errors. It improves the correlation between the RMSE and *Abnormality* from 0.432 to 0.488 (+13 %). The $CorrKMax$ measure shows its best correlations with the matrix factorization technique. However, the correlations are still not significant enough (−0.20).

In conclusion, with the three techniques, *AbnormalityCRU* is the more related measure to the system errors. At the opposite, the $CorrKMax$ is correlated to none of those three techniques errors. Another remark is that the four studied measures are more correlated with the user-based errors.

4.4 Recommendation Error for Atypical Users

The correlations studied in the previous experiments aimed at evaluating the relationship between the abnormality measures and the recommendation errors on the complete set of users. However, there may be a relationship within only a subset of users. In that case, the correlation may not allow to identify this relationship. In particular, in this paper we aim at identifying a link between users identified as atypical and error measures. Therefore, in the following experiments, we will no more focus on the correlation between identification measures and errors measures, but only on the errors observed on users identified as atypical. The users with an extreme value of the identification measure are considered as atypical (the highest ones for the abnormality measures).

To study these errors, we depict them with the minimum, the maximum, the quartiles and the median values, and draw box plots. The four identification measures: *Abnormality*, *AbnormalityCR*, *AbnormalityCRU* and $CorrKMax$ are studied.

To evaluate precisely these four measures, we compare their box plots with the one of the complete set of users (denoted by Complete in Figs. 1, 2, 3 and 4).

Recall that, the higher the RMSE, the more accurate the measure and the lower the Precision, the more accurate the measure. As the identification measures do not all have comparable values, we did not use a predefined atypicity threshold value. We chose to consider a predetermined percentage of atypical users, which we fixed experimentally at 6 % of the complete set of users. This corresponds to about 50 users among the 821 users. We compare these measures in the framework of the three recommendation techniques: the user-based technique, the item-based technique and the matrix factorization technique.

Errors Associated with Atypical Users in the User-Based Technique. The distribution of the errors (RMSE and Precision) obtained with the user-based technique, according to the identification measure, are presented in Figs. 1 and 2 respectively.

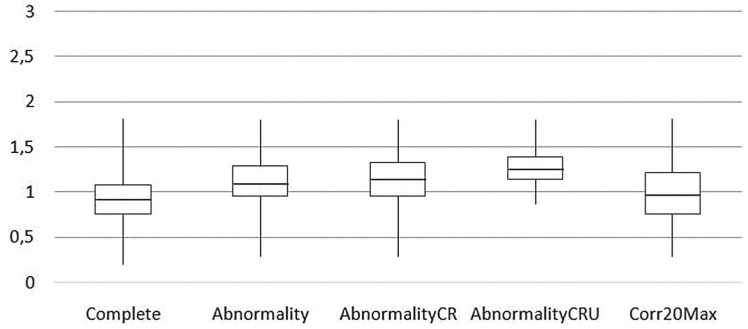


Fig. 1. Distribution of RMSE of atypical users with the user-based technique.

The median RMSE on the complete set of users (Complete) is 0.91. When exploiting the *Abnormality* measure, the median RMSE of the 6 % users with the highest *Abnormality* reaches 1.12. This represents an increase in the RMSE by more than 25 %. Furthermore, we can notice that the median value of *Abnormality* is equal to the third quartile of the Complete set. This mean that 50 % of users identified as atypical users with *Abnormality* are part of the 25 % of users with the highest RMSE in the Complete set: this measure is quite accurate. However, 25 % of the users considered as atypical have a RMSE lower than the median RMSE of the complete set of users. This means that, although *Abnormality* from the state of the art allows to identify users who will receive inaccurate recommendations, it appears to select a significant number of users who will receive accurate recommendations (false detection). *Abnormality* is thus not precise enough. Recall that users identified as atypical may either not receive any recommendations at all, or may get recommendations from another technique, which may be less accurate. It is very important to not identify users as atypical if they will receive high quality recommendations in order to not modify their recommendations. The accuracy of the measure used is thus of the highest importance. The limits of the *Abnormality* measure that we presented in the previous section (see Sect. 2.2) are confirmed: the use of the discrepancy between a rating and the average rating on a resource is not sufficient to reliably predict inaccurate recommendations.

The quality of both *AbnormalityCR* and *AbnormalityCRU* measures is higher than the one of *Abnormality*. *AbnormalityCR* slightly improves the performance of the *Abnormality* measure with a median equal to 1.17 (increase of 4 %). *AbnormalityCRU* appears to be the best one: all the users identified as atypical users have a RMSE higher than the median RMSE of the complete set of users. In addition, over 75 % of these users have a RMSE higher than 1.13, *i.e.* 75 % of the users with the highest *AbnormalityCRU* are among the 25 % of the complete set of users who will receive inaccurate recommendations. The accuracy of the *AbnormalityCRU* measure is thus high.

Once more *CorrKMax* (with $K = 20$) is not accurate, the users identified as atypical tend to receive high quality recommendations (50 % of them). The low similarity of a user's nearest neighbors is thus not a reliable information to predict the low quality of recommendations this user will receive.

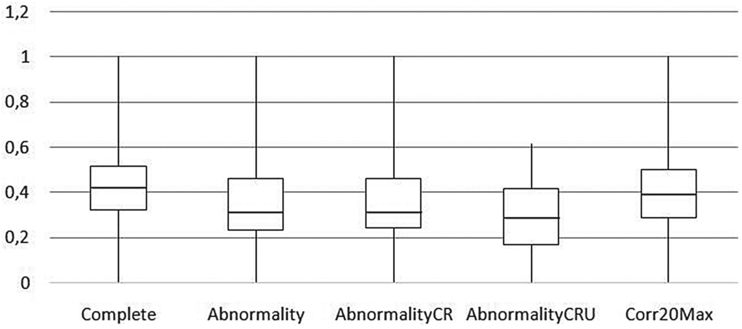


Fig. 2. Distribution of Precision of atypical users with the user-based technique.

The distributions of the Precision, presented in Fig. 2, confirm the results obtained with RMSE. The median Precision obtained on the complete set of users is equal to 0.42. The median Precision of *Abnormality* and *AbnormalityCR* is 0.31, which correspond to an improvement of 35 %. Moreover, 25 % of the complete set of users obtain a Precision lower than 0.32, which mean that 50 % of users selected with the *Abnormality* and *AbnormalityCR* measures belong to the set of 25 % worst Precisions of the system. In contrast to RMSE, we can observe that, with the Precision measure, *AbnormalityCR* does not improve the performance of *Abnormality*. Nevertheless, those measures select also users with accurate recommendations. The median Precision obtained with *AbnormalityCRU* is 0.28, which is not significantly lower than the median Precision of *Abnormality*, but we can see that *AbnormalityCRU* does not select users receiving accurate recommendations. *AbnormalityCRU* is thus the better measure to select users receiving inaccurate recommendations.

The results obtained with $CorrKMax$ (nearly a random selection) are, once more, not conclusive.

The results obtained with Precision are less clear-cut than those obtained with RMSE on those four measures. We can thus deduce that the controversy on resources is more effective at aiming high range deviations between estimations and users ratings than low range deviations. We can conclude that, when the *AbnormalityCRU* measure identifies a user as an atypical user, he/she will actually receive inaccurate recommendations with the user-based recommendation technique.

Errors Associated with Atypical Users in the Item-Based Technique. The distribution of the errors obtained with the item-based technique are presented in Figs. 3 and 4.

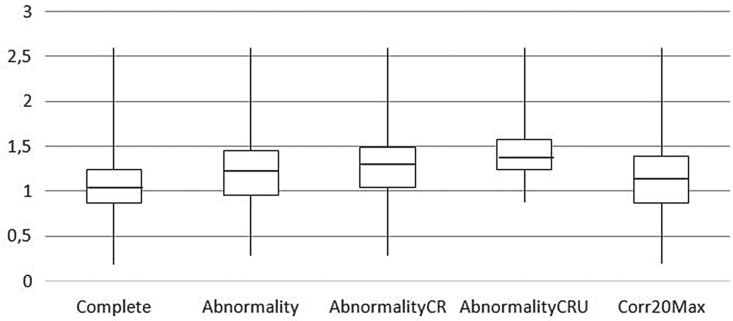


Fig. 3. Distribution of RMSE of atypical users with the item-based KNN technique.

With the item-based technique, the median RMSE of the complete set of users states at 1.04 and 25 % of users have a RMSE higher than 1.23. The median RMSE of the users select with *Abnormality* is equal to 1.27, which corresponds to an increase of the median RMSE by 22 % and means that 50 % of users selected with *Abnormality* belong to the 25 % of users of the Complete set with the worst RMSE. As with the user-based technique, those results are successively increased with *AbnormalityCR* and *AbnormalityCRU*. The best results are once more obtained with *AbnormalityCRU*: 75 % of users have a RMSE within the 25 % RMSE of the system. The conclusions about this technique are the same than those obtained with the user-based technique.

According to the Precision measure, *AbnormalityCRU* shows, once more, the best results: 75 % of the users selected belong to the set of the 25 % worst RMSE in the complete set of users. In Figs. 3 and 4, the distributions of the errors associated with $CorrKMax$ are once more not conclusive. We can then conclude that in a memory based approach (user-based or item-based), the 20 highest correlations between items or users are not enough to predict the quality of recommendations.

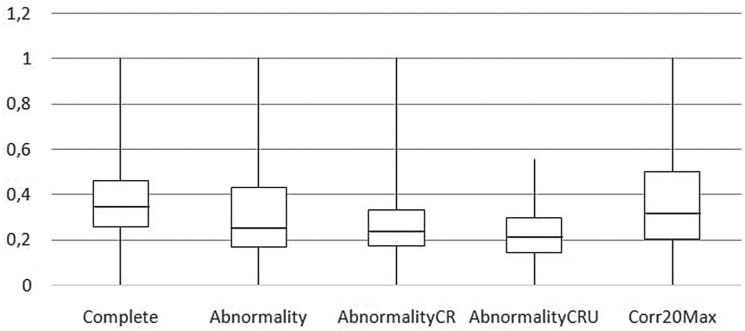


Fig. 4. Distribution of Precision of atypical users with the item-based technique.

Errors Associated with Atypical Users in the Matrix Factorization Technique.
In this section, we seek to study how the identification measures behave when using a matrix factorization-based technique. We will compare their accuracy to the item-based and user-based. The errors associated with $CorrKMax$ are not studied here, as $CorrKMax$ is dedicated to the memory-based approaches (whether item-based or user-based). Figures 5 and 6 presents the distributions of the RMSE and Precision of the three Abnormality measures with a matrix factorization technique, as well as the reference distribution on the complete set of users.

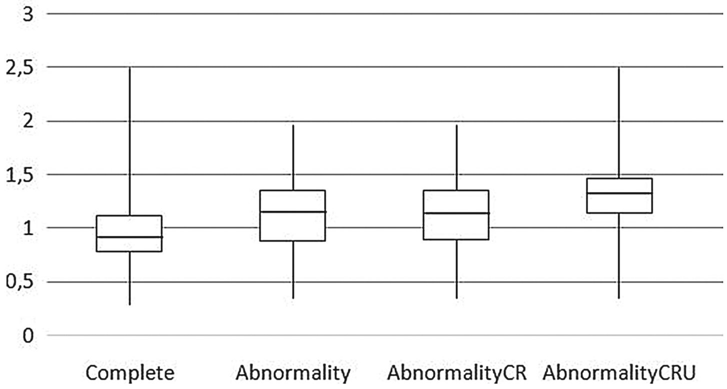


Fig. 5. Distribution of RMSE of atypical users with the matrix factorization technique.

The median RMSE (see Fig. 5)on the complete set of users is equal to 0.92 and the median RMSE obtained with *Abnormality* is 1.17, which corresponds to an increase of 27 %. For the first time, *AbnormalityCR* obtain approximately the same results than *Abnormality*, it has a median RMSE of 1.13. The controversy on resources seems to have no impact on the selection of atypical users with

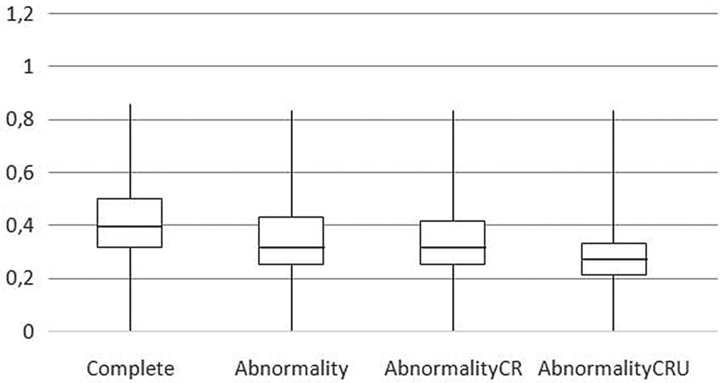


Fig. 6. Distribution of Precision of atypical users with the matrix factorization technique.

the matrix factorization technique. Nevertheless, *AbnormalityCRU* remains the most accurate measure by far for identifying atypical users. Moreover, we can observe that the accuracy of *AbnormalityCRU* is similar to the one observed with the memory-based approaches: 75 % of users identified as atypical belong to the set of 25 % of users who get the worse recommendations in the complete set of users.

In conclusion, we can say that the *AbnormalityCRU* measure, which we propose, is the most accurate measure: when it identifies a user as atypical, he/she most likely will receive low quality recommendations. Moreover, this measure is independent of the recommendation technique: it is efficient on both memory-based (item-based and user-based) and on the matrix factorization model-based approach. [8] has shown that different recommendation approaches (collaborative user-user, collaborative item-item, content, etc.) tend to fail on the same users. It would be interesting to compute the *AbnormalityCRU* on those users.

However, although *AbnormalityCRU* has a high accuracy, some users (from the complete set) with a high per user RMSE of the matrix factorization technique are identified by none of the Abnormality measures: it concerns 50 % of the users who have a RMSE greater than 1.5 (27 users). This means that further work has to be conducted to identify the characteristics of these users.

On the Fig. 6 we observe the same slight difference between *Abnormality* and *AbnormalityCR* than with the user-based technique. This observation should be studied in a future work. The same conclusions can be extracted from those repartitions of Precision, *AbnormalityCRU* is the better indicator of low quality recommendations.

4.5 Synthesis of Results

The correlations between $CorrKMax$ and errors are not conclusive, such as the errors of users selected with this measure. The $CorrKMax$ measure does not allow to identify users who will receive low quality recommendations. This indicator can not be used with any of these recommendation techniques.

At the opposite, the correlations between the abnormality measures (*Abnormality*, *AbnormalityCR* and *AbnormalityCRU*) and RMSE are significant, whatever is the recommendation technique. The *Abnormality* measure from the state of the art allows to select users with a median RMSE/Precision higher than the median RMSE/Precision of the complete set of users, regardless the recommendation technique. This measure shows its best RMSE values on the matrix factorization technique, with an increase of 27 % of the median RMSE, compared to the median RMSE of the complete set. In parallel, with the user-based technique, the *Abnormality* measure obtains the best results with a decrease of 35 % of the median Precision (compared to the complete set).

The *AbnormalityCR* measure shows slightly better results than the *Abnormality* measure except with the matrix factorization technique. Using an item-based technique, the *AbnormalityCR* measure shows its best results with an increase of 29 % of the median RMSE of the complete set of users and an increase of 40 % of the median Precision.

Finally, computing the *AbnormalityCRU* measure remains the best option to be able to identify atypical users, whatever is the recommendation technique. Indeed, *AbnormalityCRU* selects always at least 75 % of users which belong to the worst 25 % of RMSE of the system. Moreover, the *AbnormalityCR* measure does not improve the performance of the state of the art measure with all the recommenders, *e.g.* the matrix factorization technique. The controversy of items seems to improve the performance of the detection only with an item-based technique. Since the *AbnormalityCRU* measure is more complex to compute, the *AbnormalityCR* measure can be a good measure to select atypical users with the memory-based techniques (item-based and user-based), and the *Abnormality* measure would be used with a matrix factorization technique.

5 Conclusion and Perspectives

Social recommender systems is the context of this work. Our objective was to identify users who will receive inaccurate recommendations, upstream of the recommendation process, *i.e.* based only on the characteristics of their preferences. We hypothesized that users with preferences that differ from those of other users will receive inaccurate recommendations. We have referred these users to as atypical users. To validate this hypothesis, we proposed several measures for identifying atypical users, based on the similarity of users preferences with other users, on the average discrepancy of the ratings they provide in comparison with the average rating of other users, on the consensus of ratings on resources, or on users rating profile. We have shown, on a state of the art dataset, that the measure that uses all these criteria is the most accurate one and allows to reliably anticipate that a user will get inaccurate recommendations, with either a KNN-based techniques (user-based or item-based) or a matrix factorization technique.

In a future work, we will focus on the proposition of a new recommendation approach, to provide atypical users with high quality recommendations. In parallel, it will be interesting to investigate the reasons why some users do get inaccurate recommendations and are not identified by any of the measures studied, as mentioned in the previous section. Specifically, a user may be atypical on a subset of items, which is not considered by the measures studied here.

References

1. Goldberg, D., Nichols, D., Oki, B., Terry, D.: Using collaborative filtering to weave an information tapestry. Commun. ACM **35**(12), 61–70 (1992)
2. Bobadilla, J., Ortega, F., Hernando, A., Gutiérrez, A.: Recommender systems survey. Knowl. -Based Syst. **46**, 109–132 (2013)
3. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: Grouplens: an open architecture for collaborative filtering of netnews. In: Proceedings of the ACM Conference on Computer Supported Cooperative Work, CSCW 1994, (New York), pp. 175–186. ACM (1994)
4. Su, X., Khoshgoftaar, T.M.: A survey of collaborative filtering techniques. Adv. Artif. Intell. **2009**, 4:2 (2009)
5. Castagnos, S., Brun, A. Boyer, A.: When diversity is needed... but not expected!. In: IMMM, The Third International Conference on Advances in Information Mining and Management (2013)
6. Schein, A.I., Popescul, A., Ungar, L.H., Pennock, D.: Generative models for cold-start recommendations. In: Proceedings of the 2001 SIGIR workshop on recommender systems (2001)
7. Grcar, M., Mladenic, D., Grobelnik, M.: Data quality issues in collaborative filtering. In: Proceedings of ESWC- 2005 Workshop on End User Aspects of the Semantic Web (2005)
8. Ekstrand, M.: Towards Recommender Engineering. Tools and Experiments for Identifying Recommender Differences. PH.D. thesis, Faculty of the University of Minnesota (2014)
9. Del Prete, L., Capra, L.: Differs: a mobile recommender service. In: Proceedings of the Eleventh International Conference on Mobile Data Management, MDM 2010, (Washington, USA), pp. 21–26, IEEE Computer Society (2010)
10. Ormándi, R., Hegeds, I., Csernai, K., Jelasity, M.: Towards inferring ratings from user behavior in bittorrent communities. In: Proceedings of the 2010 19th IEEE International Workshops on Enabling Technologies: Infrastructures for Collaborative Enterprises (WETICE), pp. 217–222 (2010)
11. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art. IEEE Trans. Knowl. Data Eng. **17**(6), 734–749 (2005)
12. Sarwar, B.M., Karypis, G., Konstan, J.A., Riedl, J.T.: Application of dimensionality reduction in recommender system - a case study. In: ACM WebKDD Workshop (2000)
13. Billsus, D., Pazzani, M.J.: Learning collaborative information filters. In: Proceedings of the Fifteenth Intternational Conference on Machine Learning, ICML 1998, (San Francisco, CA, USA), pp. 46–54, Morgan Kaufmann Publishers Inc. (1998)

14. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: Proceedings of the Eighth IEEE International Conference on Data Mining, ICDM 2008, (Washington, DC, USA)pp. 263–272, IEEE Computer Society (2008)
15. Haydar, C., Roussanaly, A., Boyer, A.: Clustering users to explain recommender systems' performance fluctuation. In: Chen, L., Felfernig, A., Liu, J., Raś, Z.W. (eds.) ISMIS 2012. LNCS, vol. 7661, pp. 357–366. Springer, Heidelberg (2012)
16. Ghazanfar, M., Prugel-Bennett, A.: Fulfilling the needs of gray-sheep users in recommender systems, a clustering solution. In: 2011 International Conference on Information Systems and Computational Intelligence, 18–20 January 2011
17. Bellogín, A., Castells, P., Cantador, I.: Predicting the performance of recommender systems: an information theoretic approach. In: Amati, G., Crestani, F. (eds.) ICTIR 2011. LNCS, vol. 6931, pp. 27–39. Springer, Heidelberg (2011)
18. Griffith, J., O'Riordan, C., Sorensen, H.: Investigations into user rating information and predictive accuracy in a collaborative filtering domain. In: Proceedings of the 27th Annual ACM Symposium on Applied Computing, SAC 2012, (New York), pp. 937–942. ACM (2012)
19. Ekstrand, M., Riedl, J.: When recommenders fail: predicting recommender failure for algorithm selection and combination. In: Proceedings of the sixth ACM conference on recommender systems, pp. 233–236. ACM (2012)
20. Herlocker, J., Konstan, J., Terveen, L., Riedl, J.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. **22**(1), 5–53 (2004)
21. Bellogín, A., Said, A., de Vries, A.P.: The magic barrier of recommender systems – no magic, just ratings. In: Dimitrova, V., Kuflik, T., Chin, D., Ricci, F., Dolog, P., Houben, G.-J. (eds.) UMAP 2014. LNCS, vol. 8538, pp. 25–36. Springer, Heidelberg (2014)
22. Hawkins, D.M.: Identification of outliers, vol. 11. Springer, New York (1980)
23. Aggarwal, C.C.: An introduction to outlier analysis. In: Aggarwal, C.C. (ed.) Outlier Analysis, pp. 1–40. Springer, New York (2013)
24. Bobadilla, J., Ortega, F., Hernando, A.: A collaborative filtering similarity measure based on singularities. Inf. Process. Manage. **48**, 204–217 (2012)
25. Schickel-Zuber, Vincent, Faltings, Boi V.: Overcoming incomplete user models in recommendation systems via an ontology. In: Nasraoui, Olfa, Zaïane, Osmar R., Spiliopoulou, Myra, Mobasher, Bamshad, Masand, Brij, Yu, Philip S. (eds.) WebKDD 2005. LNCS (LNAI), vol. 4198, pp. 39–57. Springer, Heidelberg (2006)

Exploiting Temporal Dimension in Tensor-Based Link Prediction

Jaroslav Kuchař(✉), Milan Dojchinovski, and Tomas Vitvar

Web Intelligence Research Group, Faculty of Information Technology,
Czech Technical University in Prague, Prague, Czech Republic
{jaroslav.kuchar,milan.dojchinovski,tomas.vitvar}@fit.cvut.cz

Abstract. In the recent years, there is a significant interest in a link prediction - an important task for graph-based data structures. Although there exist many approaches based on the graph theory and factorizations, there is still lack of methods that can work with multiple types of links and temporal information. The creation time of a link is an important aspect: it reflects age and credibility of the information. In this paper, we introduce a method that predicts missing links in RDF datasets. We model multiple relations of RDF as a tensor that incorporates the creation time of links as a key component too. We evaluate the proposed approach on real world datasets: an RDF representation of the ProgrammableWeb directory and a subset of the DBpedia focused on movies. The results show that the proposed method outperforms other link prediction approaches.

Keywords: Link prediction · Temporal information · RDF · Tensor factorization

1 Introduction

Over the last few years the number of published RDF datasets in the Linked Data cloud has grown significantly. One of the key Linked Data publishing principles is to use URI references to identify Web resources and links between them[1]. Such link are usually defined at the time when the dataset was created and they are often not updated. However, over the time links can get old and loose their significance. Link prediction algorithms, on the other hand, find new links in datasets that are not explicitly present but they implicitly exist due to existing structural patterns.

An increasing amount of datasets and their evolution over time introduce another dimension to link prediction methods. In this paper we develop a novel method that is able to predict links in a single dataset that uses (i) *creation time of links*, and (ii) *existing structural patterns* in the dataset. We call this method a *time-aware link prediction*.

[1] http://www.w3.org/DesignIssues/LinkedData.html.

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 211–231, 2016.
DOI: 10.1007/978-3-319-30996-5_11

We primarily use a dataset from ProgrammableWeb[2], a leading Web APIs and mashup directory, that allows developers to publish information about their Web APIs and mashups and work in a social network of developers. At the time of creating a Web API or a mashup in the directory, a developer provides various technical and functional descriptions such as categories, tags and defines links between APIs and mashups. A link prediction method applied on the dataset from ProgrammableWeb may be used to find links to other categories, tags or Web APIs based on structural patterns in which Web APIs, mashups or developers occur. However, such method would ignore the fact that a Web API or a mashup can be outdated. Our link prediction method provides more precise results as it can effectively combine time information with structural patterns. We use (i) *tensors* as an underlying mechanism to model RDF data, (ii) *time information and an ageing function* to model the age of the data and (iii) a *tensor factorization technique* to evaluate an existence of new links. We adopted a widely used ageing to simulate the loss of the links' significance; decrease an impact of older links and promote more recent ones. Our assumption is that older links are less important due to their age, however, they can still have an influence on a link prediction due to structural patterns. We evaluate the method on a real-world datasets (Web services domain and the well-known knowledge base DBpedia) and we present its performance and capabilities.

The paper is structured as follows. Section 2 describes the time-aware link prediction method, its notations, definitions and the supporting algorithm. Section 3 describes several experiments we conducted to evaluate its performance and capabilities. In Sect. 4 we discuss various aspects of the method. In Sect. 5 we give an overview of the related work, and finally, Sect. 6 concludes the paper and presents the future work.

2 Time-Aware Link Prediction Method

2.1 Definitions

Tensor. *A multi-dimensional array of numerical values* [1]. The *order* of the tensor is the number of dimensions that the tensor uses. In our method we use a tensor of order three denoted by $\mathcal{Y}^{I \times J \times K}$, where $I, J, K \in \mathbb{N}$ and $I = J$. The (i, j, k) element of a third-order tensor is denoted as y_{ijk}.

Information Ageing. *A process of retention of information in a memory over time.* We represent the relation between time and retention using a *forgetting curve* [2]; defined as $R = e^{-\lambda T}$ where R is the memory retention, T is the amount of time since the information was received and $1/\lambda$ is the strength of the memory.

Based on the definition of the forgetting curve, we propose an ageing function

$$\mathcal{A}(t_0) = \mathcal{A}(t_x) * e^{-\lambda t}; t_0 > t_x, t = t_0 - t_x \tag{1}$$

[2] http://www.programmableweb.com/.

where $\mathcal{A}(t_0)$ is the amount of information at the time t_0, $\mathcal{A}(t_x)$ is the amount of information at the time t_x when the information was created, λ is ageing/retention factor and t is the age of the information. The information ageing is influenced by the λ parameter as the strength of the memory. The higher the value of the λ parameter is, the faster the loss of information is. Similarly, the older the information is, the lower is the amount of held information.

Note that Linked Data community has adopted several approaches to represent temporal information [3,4]. In this paper we use a single *starting time point* t_x which defines an existence of the link, i.e. the link exists since t_x (see Sect. 4 for discussion). We refer to this time as the creation time. We have no information about the duration of the existence of the link and we cannot conclude whether it is still valid (Open World Assumption).

2.2 Tensor-Based Model with Temporal Information

Simple graph structures can be modelled as matrices, which is preferred for graph structures with one type of links. However, since RDF data contain more than one type of links, we use a third-order tensor notation, which was proposed in [5]. We can project the third-order tensor as a set of incidence matrices, where each matrix contains only links between entities for a corresponding type of the link.

Let $\mathcal{Y} \in \{0,1\}^{N \times N \times M}$ be a tensor representing an RDF dataset. The tensor consists of two identical dimensions N representing a domain of entities (concepts and instances) in the dataset, and the third dimension M representing a domain of link types (properties) that explicitly exist in the dataset. The tensor element $y_{ijk} = 1$, if the i-th entity has link of a type k with the j-th entity, for $i, j \in \langle 0, N)$ and $k \in \langle 0, M)$. Otherwise, the tensor element $y_{ijk} = 0$. Each tensor element in the model has a value of 1 or 0 if a link between two entities exists or does not exist, respectively.

In this paper, we propose an extension of this model to include also temporal information. We focus on the situation, when the creation time of the links is available (see Sect. 4 for discussion). We use this information to modify the initial tensor $\mathcal{Y}$ such that values of tensor elements are reduced with respect to the creation time of the corresponding link. Let $\mathcal{X} \in \mathbb{R}^{N \times N \times M}$ be a tensor at the time t_0. We then compute a value of a tensor element x_{ijk} using the ageing function (1) as follows

$$x_{ijk} = y_{ijk} * e^{-\lambda t} \tag{2}$$

where $y_{ijk} \in \{0,1\}$ is the initial value of the tensor element, λ is the ageing factor and t is the link's age computed as a distance of the link's creation time and the time t_0 (see Sect. 2.1 for additional details about the ageing function).

Example 1. Consider an RDF dataset consisting of four instances of concepts *ls:Mashup* $(m1, m2, m3, m4)$ and *wl:Service* $(s1, s2, s3, s4)$, and three links *ls:usedAPI* that indicate usages of Web APIs in the mashups, i.e. $(m_2 \xrightarrow{t_0 - t_3} s_1, m_2 \xrightarrow{t_0 - t_1} s_3, m_4 \xrightarrow{t_0 - t_{15}} s_2)$. In this formula, each arrow indicates the age of

Table 1. Example of modelling data.

(a) Tensor $\mathcal{Y}$ model without ageing

	s_1	s_2	s_3	s_4
m_1	0	0	0	0
m_2	$\mathbf{1}_{(t_0-t_3)}$	0	$\mathbf{1}_{(t_0-t_1)}$	0
m_3	0	0	0	0
m_4	0	$\mathbf{1}_{(t_0-t_{15})}$	0	0

(b) Tensor $\mathcal{X}$ model with ageing

	s_1	s_2	s_3	s_4
m_1	0	0	0	0
m_2	$\mathbf{0.97}_{(t_0-t_3)}$	0	$\mathbf{0.99}_{(t_0-t_1)}$	0
m_3	0	0	0	0
m_4	0	$\mathbf{0.86}_{(t_0-t_{15})}$	0	0

the link in weeks since t_0. For example, $m_4 \xrightarrow{t_0-t_{15}} s_2$ indicates that the link was created 15 weeks ago.

Table 1 shows this information modelled as a tensor both with and without ageing (in this example we set the parameter $\lambda = 0.01$). Note that the link between the mashup m_4 and the service s_2 has a lower value due to the fact that this link was created earlier than the other two.

2.3 Learning Hidden Latent Factors

We use a tensor factorization technique to perform a structural analysis of an RDF dataset. We propose an extension of the RESCAL approach [5] which uses the time information. Each incidence matrix $\mathbf{X}_k$ of a tensor is factorized as

$$\mathbf{X}_k \approx \mathbf{A}\mathbf{R}_k\mathbf{A}^T, k = 0...M \tag{3}$$

where $\mathbf{A}$ is a matrix $N \times R$ which models a participation of an entity in a latent factor R, and $\mathbf{R}_k$ is a matrix $R \times R$ that models interactions of latent factors for the k-th relation (Fig. 1). The R is a configurable parameter of the factorization algorithm. It indicates the number of latent factors to be learned.

The matrix $\mathbf{A}$ and the matrices $\mathbf{R}_k$ are computed by solving the minimum optimization problem

$$\min_{\widehat{\mathbf{X}_k}} \parallel \mathbf{X}_k - \widehat{\mathbf{X}_k} \parallel_F \text{ , where } \widehat{\mathbf{X}_k} = \mathbf{A}\mathbf{R}_k\mathbf{A}^T \tag{4}$$

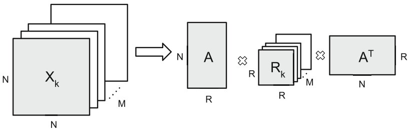


Fig. 1. Visualization of RESCAL [5].

	s_1	s_2	s_3	s_4
m_1	0	0	0	0
m_2	$0.95_{(t_0-t_3)}$	*0.04*	$0.98_{(t_0-t_1)}$	0
m_3	0	0	0	0
m_4	*0.11*	$0.83_{(t_0-t_{15})}$	*0.18*	0

Fig. 2. Example of reconstructed tensor $\mathcal{X}$ ($R = 3$).

Although there exist other tensor factorization algorithms, RESCAL [5] is the most suitable method for an analysis of multi–relational data and link prediction tasks, it scales well for larger datasets and it shows good performance [6].

In our extension of the algorithm, we use a tensor with elements as real positive numbers; lower values for older links and higher values for newer links. By using this tensor, latent factors can learn regularities in the model while reconstructed values are approximately the same as the original values. The extra non-zero values in the reconstructed matrices reflect the temporal information and the higher values are influenced by the higher values in the original model. The higher values represent the predicted links influenced by the recent links in the original model.

2.4 Time-Aware Link Prediction

The *link prediction* task evaluates a possible existence of a link between a pair of entities by using structural patterns in the dataset. Our *time-aware link prediction* task, on the other hand, evaluates a possible existence of a link between two entities while taking into account the age of explicit links in the dataset as well as structural patterns in the dataset.

To evaluate an existence of a link between i-th and j-th entity we do a reconstruction $\widehat{\mathbf{X}_k} = \mathbf{A}\mathbf{R}_k\mathbf{A}^T$ of a matrix $\mathbf{X}_k$ for a link of type k. The algorithm solves a minimum optimization problem with goal to predict links of type $\mathbf{k}$ from domain M from the $\mathbf{i}$-th entity from domain N. Note that in the following algorithm the terms *source entity*, *link* and *target entity* refer to the RDF terminology *subject*, *predicate* and *object*, respectively:

Inputs:

- An RDF dataset where each link contains information when the link was created.
- Ageing constant λ.
- A link of type $\mathbf{k}$ and an entity $\mathbf{i}$ as a source of links.
- A maximum number of target entities L.

Outputs:

- A set of Top-L entities as targets of links.

Algorithm:

1. Model a tensor $\mathcal{X}$ for the input RDF dataset and the ageing constant λ (see Sect. 2.2).
2. Compute factorization for the tensor $\mathcal{X}$ with the extended RESCAL algorithm (see Sect. 2.3).
3. Reconstruct a matrix $\widehat{\mathbf{X}_k}$ using the latent factor $\mathbf{R}_k$ and a matrix $\mathbf{A}$, where k indicates a link type in the query (see Formula (3)).
4. Read values x_{ijk} for the $\mathbf{i}$-th row and each $\mathbf{j}$-th column of $\widehat{\mathbf{X}_k}$. The values indicate whether a link between the $\mathbf{i}$-th entity and entity in the $\mathbf{j}$-th column should exist.

5. Sort values in decreasing order and return Top-L values. These values indicate target entities that should be linked with the source entity using the link type k. Note that the Top-L entities can also be evaluated by comparing $x_{ijk} > \theta$, where θ is some threshold.

Example 2. Consider data from Example 1 as an input RDF dataset. For simplicity, it contains only one type of the link ($k = usedAPI$). A tensor $\mathcal{X}$ in Table 1 corresponds to the first step of the algorithm for $\lambda = 0.01$. The second step factorizes the tensor to matrices $\mathbf{A}, \mathbf{R}_k$ and the third step provides approximation of the tensor. Figure 2 shows an example of the reconstructed matrix $\widehat{\mathbf{X}_k}$ ($R = 3$). For entity $i = m_4$ the corresponding row contains three possible candidates as new links (s_3, s_1, s_4) sorted decreasingly by the reconstructed value. Given the list of candidates, we can select either a set of Top-L elements or elements with the value above predefined threshold θ. Please note that the higher value for s_3 was influenced by the existing link with higher value, that was created more recently than the second one.

3 Evaluation and Experiments

In this section we demonstrate the time-aware link prediction method on the real-world dataset from ProgrammableWeb. Furthermore, we also performed evaluation experiments on other datasets.

The following questions we address in our experiments:

- *How temporal aspects influence the link prediction?*
- *How the evolution of dataset structure influences the link prediction?*

On several experiments, we evaluate the quality of the proposed method when compared with a set of baseline algorithms. The first experiment shows the difference of the proposed time-aware link prediction and a link prediction without temporal information. The following two experiments clarify the connection between predicted links, the time information and the structure of the dataset.

3.1 Linked Web APIs Dataset

For evaluation purposes, we created an extended version of the *Linked Web APIs* dataset. The dataset is an RDF representation of the ProgrammableWeb[3] directory, the largest mashup and Web APIs directory. It contains information about developers, mashups they created and Web APIs they used, together with categories they belong to. In addition, the dataset has information about tags assigned to each mashup and a Web API, formats and protocols that Web APIs support. We also collected information about the time when users, mashups or Web APIs appeared in the directory for the first time. The dataset contains information from June 2005 till the end of March 2013, it has in total 22 286 entities, 8 types of links and contains approx. 123 000 links.

[3] http://www.programmableweb.com/.

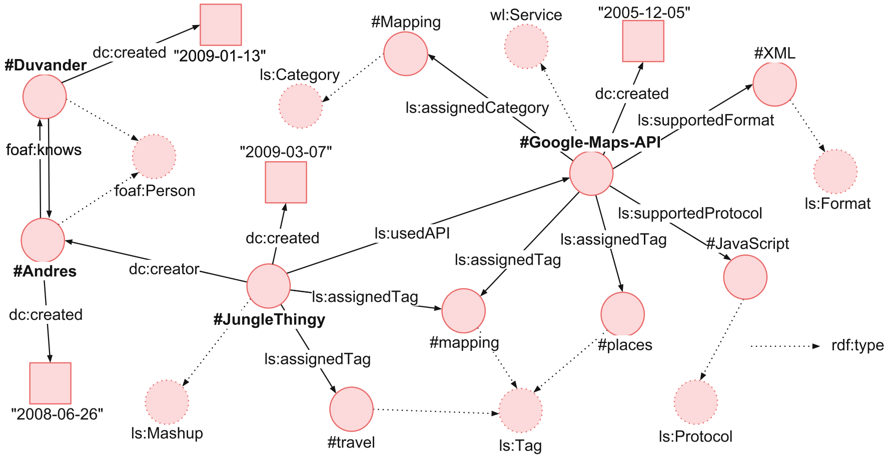


Fig. 3. Excerpt from the extended Linked Web APIs dataset.

The dataset (Fig. 3) uses several well know ontologies and vocabularies: FOAF[4] ontology (*prefix foaf*) - concept *foaf:Person* describes users and property *foaf:knows* describes a social relationship between users, WSMO-lite [7] ontology (*prefix wl*)- concept *wl:Service* describes Web APIs, Dublin Core[5] vocabulary - property *dc:creator* describes relation between a user and a mashup, and property *dc:created* indicates creation date of a mashup, a user or a Web API, SAWSDL [8] vocabulary (*prefix sawsdl*) - property *sawsdl:modelReference* describes a tag or a category of a Web API or a mashup. Additionally, we create new concepts and properties (*prefix ls*): *ls:Protocol* that identifies a protocol, *ls:Format* that identifies data format, and ls:Tag and ls:Category which identify a tag or a category respectively. We also create following new properties: *ls:usedAPI* - between concepts *ls:Mashup* and *wl:Service*, *ls:supportedFormat*, *ls:supportedProtocol* - between concepts *wl:Service* and *ls:Format* or *ls:Protocol*, *ls:assignedTag* and *ls:assignedCategory* - between concepts *wl:Service*/*ls:Mashup* and *ls:Tag*/*ls:Category*.

3.2 Experiments Settings

Implementation. We implemented the proposed method in *R*. It contains functionalities to construct a tensor with temporal aspects, RESCAL factorization algorithm, link prediction method and a running example. The implementation is available under an open source licence on GitHub[6].

Time Information. Our dataset does not contain the time information for each link. Therefore, we derive this information from $< n, dc : created, t_{cn} >$, where

[4] http://xmlns.com/foaf/spec/.

[5] http://dublincore.org/documents/.

[6] https://github.com/jaroslav-kuchar/Time-Aware-Link-Prediction.

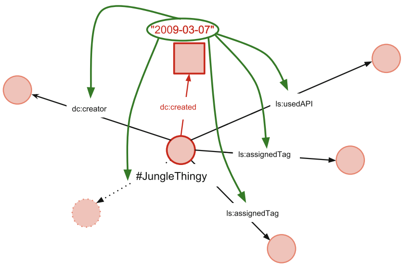


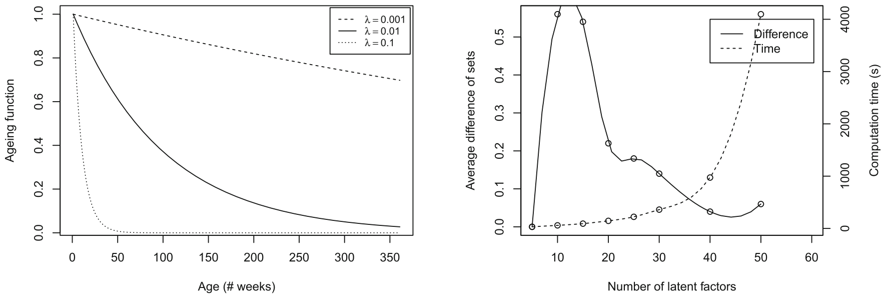
Fig. 4. Example of the propagation of the temporal information from *dc* : *created* to all related links with the same entity.

n represents a mashup, a Web API or a person and t_{cn} denotes the time the entity was created. Since all entity links are created in our dataset at the same time as the entity is created, we propagate t_{cn} as a creation time to all the links of the entity n (Fig. 4).

Snapshots. For purposes of analysing data over different time periods we prepared 22 snapshots of the dataset. The first snapshot contains data from June 2005 until January 2008. It contains approx. 21 000 links which is a significant portion of the total number of links while it is a sufficient information for the link prediction. We then created subsequent snapshots with a step of 3 months where each snapshot always contains the data of a previous snapshot. In order to compare capabilities of the time-aware link prediction and the link prediction that does not use time information we modelled all 22 snapshots as tensors with and without time information. The ageing function parameter t_0 (see Formula (1)) denotes the end of a snapshot.

Setting the Ageing Constant. In the experiments, we set the ageing constant empirically to $\lambda = 0.01$ and the age period t in weeks. Figure 5(a) depicts the influence of the ageing function for different λ. Value $\lambda = 0.01$ provides a distribution of values over the whole seven years period. Note that a higher λ value (i.e. $\lambda = 0.1$) promotes less than the last 50 weeks while a lower λ value (i.e. $\lambda = 0.001$) does not provide significant change of values over the period. This is a configurable parameter that can be used to control the forgetting rate and it depends on specific requirements and dataset. Since we want all data in the dataset to participate in our experiments, the value $\lambda = 0.01$ provides us with the best setting. The results from the evaluation also supports this setting in terms of overall quality of the predictions.

Setting the Tensor Factorization. In the tensor factorization, we experimentally set the number of latent factors to 40. We terminate the factorization when a change of the factor matrices between two iterations is < 1. This is a terminating condition for the minimum optimization problem which means that the solution found during the iteration will not change in subsequent iterations. Figure 5(b) depicts the impact of various settings on the method. We performed 10 runs on the same model and measured the difference of predicted sets of links.



(a) Ageing function

(b) Number of Latent factors

Fig. 5. Experiments settings.

The same figure also illustrates a computation time on a computer with 1,6 GHz Intel Core i5 and 4 GB RAM. Note that in this paper we do not focus on the performance and scalability of the algorithm. We refer the reader to [5] for more details on the performance of the RESCAL factorization.

3.3 Evaluation

In this section, we describe the results from the experimental evaluation where the goal is to measure the quality of the time-aware link prediction. We created two sets, namely a *training set* and a *testing set*, from the whole dataset. We randomly selected 1 % of the newest links from the last snapshot (the last 3 months) and put them to the testing set. The rest of the data we put to the training set. We performed repeated random sub-sampling cross-validation.

We evaluated our method (including different functions and parameters for ageing) compared to the following set of algorithms.

- *Random:* for each source of a link in the testing set, randomly choose a set of targets that correspond to the type of the link. For example, for a *Mashup* and a link *usedAPI* it randomly chooses a set of *Web APIs*.
- *Recent:* select targets from the testing set that are connected to the newest links in the training set.
- *Most Popular:* select targets from the testing set that are connected to the highest number of links in the training set.
- *Regular TB Link Prediction:* a tensor model without ageing and the original RESCAL tensor factorization.
- *Time-aware Link Prediction with Ageing:* our proposed method with different values of λ parameter for ageing function. "*Linear*" decreases importance of older links linearly over the whole time period, "$1 - $*Ageing*" and "$1 - $*Linear*" promotes older links.
- *CP and Tucker:* tensor decomposition CP (CANDECOMP/PARAFAC) and Tucker [1] using tensor model with ageing function and $\lambda = 0.01$.

- *Jaccard and Adamic Adar:* baseline graph based methods for link prediction in social networks [9] that use node neighbourhoods to predict new links.

Note that the *Recent* and *Most Popular* are exploited as recommendation methods in the ProgrammableWeb service repository.

Since we only have one relevant target for each testing item, and we measure a position of a predicted link, we did not perform evaluation related to Precision and Recall. Instead, we measured Mean Reciprocal Rank (MRR), which is appropriate for evaluation tasks with a single target. It is computed as a reciprocal value of a position at which the relevant target was evaluated and is averaged across all testing items (TI): $MRR = \frac{1}{|TI|} \sum_{i=1}^{|TI|} 1/position_i$.

The second metric we evaluate is HitRatio at top-k ($HR@k$) that indicates whether the relevant link occurs in the top-k predicted links. It is computed as $HR@k = \frac{1}{|TI|} \sum_{i=1}^{|TI|} hit_i^k$, where $hit_i^k = 1$ if the relevant link is in top-k predicted links, otherwise it is 0.

Figure 6 shows results from the evaluation. *Random* neither works with structural nor temporal information and has the lowest values for all metrics. *Recent* has slightly better results since it takes into account temporal aspects. Taking into account popularity leads to better results with *Most Popular*. *Regular Link Prediction* has good results since it considers the data structure. *Time-Aware Link Predictions* based on *Linear*, $1 - Linear$ or $1 - Ageing$ do not show better results than the Regular Link prediction since they do not reflect properly temporal aspects of links in the dataset. *Jaccard and Adamic Adar* does not perform well since they consider only information about the closest neighbourhood of each node in graph and they do not take into account types of nodes or semantics of links. *CP decomposition* achieved comparable results with RESCAL in terms of MRR but lower results in HR@k. *Tucker* decomposition has good results since it takes into account structure but does not have better results than *Regular Link Prediction* with RESCAL. Our time-aware link prediction based on RESCAL ($\lambda = 0.01$) outperforms other baseline algorithms in MRR and HR@1, HR@5, HR@10. It is able to predict links on better positions (lower k) than the other algorithms. In the following experiments, we focus on the *Time-Aware Link Predictions* with ageing function ($\lambda = 0.01$).

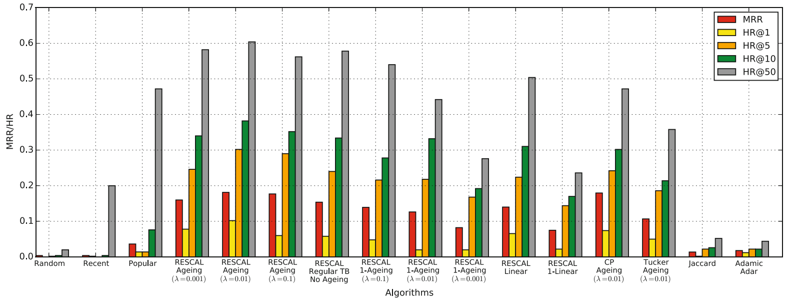
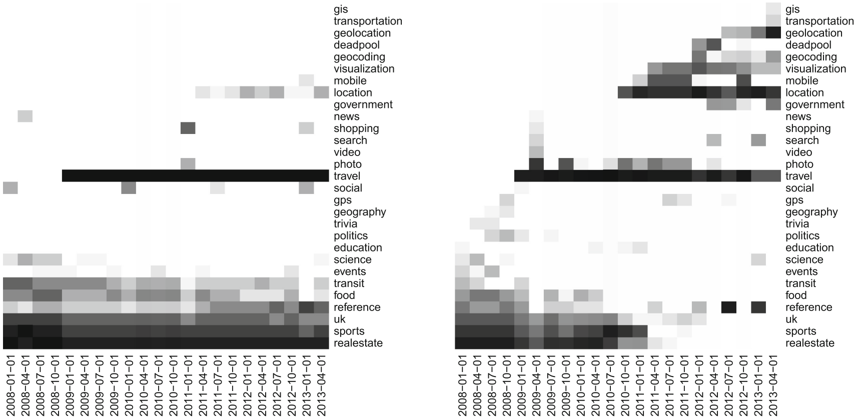


Fig. 6. ProgrammableWeb: Mean Reciprocal Rank (MRR), HitRatio at top-k (HR@k).



(a) Without Ageing (b) With Ageing

Fig. 7. Visualization of positions for each snapshot.

3.4 Significance of Time-Aware Link Prediction

In this experiment we test how the time information influences items and their position in a list of top-L predicted links. To study the influence of time, we focused on a simple tagging task. The goal is to find a set of tags which should be assigned to a specific API (predicted links to tags can be used to improve description of APIs). We run this experiment for the well-known *Google Maps API*.

Table 2 shows results using the tensor models with and without ageing for the last snapshot. The column *Without ageing* contains a list of tags representing targets of predicted top-10 links. This list is influenced only by structural

Table 2. Top 10 tags for *Google Maps API* on the 1st April 2013.

Position	Without ageing	With ageing
1	**travel**	geolocation
2	realestate	**location**
3	sports	**travel**
4	reference	**government**
5	uk	geocoding
6	**location**	visualization
7	transit	transportation
8	food	gis
9	science	weather
10	**government**	deadpool

patterns in the whole dataset, since the snapshot without ageing is used. The column *With Ageing* contains a list of tags, which is not only influenced by structural patterns, but also by time. Some of the predicted tags are the same in both sets, but on different positions. For example *travel* lost the first position, but *location* or *government* moved up to better positions.

In order to explore differences in both sets we run the same experiment over time (i.e., by using the 22 snapshots). Figure 7(a),(b) depict positions of tags in a top-10 set for each snapshot. The position is represented by a color on a scale from white to black where a darker color corresponds to a better position of a tag. Figure 7(a) depicts positions when the ageing is not used. It can be observed that a position of tags do not change very much over time once a tag gets to a certain position (e.g., realestate, travel). This is influenced by global structural patterns that the algorithm uses once they appear in the dataset. Note that each snapshot always contains data of a previous snapshot (see Snapshots paragraph in Sect. 4).

Figure 7(b) depicts positions when the ageing is used. There is a group of tags (food, reference, uk, sports, realestate) that were on better positions in the past (the darker colors in the bottom-left corner), however, they lost significance in recent time. On the other hand, a group of tags (e.g., geolocation, geocoding, location) had no significance in the past but is more preferred in recent time (darker colors in the top-right corner). This is caused by evolution of the structure of the dataset over time. Intuitively, this also proves the fact that mapping APIs and mashups (i.e., tags geocoding, location, geolocation) started to gain a popularity only 5 years ago and travel mashups and APIs are all-time popular. Please also refer to experiment in Sect. 3.6 for more details.

3.5 Influence of Time Information on Prediction

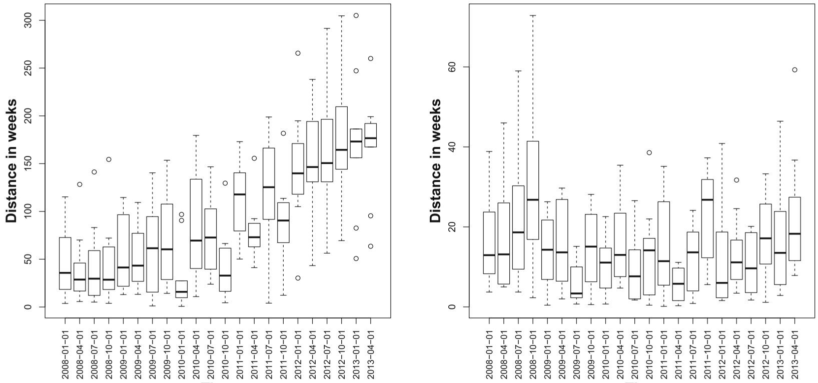
In this experiment, we present a relation of predicted links and time information of entities which participate in the predicted links. This experiment is motivated by a need to predict links between tags and APIs or mashups and APIs. For example, to find top-10 APIs that should also have the tag *mapping* or top-10 mashups that could benefit from the Flicker API.

Table 3 presents the top 10 Web APIs (their names and dates, they were added to directory), which should also have the tag *mapping*. For this experiment we used the models of last snapshots with and without ageing from March 2013. In case the temporal information is not considered, the distances of dates for predicted APIs are higher. With Ageing, the predicted APIs are influenced by time and more recent APIs are predicted.

In the second experiment we performed the following task: find top-10 mashups that could benefit from the Flicker API. We run the experiment for all 22 snapshots. Figure 8(a) and (b) depict a distance in weeks of top 10 mashups from t_0 of every snapshot. We use a standard box plot to examine distributions of distances graphically. Figure 8(b) presents much lower distances than Fig. 8(a). These results support our assumption that predicted entities in top-10 lead to links between entities with time information closer to t_0 (i.e., the present time of a particular snapshot) than the link prediction that does not use time information.

Table 3. Top 10 APIs which should have tag *mapping* on the 1st of April 2013.

	Without ageing		With ageing	
Position	**Name**	**Date**	**Name**	**Date**
1	Placr	2011-07-07	SetGetGo IP Geolocation	2013-12-29
2	BestParking	2011-01-06	JetSetMe	2012-11-11
3	Tube Updates	2011-03-27	Frontier Airlines Word Wheel Local	2013-02-10
4	ParkWhiz	2010-09-30	Eaupen	2012-12-29
5	Eaupen	2012-12-29	DC Location Verifier	2012-10-17
6	NAVTEQ Traffic	2011-09-11	TripCheck	2013-03-02
7	Jeppesen Journey Planner	2011-10-26	View	2013-03-01
8	View	2013-03-01	WikiSherpa	2012-07-06
9	MyTTC	2011-08-13	ThinkGeo Cygnus Track	2012-10-26
10	Pearson Eyewitness Guides	2011-09-16	weather-api.net	2012-12-19



(a) No Ageing

(b) Ageing

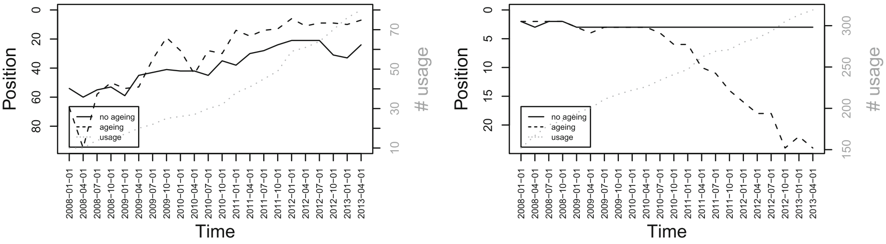
Fig. 8. Distance of predicted Mashups from the ending time of snapshot.

We also performed a quantitative experiment of this prediction task. We randomly selected 100 tags and predicted top-10 APIs that should be assigned to each tag. At the same time we randomly selected 1000 Mashups and predicted top-10 APIs which should be used in the specific Mashup. The mean value of distance is 33 weeks for the time-aware link prediction and 184 weeks for the link prediction that does no use time information.

3.6 Impact of Evolution of Structure

In this experiment, we demonstrate how the proposed method takes into account the evolution of the datasets' structure when predicting new links.

We run the prediction for two tags *realestate* and *geocoding* and evaluate their positions in top-L predicted links over time for the well-known *Google Maps API*. Figure 9(a) and (b) depict an evolution of the position for both tags on the left axis and a number of usages of the tags on the right axis (a usage of a tag means that an explicit link between an entity and the tag exists in the dataset).



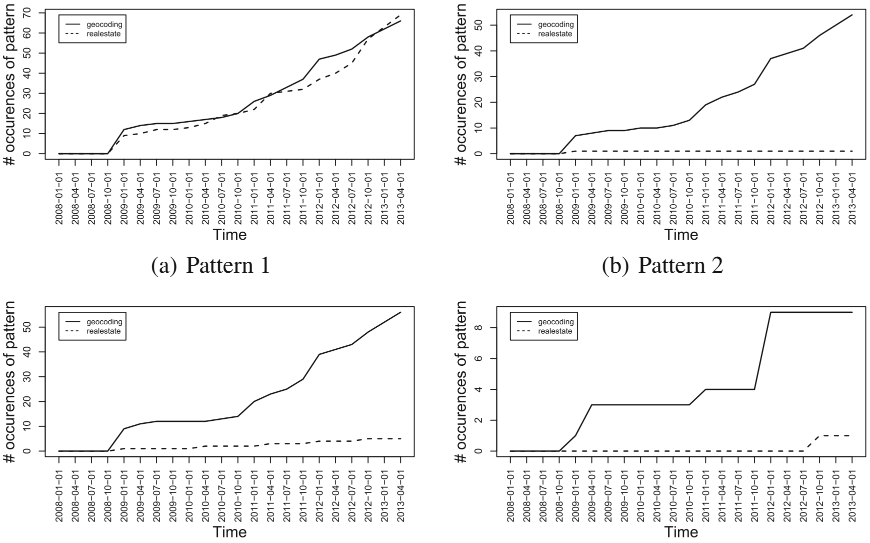
(a) Position of tag *geocoding* (b) Position of tag *realestate*

Fig. 9. Evolution of position over time for a specific tag.

Figure 9(b) shows a high position of the tag *realestate* when no ageing is used. This is influenced by the high number resources (APIs and mashups) tagged with this tag and the supporting structural patterns that exist throughout the history. However, when ageing is applied, the tag is gradually loosing its position since the structural patterns were created earlier in the past rather than in recent time (in a snapshot's time t_0). Figure 9(a) shows that the tag *geocoding* gets to slightly better positions when ageing is applied. This is caused by the fact that supportive structural patterns for this tag appeared in recent time. The next paragraph describes an example of elementary structural patterns that may influence positions of tags in link prediction.

Significant Sub-graphs. Our method is based on identification of hidden patterns in the structure of data (tensor factorization) in connection to the time information and ageing. Identified hidden patterns are used to predict new links in data. In order to find such significant patterns we can use an existing local property of graphs, called motifs. Motifs are defined as recurrent and statistically significant sub-graphs. We adopted the idea of motifs in this experiment as an "evidence" of influence of structure and temporal information in tensor factorization with ageing. The goal of this experiment is to some extent provide an explanation of results from the previously described experiment in this section.

New links for *Google Maps API* can be predicted only when a similar pattern exists in the data and the pattern contains information related or similar to the *Google Maps API* structure. Based on the dataset structure, we define several



(a) Pattern 1

(b) Pattern 2

(c) Pattern 3

(d) Pattern 4

Fig. 10. Number of occurrences for each pattern.

elementary patterns which may influence the link prediction of the tags *realestate* and *geocoding* for the *Google Maps API*. By looking at the *Google Maps API* structure, we can see that it is a service, it has assigned a category mapping, a tag mapping, and supports JavaScript protocol. We breakdown this structure to the following queries (that we call patterns), where X can be either *realestate* or *geocoding*. We then measure the number of occurrences for each of the 8 patterns in the 22 snapshots.

1. *?var rdf:type wl:Service AND*
   ?var ls:assignedTag ?X
2. *?var ls:assignedCategory ls:Mapping AND*
   ?var ls:assignedTag ?X
3. *?var ls:assignedTag ls:mapping AND*
   ?var ls:assignedTag ?X
4. *?var ls:supportedProtocol ls:JavaSript AND*
   ?var ls:assignedTag ?X

Figure 10(a–d) depict a number of occurrences for each pattern over time (i.e., for each of the 22 snapshots). The tag *geocoding* has a higher number of occurrences of the patterns than the tag *realestate*. This means that there are more structures similar to the Google Maps API structure that have assigned tag *geocoding* rather than the tag *realestate*. Although this does not provide much evidence for the tag *realestate* and its high positions when no ageing is used (in Fig. 9(b)) which is influenced by other structural patterns not shown here, it shows that a higher presence of the patterns in recent time promotes the tag *geocoding* to better positions when compared to positions when no ageing is used (Fig. 9(a)).

3.7 Evaluation on Other Datasets

For evaluation purposes of the proposed method, we prepared another graph-based RDF dataset. We focused on a subset of data from the well-known knowledge base *DBpedia*[7].

Movies Dataset. We selected a subset of movies from the DBpedia according to existing mappings [10] for the *MovieTweetings* dataset [11]. A movie ratings dataset extracted from tweets on Twitter. The mappings dataset contains links for over *15 000* movies. We extracted from the DBpedia following main information for each movie: assigned types and categories, actors starring in a movie, distributors, music composers, producers, writers and directors. Those information also represent eight types of links in a complete dataset. In total, there are over *66 600* unique entities and more than *280 000* links in the dataset. As temporal information we use the release date of each movie and we propagate it as a creation time to all links associated to a movie.

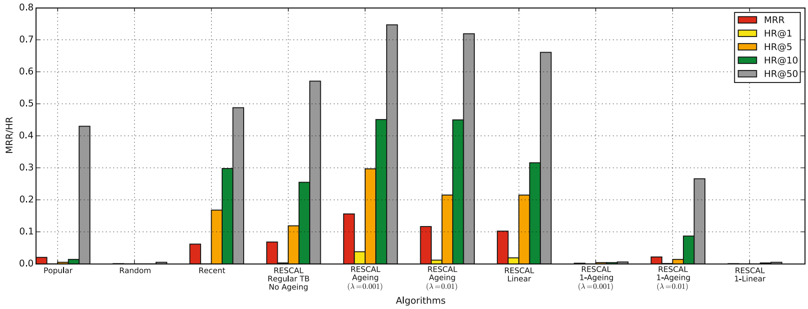


Fig. 11. DBpedia Movies: Mean Reciprocal Rank (MRR), HitRatio at top-k (HR@k).

Evaluation Settings. We set the ageing constant to $\lambda = 0.001$, since this value provides a distribution of values over the whole period of release dates since *1900s* till present 2015. The number of latent factors was experimentally set to *30*. The value *30* provides best results from the perspective of used evaluation metrics. As evaluation metrics we used both Mean Reciprocal Rank (*MRR*) and Hit Ratio at top-k (*HR@k*). On this dataset we focused on the following algorithms: *Random*, *Recent*, *Most Popular*, *Regular TB Link Prediction* and *Time-aware Link Prediction with Ageing*. *Training set* and *Test set* were created similarly to the previous evaluation: we put randomly selected 1 % of newest links (from year 2015) to the testing set. The rest of the data we put to the training set. We performed repeated random sub-sampling cross-validation.

[7] http://dbpedia.org.

Results of Evaluation. Figure 11 depicts results of the evaluation on movies from the DBpedia. Both *Random* and *Most Popular* does not perform well, since they do not consider structure and temporal dimension. *Recent* takes into account time and has significantly better results than *Random* and *Most Popular*. *Regular TB Link Prediction* does not reflect creation time. It has slightly better results in terms of *MRR* but does not provide better results for *HR@k*. *1-Ageing* and *1-Linear* dos not perform well because of promoting older links. *Ageing (*$\lambda = 0.01$*)* and *Ageing (Linear)* outperforms the regular link prediction, but the distribution of values is not over the whole interval and does not follow forgetting curve, respectively. Our proposed method *Ageing (*$\lambda = 0.001$*)* outperforms all other approaches. It takes into account both structural and temporal information.

4 Discussion

Robustness. Although we evaluated our method on a domain-specific datasets from ProgrammableWeb and DBpedia, the method is capable to predict links in a dataset from any other domain. We have chosen the dataset from ProgrammableWeb as it contains sufficient information about creation time of entities that we can propagate to relevant links. Similarly, we have selected the movies from DBpedia since the release date can be used as the creation time for the proposed method.

Temporal Information. Due to the nature of the data from ProgrammableWeb and DBpedia we deal with a specific form of time assigned to an entity as the created time (see also Sect. 3 for information how we propagate this time to corresponding links). We understand the created time as a starting time from which the link exists in the dataset and we have no information about a duration of the link's existence. It is our future work to study various representations of time in linked datasets and incorporate them into the time-aware link prediction method (e.g. presence of termination time of a link).

Further, there are two basic types of expressing an existence of data - an explicitly defined *time point* using a document-centric and a fact-centric information (e.g., reification, N-ary relationships, snapshots of graphs, provenance, PROV-O, Memento etc.) [3] or deduced from other facts in an RDF dataset. The first category can be immediately used in our model. Since the availability of temporal information in Linked Data is still limited [3], especially for links, we derive the temporal restrictions from available data in dataset.

The types of links that never evolve or should not evolve (e.g. *dc:creator*, *rdf:type*) can be excluded from the temporal extension of tensor using value 1.

Ageing Function. Our goal was to show that time information is a very important aspect for link prediction and how a method to predict links can be extended with time information by modelling a retention of information using the ageing function. The formula we use for the ageing function is inspired by a representation of forgetting and retention mechanisms in the human mind. We have demonstrated that the proposed ageing function outperforms other formulas. However, the formula may vary in different use cases and domains.

Structural Patterns. Results of the time-aware link prediction highly depend on a structure of the RDF graph and a time when links were created. In Sect. 3.6 we identified simple structural patterns that may influence the link prediction in this specific dataset. However, there is no reason to assume that there cannot be present also other, more complex structural patterns that influence the link prediction. In our future work we plan to explore methods for automatic detection of more complex patterns.

Snapshot Creation. We have chosen the size of snapshots so that they have a sufficient amount of data for learning. Note that the data of some snapshots can be differently distributed with respect to time. Some snapshots might have data normally distributed but in some snapshots the majority of the data can be at the start or at end of the snapshot. Such distribution of the data has an impact on the link prediction.

5 Related Work

There are two main topics closely related to our time-aware link prediction method, namely tensor factorization and relational learning. The models and methods covered by these topics are used to model multi-relational data and to perform the link prediction.

Most researches in relational learning are based on a statistical relational learning. These approaches are build upon the Bayesian or Markov networks [12,13] or their combinations with tensor representations [14].

There is a growing interest in tensor models and factorizations in multi-relational data modelling. An overview of tensor factorizations and their applications is in [1]. There are two basic approaches, namely link-information-based approaches and node-information-based approaches. We adopted a model from link-information-based approaches by [5], where each frontal slice of a tensor represents a relation. A similar model was also used in [15]. These modelling approaches, however, do not work with time information. They only take into account entities and relations among them.

On the other hand, node-information-based approaches, take into account attributes of entities [16,17]. An extension of this work in [6] is able to work with attributes (time attribute can also be included) and combine both approaches.

There are also existing approaches related to frameworks LIMES [18] and SILK [19] that are focused on link discovery between different datasets. Our approach is focused on link prediction within one dataset.

There are existing researches, that use time for predicting links. In [20], the authors use the third-order tensor factorization, where two dimensions are used to represent relations and the third dimension represents time. This approach is however suitable only for one type of relation. A similar work was done in [21–23] where authors also work with a dataset with one type of a relation.

There are also other approaches that use either multi-modal representation of graph or temporal information for link prediction in Social Networks, e.g. prediction links in asynchronous communication [9], prediction based on hypergraph [24], prediction in multi-modal networks [25], however, they are less relevant to our work.

6 Conclusion and Future Work

Despite many existing link prediction algorithms, there is still lack of approaches that are focused on multiple types of links and time information about existence of links. Rich graph-based datasets, including RDF, can contain links that were included in the past. Such links may loose their significance in the future. The creation time of a link thus provides a meaningful information that might reflect the credibility of the link. In this paper we present a method for a link prediction task that considers both structural and temporal aspects of graph-based datasets - *Time-Aware Link Prediction*. Our method incorporates the creation time of links to a tensor model and we use the tensor factorization as an underlying concept for the link prediction. We evaluated our method on the RDF representation of the ProgrammableWeb directory and a subset from the well-known knowledge-base DBpedia. Furthermore, we also performed a set of experiments to explore the details and demonstrate the capabilities of the method. The results from the evaluation and experiments show that the temporal dimension influences the results and it is an important aspect for the link prediction.

For future work, we plan to focus on the evaluation of the method on other datasets from Linked Data Cloud that incorporate links across data sources. We also want to address the presence of other temporal information, including mainly the termination time of a link. Further, we plan to explore different applications of the method. In particular, we want to evaluate the performance of the method in evaluation of existing links.

Acknowledgements. This work was supported by the Grant Agency of the Czech Technical University in Prague, grant No. SGS14/104/OHK3/1T/18. We also thank to ProgrammableWeb.com for supporting this research.

References

1. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. SIAM Rev. **51**, 455–500 (2009)
2. Ebbinghaus, H.: Memory: A Contribution to Experimental Psychology. Teachers College, Columbia University, New York (1913). Number 3
3. Rula, A., Palmonari, M., Harth, A., Stadtmüller, S., Maurino, A.: On the diversity and availability of temporal information in linked open data. In: Cudré-Mauroux, P., et al. (eds.) ISWC 2012, Part I. LNCS, vol. 7649, pp. 492–507. Springer, Heidelberg (2012)
4. Gutiérrez-Basulto, V., Klarman, S.: Towards a unifying approach to representing and querying temporal data in description logics. In: Krötzsch, M., Straccia, U. (eds.) RR 2012. LNCS, vol. 7497, pp. 90–105. Springer, Heidelberg (2012)

5. Nickel, M., Tresp, V., Kriegel, H.P.: A three-way model for collective learning on multi-relational data. In: Getoor, L., Scheffer, T. (eds.) Proceedings of the 28th International Conference on Machine Learning (ICML-11), ICML 2011, pp. 809–816. ACM, New York (2011)
6. Nickel, M., Tresp, V., Kriegel, H.P.: Factorizing yago: scalable machine learning for linked data. In: Proceedings of the 21st International Conference on World Wide Web, WWW 2012, pp. 271–280. ACM, New York (2012)
7. Vitvar, T., Kopecký, J., Viskova, J., Fensel, D.: WSMO-Lite annotations for web services. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 674–689. Springer, Heidelberg (2008)
8. Kopecky, J., Vitvar, T., Bournez, C., Farrell, J.: Sawsdl: semantic annotations for wsdl and xml schema. IEEE Internet Comput. **11**, 60–67 (2007)
9. Oyama, S., Hayashi, K., Kashima, H.: Cross-temporal link prediction. In: Proceedings of the 2011 IEEE 11th International Conference on Data Mining, ICDM 2011, pp. 1188–1193. IEEE Computer Society, Washington, D.C. (2011)
10. Kuchar, J.: Augmenting a feature set of movies using linked open data. In: Proceedings of the RuleML 2015 Challenge, the Special Track on Rule-based Recommender Systems for the Web of Data, the Special Industry Track and the RuleML 2015 Doctoral Consortium hosted by the 9th International Web Rule Symposium (RuleML 2015), Germany, Berlin, 2–5 August 2015
11. Dooms, S., De Pessemier, T., Martens, L.: Movietweetings: a movie rating dataset collected from twitter. In: Workshop on Crowdsourcing and Human Computation for Recommender Systems, CrowdRec at RecSys 2013 (2013)
12. Friedman, N., Getoor, L., Koller, D., Pfeffer, A.: Learning probabilistic relational models. In: IJCAI, pp. 1300–1309. Springer, Heidelberg (1999)
13. Khosravi, Hassan, Bina, Bahareh: A survey on statistical relational learning. In: Farzindar, Atefeh, Kešelj, Vlado (eds.) Canadian AI 2010. LNCS, vol. 6085, pp. 256–268. Springer, Heidelberg (2010)
14. Gao, S., Denoyer, L., Gallinari, P.: Probabilistic latent tensor factorization model for link pattern prediction in multi-relational networks. CoRR abs/1204.2588 (2012)
15. London, B., Rekatsinas, T., Huang, B., Getoor, L.: Multi-relational learning using weighted tensor decomposition with modular loss. CoRR abs/1303.1733 (2013)
16. Taskar, B., fai Wong, M., Abbeel, P., Koller, D.: Link prediction in relational data. In: In Neural Information Processing Systems (2003)
17. Raymond, R., Kashima, H.: Fast and scalable algorithms for semi-supervised link prediction on static and dynamic graphs. In: Balcázar, J.L., Bonchi, F., Gionis, A., Sebag, M. (eds.) ECML PKDD 2010, Part III. LNCS, vol. 6323, pp. 131–147. Springer, Heidelberg (2010)
18. Ngomo, A.C.N., Auer, S.: Limes: a time-efficient approach for large-scale link discovery on the web of data. In: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, IJCAI 2011, vol. 3, pp. 2312–2317. AAAI Press (2011)
19. Bizer, C., Volz, J., Kobilarov, G., Gaedke, M.: Silk - a link discovery framework for the web of data. In: 18th International World Wide Web Conference (2009)
20. Spiegel, S., Clausen, J., Albayrak, S., Kunegis, J.: Link prediction on evolving data using tensor factorization. In: Cao, L., Huang, J.Z., Bailey, J., Koh, Y.S., Luo, J. (eds.) PAKDD Workshops 2011. LNCS, vol. 7104, pp. 100–110. Springer, Heidelberg (2012)

21. Acar, E., Dunlavy, D.M., Kolda, T.G.: Link prediction on evolving data using matrix and tensor factorizations. In: Proceedings of the 2009 IEEE International Conference on Data Mining Workshops, ICDMW 2009, pp. 262–269. IEEE Computer Society, Washington, D.C. (2009)
22. Dunlavy, D.M., Kolda, T.G., Acar, E.: Temporal link prediction using matrix and tensor factorizations. ACM Trans. Knowl. Discov. Data **5**, 10:1–10:27 (2011)
23. Ermis, B., Acar, E., Cemgil, A.T.: Link prediction via generalized coupled tensor factorisation. CoRR abs/1208.6231(2012)
24. Li, D., Xu, Z., Li, S., Sun, X.: Link prediction in social networks based on hypergraph. In: Proceedings of the 22nd International Conference on World Wide Web Companion, WWW 2013 Companion, Republic and Canton of Geneva, Switzerland. International World Wide Web Conferences Steering Committee, pp. 41–42 (2013)
25. Symeonidis, P., Perentis, C.: Link prediction in multi-modal social networks. In: Calders, T., Esposito, F., Hüllermeier, E., Meo, R. (eds.) ECML PKDD 2014, Part III. LNCS, vol. 8726, pp. 147–162. Springer, Heidelberg (2014)

Extracting Product Offers from e-Shop Websites

Andrea Horch(✉), Holger Kett, and Anette Weisbecker

Fraunhofer Institute for Industrial Engineering IAO, Stuttgart, Germany
andrea.horch@iao.fraunhofer.de
https://www.iao.fraunhofer.de/

Abstract. On-line retailers as well as e-shoppers are very interested in gathering product records from the Web in order to compare products and prices. The consumers compare products and prices to find the best price for a specific product or they want to identify alternatives for a product whereas the on-line retailers need to compare their offers with those of their competitors for being able to remain competitive. As there is a huge number and vast array of product offers in the Web the product data needs to be collected through an automated approach. The contribution of this papers is a novel approach for automatically identify and extract product records from arbitrary e-shop websites. The approach extends an existing technique which is called Tag Path Clustering for clustering similar HTML tag paths. The clustering mechanism is combined with a novel filtering mechanism for identifying the product records to be extracted within the websites.

Keywords: Web data extraction · Product record extraction · Tag path clustering

1 Introduction

E-commerce is a vast and growing market. In 2014 the turnover of Europe's e-commerce increased by 16.3 % to 363.1 billion Euros and the number of on-line retailers in Europe was estimated to be 640,000 [1]. According to [2] price reductions compared to classical advertising activities are more efficient and can be realised significantly easier and faster. The authors of [3] show that one of the success factors for on-line retailers is knowing the prices of the competitors and being able to adjust the own prices.

The rising number of on-line retailers leads to an increasing number of on-line product offers. Handling the comparison of such an amount of products and the corresponding price data is hardly manageable on a manual basis. Thus, on-line retailers as well as consumers need software support for automatically comparing products and prices. Such software tools need to automatically identify, extract and structure the product and price information on the e-shop websites for comparing prices and displaying the analysed data to the users. Automatically identifying and extracting product and price information from arbitrary e-shop websites is a very challenging task as different e-shops are selling a variety

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 232–251, 2016.
DOI: 10.1007/978-3-319-30996-5_12

of products of different domains and there are various types of e-shop software using differently structured templates for displaying the product information. The first and one of the most important steps when gathering product and price information from the Web is the identification and extraction of the single product records within the e-shop website.

This paper proposes a novel approach, called LightExtraction, for identifying and extracting product records from arbitrary e-shop websites. The approach was built up by following the idea of Tag Path Clustering as presented in [4,5] and simplifying the filtering steps by exploiting the need for using common attributes to display product descriptions on e-shop websites like product name, product image and product description.

The paper is structured as follows: In Sect. 2 we present the related work. Section 3 shows the results of the analysis of a set of product records of 30 different e-shop websites. Section 4 introduces our novel approach. We demonstrate the adequacy of our approach through an experiment and highlight its results in Sect. 5, and we conclude in Sect. 6.

2 Related Work

There are several existing approaches in science and practice for extracting data records from the Web.

Tools like Dapper[1], Kimono[2] or import.io[3] can be used to extract data directly from a website. Before such a tool is able to extract the relevant data it needs to be configured. The configuration is made manually by a graphical user interface, e.g. an integrated browser, where the navigation steps to reach the page, which includes the data of interest, have to be simulated in the graphical interface and the data for the extraction has to be marked.

Another interesting tool for Web data extraction is Crawlbot[4]. Crawlbot offers a Web service as well as an API for crawling product price data, historical weather data or news articles from the Web. For the automated extraction Crawlbot needs the URL (Uniform Resource Locator) of the product or article to be scraped. Crawlbot analyses the website of the given URL, structures it into its attributes (e.g. product name, product price) and returns the attributes of the product or article in a well structured format. The problem with using Crawlbot to get the structured product descriptions of a whole e-shop is that the URL of each product detail page has to be defined as input since Crawlbot cannot handle pages including more than one product record. Thus, for obtaining the URLs of the product detail pages another automated mechanism is needed.

Over the years many scientific approaches to automatically identify and extract data records from the Web have been developed. Some of the approaches

[1] http://open.dapper.net/.

[2] https://www.kimonolabs.com/.

[3] https://import.io/.

[4] http://www.diffbot.com/products/crawlbot/.

are based on machine learning, others are based on phrase analysis or Tag Path Clustering. There are also hybrid methods, which rely on several techniques.

The most popular scientific approaches for automatically detecting and extracting data records from websites are the MDR (Mining Data Records in Web Pages) algorithm described in [6] and the ViNTs (Visual information aNd Tag structure based wrapper generator) tool introduced in [7].

The MDR algorithm compares the child nodes of each node in an HTML tree starting at the root node for discovering data regions inside a Web page. The node comparison is done either by calculating the string edit distance (e.g. Levenshtein distance) or by a tree matching algorithm (e.g. Simple Tree Matching). The similarity of nodes is defined by a preset threshold. Through this procedure the algorithm searches for similar child node combinations in each node. A node containing several similar child nodes is considered as a data region including a set of data records. MDR traverses only the trees of nodes which are not covered by already identified data regions and which include at least three child nodes. The similar-structured child nodes of a data region are the data records.

The MDR algorithm is very useful when searching for the data regions inside a website, but another approach is needed for identifying the data region of a website containing the relevant data records for extraction.

ViNTS is a tool for automatically generating a wrapper for extracting search result records of an arbitrary search engine. For building a wrapper for a search engine ViNTS needs some sample result pages and an empty result page of the search engine as input. ViNTS renders the sample result pages and removes all content lines, which also appear in the empty result page, in order to remove all irrelevant content. On the sample result pages ViNTS identifies some candidate search result records as sample input for the wrapper generation step. The candidate search result records are detected by three steps. In the first step the Candidate Content Line Separators, which are HTML tags like <*p*> or <*tr*> separating single search result records, have to be determined. For this purpose ViNTS translates the content lines of the HTML tree into a pair of type code and position code. The type code specifies the content type of the content line like, for example, *text*, *link-text* or *link*, whereas the position code represents the left x coordinate of the rendering box of the content line. All pairs (type code, position code), which can be found at least three times inside the HTML tree, are considered as probable Candidate Content Line Separators. In [7] this step is done by a suffix tree. In the next step the search result page is segmented into multiple content line blocks by using the Candidate Content Line Separators. Consecutive blocks are grouped by their visual similarity with regard to a preset threshold. The visual similarity is calculated by the type distance, shape distance and position distance. The type distance of blocks specifies their edit distance (e.g. Levenshtein distance) of their type codes. The shape distance measures the difference between the indention sequences of the shapes of the blocks. The position distance of two blocks defines the difference between their closest points to the left boundary of the search result page. In the last step of the candidate record detection the first line of every record becomes identified by a set of four predefined heuristic rules: (1) the line following an <*hr*>-tag, (2) the only line in the block starting with a number, (3) the only line in the block having the smallest position code or (4) the only line in the block following a blank line. After having identified the sample candidate result records ViNTS builds the wrapper. For this purpose ViNTS determines the tag paths beginning at the root node of the result page (<*html*>-tag) for each identified first line element. The minimal sub-tree of the result page, including all search result records, is calculated based on the tag paths. The search result records are sub-trees of the result page, which are siblings and have the same or a similar tag structure. These sub-trees can be separated by a separator fulfilling the following conditions: (1) common subset of the sub-trees of all records, (2) occurs only once in a sub-tree of each record and (3) contains the rightmost sub-tree of each result record. There can be several separators for a dataset. The wrapper is built by using the smallest tag path for detecting the data region including the search result records and the separators to separate the result records within the data region.

ViNTS needs sample result pages and an empty result page as input, which can be difficult when extracting product records from e-shop websites since there is usually no empty result page which can be used.

[8] present an approach for extracting structured product specifications from producer websites. For the retrieval of the product specification the algorithm locates the product detail page on the producer's website and extracts and structures the product attributes of the product specification. For searching the producer's page with the product specification [8] process keyword-based Web search by using the popular search engines Google, Bing and Yahoo. After the Web search step [8] rank the results by using a method called "Borda ranking" described in [9] followed by the analysis of the page URI, the page title and the page content based on domain specific terms for finding the producer site within all candidates which were found by the Web searches. For extracting the product data in form of key-value pairs [8] execute three different wrapper induction algorithms on the product detail page. Each of the three algorithms cluster the HTML nodes, which contain text to a node cluster as a first step. The first algorithm is chosen if there are example key phrases provided as input. The algorithm filters the clusters created in the first step of the nodes, which contain the example phrases. The XPath description of the nodes is used for wrapper generation. If no key phrases are provided as input the second algorithm is used, which exploits domain knowledge from already stored product data as key phrases to find the relevant nodes in the cluster for generating the wrapper. If there are neither example key words nor domain knowledge provided as input the third algorithm generates the wrapper from training sets, which are product pages of related products. In the last step the key-value pairs are extracted by text node splitting based on identifying separators like a colon in the text nodes.

The problem when using the approach of [8] for automated product record extraction is that example product data for arbitrary product domains is required, which has to be given by the users in the form of key phrases, or which must be provided from the system as knowledge. For obtaining good results, the key phrases provided by the users or system must fit the phrases of the product detail page of the producers, otherwise the approach will not work. Additionally, numerous steps and different algorithms are needed for the data extraction task.

The approach proposed in [10] for extracting product records from the Web is based on Visual Block Model (VBM) a product of the HTML tag tree and the Cascading Style Sheets (CSS) of a Web page. The VBM is created by the rendering process of a layout engine like WebKit[5]. [10] filter the basic blocks of the page, which are blocks containing other visual blocks. In the next step the similarity of the basic blocks is defined by calculating the visual similarity, the width similarity and the block content similarity. Blocks are visually similar since all of their visual properties are the same. Width similarity of blocks is given if their width properties are within a 5 pixels threshold of each other. The block content similarity exists if the blocks include similar child blocks, which is calculated by using Jaccard index described in [11] and a preset similarity threshold. For the product record extraction [10] select a seed candidate block, which is a single basic block. The seed block is identified by selecting a visual block in the centre of the page and tracing the visual blocks around that block by moving clockwise in the form of a Ulam Spiral[6] until reaching a basic block which is taken as seed block. The seed block is within one or more container blocks where one is assumed to be a data record block. Thus, all of them are taken as candidate blocks. Clusters for all of the candidate blocks are created based on the calculation of block content similarity to all blocks in the VBM. The cluster including the maximum number of container blocks is taken as the cluster containing the product records.

The approach of [10] depends on the selection of a correct seed block. For the selection of the seed block the algorithm starts in the page centre and moves clockwise in the form of a Ulam Spiral to identify a basic block including product information. The clockwise direction was chosen so as to not reach the edge of the page or include noisy features like a left menu. The algorithm follows the assumption that the page menu appears on the left side of the page. But this assumption is not correct for pages of the Arab world like bestarabic.com[7] where the page menu is located on the right side of the page. Thus, the approach can fail for pages with a non-standard page structure.

Another approach called ClustVX described in [4,5] is based on clustering XPaths and CSS elements of the HTML elements in the DOM tree of a Web page. The proposed extraction process takes the Web page rendered by a Web browser and starts pre-processing. The pre-processing comprises the embedding of visual features into the element attributes, the transforming of the HTML code into valid XHTML and the removing of text formatting elements (e.g. <*i*> or <*b*>). After the pre-processing an XPath string enriched with visual information (e.g. font, font-size, font-colour), called Xstring, is generated for each element of the page tree. The elements are clustered by Xstring similarity,

[5] http://www.webkit.org/.

[6] http://mathworld.wolfram.com/PrimeSpiral.html.

[7] http://www.bestarabic.com/mall/ar/.

that means elements having the same Xstring belong to the same cluster. For identifying the data region of the elements in the cluster the longest common XPath prefix of all elements in the cluster is calculated. In order to segment data records the approach identifies if each data record of a data region has its own parent node or if all data records of the region are under the same parent node, which is done by comparing the XPath strings of the elements beginning after the longest common XPath prefix. If each element has its own parent node the data records are the children of the longest common tag path node. In the case that all data records have the same parent node, the approach uses a technique called "HTML tree hopping". The HTML tree hopping technique searches the first data item in the first data record of the data region, then searches the first item of the second data record. The separator of the data records can be found in the HTML tree above the first item of the second data record and can be used to separate all data records of the data region. For determining the importance of the data regions in order to detect the data region including the relevant page data [5] calculate the visual weight of each data region. The visual weight of a data region is the product of the average area of one data record and the square of the number of data items. The data records of the most important data region are extracted by collecting the elements of the identified tag paths from the HTML tree.

ClustVX is suitable for identifying and extracting data records from arbitrary Web pages, but it includes many different process steps which are not necessary. Using the novel approach proposed in this paper the data records of a Web page can be identified and extracted by a few steps. As we prove in Sect. 5 our novel approach achieves as good results as the approaches of [5,6].

3 Product Record Analysis

In order to develop an algorithm for identifying and extracting product records from arbitrary e-shop websites we have analysed the element structure of product records of 30 different e-shop websites.

The selected e-shop websites comprise a wide variety of product categories, various page structures as well as different languages, diverse character sets and different currencies as there were e-shop websites selected from the United States of America, the United Kingdom, Spain, Greece and Germany.

For each of the e-shop websites the product records of a randomly selected product overview page was analysed. The selected websites, the criteria and the collected data for the analysis of the product records as well as the result data for the assay are shown in Table 1.

In the rows of the table the following data for the selected e-shop websites can be found:

- Column 1: The URL of the e-shop websites.
- Column 2: The number of analysed product records which corresponds to the number of product records available on the selected product overview page of the e-shop website.

Table 1. Product record analysis.

URL	Number of analysed product records	Tag name of product record element	Most frequent number of parents	Min. number of parents	Max. number of parents	Most frequent number of children	Min. number of children	Max. number of children	Most frequent number of img elements	Min. number of img elements	Max. number of img elements	Most frequent number of anchor elements	Min. number of anchor elements	Max. number of anchor elements	Average length of included text
http://de.vila.com/	10	div	11	11	11	41	25	76	6	2	20	5	3	12	609
http://raanthai.co.uk/	72	div	19	19	19	28	27	29	1	1	1	4	4	4	129
http://www.bestbuy.com/	14	div	7	7	7	158	126	158	1	1	1	15	14	15	1,056
http://merseyfuels.co.uk/	10	tr	11	11	11	13	13	13	1	1	1	2	2	2	170
http://www.zazzle.de/	60	div	11	11	11	8	8	8	1	1	1	2	2	2	45
http://www.e-shop.gr/	9	table	19	19	19	25	20	25	3	3	3	3	2	3	305
https://www.hairshop-pro.de/	24	div	10	10	10	21	21	21	1	1	1	3	3	3	102
http://www.my-hairshop.de/	10	li	10	10	10	20	20	20	1	1	1	4	4	4	167
http://www.basic-hairshop.de/	15	li	10	10	10	11	11	11	1	1	1	3	3	3	62
http://www.powells.com/	25	li	9	9	9	15	15	23	1	1	6	3	3	3	61
http://heyshop.es/	42	div	8	8	8	6	6	8	1	1	1	1	1	1	33
http://us.nextdirect.com/	24	div	9	9	9	8	8	8	1	1	1	2	2	2	35
http://www.media-dealer.de/	19	form	9	9	9	41	41	45	2	2	3	4	4	4	471
http://www.thinkgeek.com/	11	div	8	8	8	6	6	9	1	1	1	1	1	1	53
http://www.flaconi.de/	17	div	7	7	7	15	15	17	1	1	1	3	3	3	85
http://www.mrwonderfulshop.es/	16	li	10	10	10	13	13	13	2	2	2	2	2	2	71
http://www.dutyfreeshops.gr/	15	div	7	7	7	18	18	20	1	1	2	4	4	4	77
http://www.sammydress.com/	60	li	8	8	8	14	13	17	1	1	1	3	3	3	98
http://www.fragrancenet.com/	17	section	9	9	9	20	20	20	1	1	1	3	3	3	86
http://www.perfume.com/	20	div	10	10	10	10	10	10	1	1	1	2	2	2	47
http://www.sunglasshut.com/	13	div	14	14	14	47	39	47	2	2	2	8	7	8	1,740
http://surrealsunglasses.es/	18	li	10	10	10	39	39	42	2	2	2	6	6	6	65
http://www.smartbuyglasses.gr/	44	ul	11	11	11	17	15	20	1	1	1	4	3	4	58
http://zyloeyewear.com/	33	div	9	9	9	10	10	10	1	1	1	2	2	2	45
http://www.tokotoukan.com/	17	div	8	8	8	9	9	18	1	1	1	3	3	6	82
http://www.adidas.de/	44	div	13	13	13	28	28	56	1	1	7	5	3	7	97
http://batterypark.gr/	18	div	12	12	12	58	56	58	1	1	1	3	3	3	974
http://www.you.gr/	20	div	11	11	11	32	31	34	1	1	1	6	6	6	258
http://www.fk-shop.es/	96	div	7	7	7	38	36	39	3	2	4	7	6	7	108
http://la-shop.es/	9	div	8	8	8	19	19	19	1	1	1	4	4	4	299

- Column 3: The name of the tag which represents a product record on the page.
- Column 4–6: The most frequent number of parent elements of the HTML elements which represent the product records. That means, most of the product record elements have this number of parents. The maximum number of parent elements shows the maximum number of parents one or more of the product record elements have. The minimum number of parent elements correspond to the minimum number of parent elements one or more product record elements have.
- Column 7–9: The most frequent number of children elements of the HTML elements which represent the product records. That means, most of the product record elements have this number of children. The maximum number of children elements shows the maximum number of children one or more of the product record elements have. The minimum number of parent elements correspond to the minimum number of parent elements one or more product record elements have.
- Column 10–12: The most frequent number of image elements (<*img*> tag) included in the HTML elements which represent the product records. That means, most of the product record elements include this number of images. The maximum number of image elements shows the maximum number of images one or more of the product record elements contain. The minimum number of images elements correspond to the minimum number of image elements one or more product record elements include.
- Column 13–15: The most frequent number of anchor elements (<*a*> tag) included in the HTML elements which represent the product records. That means, most of the product record elements include this number of anchors. The maximum number of anchor elements shows the maximum number of anchors one or more of the product record elements contain. The minimum number of anchor elements correspond to the minimum number of anchor elements one or more product record elements include.
- Column 16: The average length of included text shows the average length of text included in a product record built over all product record elements on the product overview page.

The analysis of the collected product record data led to the following results:

- There are seven different tag types including the product records in the selected page set for the analysis. With 63.3 % the most product records are represented by a <*div*> tag and still 20 % are included in a <*li*> tag. Additionally, the product records were represented once by a <*tr*> tag, a <*table*> tag, a <*form*> tag, a <*section*> tag and a <*ul*> tag. Since the tag type of the product records vary in almost 40 % of the selected pages there cannot be made an assumption about the type of tag including the product records.
- Considering the parent elements of the product record elements of the selected pages the number and path of parent elements was exactly the same for all product records inside a page for 100 % of the analysed pages. That leads to

the conclusion that all product records of one page are located in the same data region. Thus, if the parent path of one or more product records could be identified the remaining product records can be obtained based on the path built from parent elements.
- 36.7 % of the product records of the selected pages include the same children whereas even 63.3 % contain a different number of child elements. The number of included child elements in the considered page set comprises a range from 6 to 158 children. For this reason the product records cannot be identified based on analysing their child element structure.
- 66.7 % of the product records of the considered pages include exactly one image element. 80 % of the product records include the same number of image elements as the other product record elements of the page, whereas this number differs for 20 %. The range of the number of image elements varies from 1 to 20 elements. On this account one can only assume that a product record will usually include at least one image element, but no assumption can be made about the number of image elements.
- 73,3 % of the product records include the same number of anchor elements as the other product record elements of the page, whereas it differs for 26,7 %. The range of the number of anchor elements varies from 1 to 15 elements. Therefore it can be only assumed that a product record will usually include at least one anchor element, but it cannot be made an assumption about the number of anchor elements.
- Each product record element of the selected pages contains text. The average length of the text included in one product record element comprises a range from 33 to 1,740 characters. Thus, no assumption of the text length inside a product record element can be made, but it can be expected that a product record contains some text.

4 Approach

The proposed approach, called LightExtraction, is a fast and pragmatic method for automatically detecting and extracting product records from e-shop websites.

The existing approaches presented in Sect. 2 need many steps to identify and extract relevant data records from Web pages. The MDR algorithm is a slim approach, but for the identification of the relevant datasets an additional method is needed. In contrast, LightExtraction automatically detects and extracts the product records of an e-shop Web page through only a few steps.

The functionality of the developed algorithm is based on the results of the analysis of the product records presented in Sect. 3. The LightExtraction algorithm is shown in the form of pseudo code in Fig. 1. LightExtraction uses a clustering technique based on a special tag path representation of the elements in the HTML page tree of a Web page for identifying and extracting product records.

The input for the algorithm is the URI of a Web page, which is retrieved and rendered in the first step e.g. by using Selenium WebDriver[8]. All information like

[8] http://docs.seleniumhq.org/projects/webdriver/.

```
render Web page
(add external styles)

for each element in HTML page tree:
  ignore style & script elements
  if product record filter matches:
    generate tag path
    add element to tag path cluster

 get tag path of cluster with max. elements

results = elements with identified tag path
results += elements with same parent path

return results
```

Fig. 1. LightExtraction algorithm.

CSS or information created by JavaScript code is made available in the HTML page tree. Since the majority of e-shop websites are automatically generated by modern e-shop software which uses templates for viewing the content of a database the product records are usually located in the same data region of the e-shop website and contain similar or even the same elements. For this reason the page elements are filtered by analysing their basic structure as described in Sect. 4.1 and then the elements are clustered based on a special tag path representation as depicted in Sect. 4.2. The elements are extracted by the created tag paths as described in Sect. 4.3. The output of the algorithm are the HTML elements of the Web page including the product records.

The first version of our approach was designed as we presented in [12]. Although the first version depicted in [12] returned good results we made some improvements within the element filtering as well as the element clustering parts of the approach in order to obtain even better results. In this article the whole approach including all improvements since the last published version of LightExtraction is described.

In [13] we describe the subsequent step of product attribute extraction which follows the product record extraction step done by LightExtraction. As for the attribute extraction additional information like data from external style sheet files is required some further pre-processing of the input data is needed after the page rendering process of LightExtraction, e.g. the embedding of the style sheet information within the html elements of the page tree. Those style sheet information is used for identifying and extracting the product attributes like product name or product price. The step of product attribute extraction follows the step of product record identification and extraction and is based on the results of LightExtraction. The product attribute identification and extraction is not subject of this article. We proposed an approach for that subject in [13].

4.1 Element Filtering

After the rendering process LightExtraction runs through all elements inside the HTML page tree. The algorithm rejects all elements having only a styling purpose like <*b*>, <*strong*> or <*em*> or elements including JavaScript code like <*script*>. In the next step LightExtraction checks if the element probably is a product record by using a special filter. The filter compares the structure of the element to a basic element structure, which is expected for a product record. The filter of LightExtraction assumes that a product record (1) contains at least five child nodes and additionally, that it (2) includes some text (product name and description) and (3) an image tag (<*img*>, product image) as well as (4) an anker tag (<*a*>, hyperlink to product detail page). In this way LightExtraction prevents the detection of single record items or items of large navigation menus as product records. The filter was implemented as a simple if-statement which checks the elements for having the mentioned structure.

For websites including menus with menu items which also contain elements meeting the filter criteria mentioned above further filter criteria are needed. Thus, the current version of LightExtraction add two more criteria for the filtering step: (1) the text inside the element or one of its child elements must include a price as well as (2) a currency. We have defined a regular expression for identifying prices within the text of an element and another regular expression for finding currencies.

1. Regular expression for identifying prices: [0-9]+[,|.]?[0-9]0,3
2. Regular expression for identifying currencies: (\$|&#36;|&#x24;|Dollar|USD|£|&pound;|&#163;|&#xA3;|&#8364;|Pound|POUND|GBP|€|&euro;|&#128;|&#x80;|EUR|Euro|EURO)+

Currently, the regular expression for currencies finds only the following currencies: (1) British Pounds (GBP), (2) Euros (EUR) and (3) United States Dollars (USD). In order to be able to find further currencies the regular expression would need to be extended.

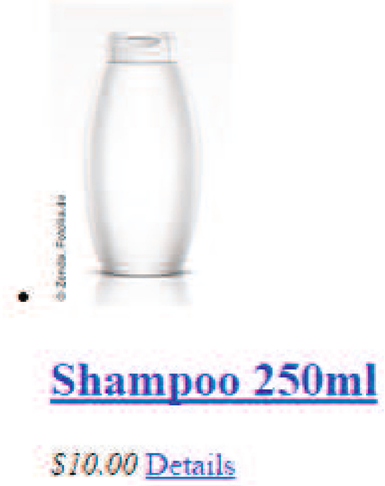
4.2 Element Clustering

For the element clustering a special tag path is built for each element. The first part of the tag path describes the path from root element (< *html* >) of the Web page to the actual element (including the actual element). An asterisk in square brackets is added after the actual element for marking it. The second part of the tag path consists of the tag paths of all child elements from the child element to its last element. That means the tag paths of all child elements are connected together to one long tag path. In order to be able to distinguish different elements which would have the same paths (e.g. the *tr* and *td* elements of the same table) as well as to store the information which part is the parent path an asterisk in square brackets is used to highlight the actual element.

Figure 2(a) shows the HTML snippet of an example product record. The Web browser view of this product is presented in Fig. 2(b) and its tag paths created

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de" lang="de">
 ...
    <body>
        <div>
            <ul class="product-list" id="product-list">
                ...
                <li class="item">
                    <a href="http://www.shop.com/shampoo250.html">
                        <img src="img1.jpg" alt="Shampoo 250ml">
                    </a>
                    <div class="col">
                        <h2>
                            <a href="http://www.shop.com/shampoo250.html">Shampoo 250ml</a>
                        </h2>
                        <em class="discount_price">
                            <span class="price">&#36;10.00</span>
                        </em>
                        <a href="http://www.shop.com/shampoo250.html">Details</a>
                    </div>
                </li>
                 ...
            </ul>
        </div>
    </body>
</html>
```

(a) HTML snippet.



Shampoo 250ml

$10.00 Details

(b) HTML snippet.

```
...
/html/body/div/ul/li[*]/a/img/div/h2/a/span/a
/html/body/div/ul/li/a[*]/img
/html/body/div/ul/li/a/img[*]
/html/body/div/ul/li/div[*]/h2/a/span/a
/html/body/div/ul/li/div/h2[*]/a
/html/body/div/ul/li/div/h2/a[*]
/html/body/div/ul/li/div/span[*]
/html/body/div/ul/li/div/a[*]
...
```

(c) Tag path snippet.

Fig. 2. HTML snippet, Web browser view &tag path snippet of a product record.

by LightExtraction are shown in Fig. 2(c). The first line of Fig. 2(c) represents the tag path of the searched list element including the product record data. The tag path until the asterisk in square brackets represents the tag path from the root element to the actual element, the path after this marking shows the element paths of the actual element's children which were compound to one long path string. The tag path built by LightExtraction does not properly represent the HTML tree of the element since the tag paths of the child elements are put together to a single long tag path string not respecting the structure of the children in the HTML tree. But the structure of the element's children is not important for the clustering of the HTML elements since the goal is to cluster elements with the same tag name, having the same path from root and containing the same child elements.

The elements are clustered based on the created tag path. That means each element cluster comprises elements having exactly the same tag path. Thus, an element cluster contains only the same element types (e.g. $< div >$), which are probably elements of the same data region of the Web page and which include the same child elements.

During the usage of our first version of LightExtraction we described above and in [12] we gained some experience and as we could learn the product offers within in a website and even within a Web page are structured less homogeneous than initially expected. The structure of the product offers within a website or even a Web page differ in the number of elements they include as some of them for example include a rating and some do not, some offers list a regular as well as an offer price whereas others list only a regular price. As the product offers are structured so different clustering them by considering the child elements they include does not yield in very good results. Hence, we have changed the tag path generation mechanism of LightExtraction as shown in the tag path snippet presented in Fig. 3.

The tag path shown in Fig. 3 is created by LightExtraction for clustering the elements of the same data region of a Web page. Thus, LightExtraction generates a tag path for each element of the HTML page tree which represents the path of the element within the HTML page tree from the root element (*html*) to itself including the tag name of the element described by the tag path. That means the clustering mechanism strings the tag names of the element's parent elements together separated by a slash whereby the last element name is the name of the element described by the tag path. The elements of the Web page are clustered by the tag path. Therefore elements having the same tag path are within the same cluster. Following this approach all elements located within the same data region of the Web page will be clustered together.

```
...
/html/body/div/ul/li
/html/body/div/ul/li/a
/html/body/div/ul/li/a/img
/html/body/div/ul/li/div
/html/body/div/ul/li/div/h2
/html/body/div/ul/li/div/h2/a
/html/body/div/ul/li/div/span
/html/body/div/ul/li/div/a
...
```

Fig. 3. Tag path snippet of the novel tag path structure.

4.3 Product Record Extracting

LightExtraction assumes that the cluster containing the maximum number of elements includes the majority of the product records. Thus, LightExtration takes the tag path of that cluster for identifying all clusters containing data records by searching for all elements in all clusters having the same element tag path. The elements with the same element tag path are the elements of the same data region as the elements of the cluster including the maximum number of elements. LightExtraction considers the result set of that last step as the product records of the Web page and extracts them.

In the case that several clusters have an identical number of elements the cluster with the longest tag path is considered as the cluster including the data records. In such cases LightExtractor assumes that the product offers are embedded into several HTML elements. Taking the longest tag path LightExtractor expects to extract the element which is the direct parent of all product attributes elements (e.g. product image, product name, product description).

5 Experiment

For the experiment we implemented the LightExtraction approach in Python[9]. The rendering of the HTML page tree is done by Firefox Selenium Driver for Python[10]. For the navigation in the HTML page tree we use Beautiful Soup[11].

For MDR we use the MDR implementation available on the MDR Website[12] of the Department of Computer Science of the University of Illinois at Chicago (UIC).

Since the ClustVX Demonstration Website[13] is not available we have made our own implementation in Python. For the rendering of the HTML page tree and the navigation in the HTML tree we use Firefox Selenium Driver for Python and Beautiful Soup.

According to [14] in 2014 the five product categories most often bought online in Europe were clothes, books, home electronics, cosmetics and CDs. We have created an experimental dataset containing randomly selected Web pages. The dataset has to be a mixture of Web pages of at least three different countries and each Web page must include one of the most popular product categories. The resulting experimental dataset is shown in Table 2.

5.1 Experimental Setup

For the experiment we implemented the LightExtraction approach in Python[14]. The rendering of the HTML page tree is done by Firefox Selenium Driver for Python[15]. For the navigation in the HTML page tree we use Beautiful Soup[16].

[9] https://www.python.org/.

[10] http://selenium-python.readthedocs.org/en/latest/api.html.

[11] http://www.crummy.com/software/BeautifulSoup/.

[12] http://www.cs.uic.edu/~liub/WebDataExtraction/MDR-download.html.

[13] http://clustvx.no-ip.org/.

[14] https://www.python.org/.

[15] http://selenium-python.readthedocs.org/en/latest/api.html.

[16] http://www.crummy.com/software/BeautifulSoup/.

For MDR we use the MDR implementation available on the MDR Website[17] of the Department of Computer Science of the University of Illinois at Chicago (UIC).

Since the ClustVX Demonstration Website[18] is not available we have made our own implementation in Python. For the rendering of the HTML page tree and the navigation in the HTML tree we use Firefox Selenium Driver for Python and Beautiful Soup.

According to [14] in 2014 the five product categories most often bought online in Europe were clothes, books, home electronics, cosmetics and CDs. We have created an experimental dataset containing randomly selected Web pages. The dataset has to be a mixture of Web pages of at least three different countries and each Web page must include one of the most popular product categories. The resulting dataset is presented in Table 2.

Table 2. Experimental dataset.

No.	URI for data extraction
1	http://www.barnesandnoble.com/u/new-books-fiction-nonfiction-bestsellers/379004022
2	http://www.ebay.com/chp/Baitcasting-Reels-/108153
3	http://www.terrashop.de/
4	http://www.very.co.uk/home-garden/curtains-blinds/made-to-measure curtains-blinds/e/b/116982.end
5	http://www.electricshop.com/televisions/televisions/icat/subtelevisions/iflt/tag-screentype%7C46_4kultrahd_2755
6	http://www.alconeco.com/makeup/eyes
7	https://thecomicbookshop.comicretailer.com/comics-sale
8	http://coozina.gr/store/home.php?cat=188
9	http://atlasstoked.com/
10	http://www.bestarabic.com/mall/ar/
11	http://coozina.gr/store/home.php?cat=20

As MDR can identify and extract data regions as well as the included data records but it is not able to decide which is the data region containing the relevant data records, we manually identify the relevant data region (if extracted) and count the correctly and incorrectly extracted data records for the evaluation.

Since the set up of the first version of the LightExtraction algorithm it has been improved as described in Sect. 4. Therefore we compared the first version of LightExtraction by extracting product records of the Web pages shown in Table 2

[17] http://www.cs.uic.edu/~liub/WebDataExtraction/MDR-download.html.
[18] http://clustvx.no-ip.org/.

to MDR and ClustVX. After the development of the improved version of LightExtractor (about 6 month after the first version) we tested the novel version by extracting data records from the same experimental dataset of Web pages as in the first round of the experiment. In the first experimental round we compared the first version of LightExtraction to MDR and ClustVX, whereas in the second round we compared the improved version to the first version of LightExtraction as well as to the best other approach of the first round which has been ClustVX.

5.2 Evaluation Metrics

For the evaluation of the results and the comparison of the different approaches we use the precision and recall measures, which are common metrics in the field of information retrieval. The definition of precision and recall in the context of information retrieval is given in Eqs. 1 and 2 [15].

$$Precision = \frac{|Relevant\ Records\ \cap\ Retrieved\ Records|}{|Retrieved\ Records|} \tag{1}$$

$$Recall = \frac{|Relevant\ Records\ \cap\ Retrieved\ Records|}{|Relevant\ Records|} \tag{2}$$

Since LightExtraction and the other approaches have to classify the elements of the Web pages into relevant data records (product record) and other elements (irrelevant data) we can use the terms True Positives, False Positives, True Negatives and False Negatives for calculating precision and recall.

A True Positive (TP) is a correct hit, which is a correctly extracted data record (in our case: a product record). False Positives (FP) are incorrect hits or false alarms, which are incorrectly extracted data records. True Negative (TN) means a correct rejection, which is a correctly rejected data record. In our context True Negatives cannot be measured since the number of all negative data records on website is unknown. Incorrect rejections or missing hits are defined as False Negatives (FN), which are incorrectly rejected data records (product records, which have not been detected).

Expressing the equations using the terms True Positive (TP), False Positive (FP) and False Negative (FN) Eq. 1 leads to Eq. 3 and Eq. 2 to Eq. 4.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{3}$$

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives} \tag{4}$$

In order to create the best conditions for being able to compare both experimental rounds we certainly used the same evaluation metrics.

Table 3. Experimental Results I. See Table 2 for the URIs of the websites.

No.	**Total**	**MDR algorithm**					**ClustVX**					**LightExtraction**				
		TP	**FP**	**FN**	**Precision**	**Recall**	**TP**	**FP**	**FN**	**Precision**	**Recall**	**TP**	**FP**	**FN**	**Precision**	**Recall**
1	43	0	0	43	0.00%	0.00%	30	0	13	100.00%	69.77%	30	0	13	100.00%	69.77%
2	25	0	0	25	0.00%	0.00%	25	0	0	100.00%	100.00%	25	0	0	100.00%	100.00%
3	12	8	24	4	25.00%	66.67%	12	0	0	100.00%	100.00%	12	3	0	80.00%	100.00%
4	12	0	0	12	0.00%	0.00%	12	0	0	100.00%	100.00%	12	0	0	100.00%	100.00%
5	24	24	0	0	100.00%	100.00%	24	0	0	100.00%	100.00%	0	0	24	0.00%	0.00%
6	45	0	0	45	0.00%	0.00%	45	0	0	100.00%	100.00%	45	0	0	100.00%	100.00%
7	59	0	0	59	0.00%	0.00%	10	0	49	100.00%	16.95%	59	0	0	100.00%	100.00%
8	36	0	0	36	0.00%	0.00%	36	0	0	100.00%	100.00%	36	0	0	100.00%	100.00%
9	12	3	29	9	9.38%	25.00%	3	0	9	100.00%	25.00%	12	0	0	100.00%	100.00%
10	24	0	0	24	0.00%	0.00%	0	2	24	0.00%	0.00%	20	1	4	95.24%	83.33%
Total:	*292*	*35*	*53*	*257*	*39.77%*	*11.99%*	*197*	*2*	*95*	*98.99%*	*67.47%*	*251*	*4*	*41*	*98.43%*	*85.96%*

5.3 Experimental Results

Table 3 shows the experimental results of the first round where the first version of LightExtraction is compared to MDR and ClustVX.

The number of product records available on each page of the experiment is given in the column "total". MDR obtains a precision of 39.77 % and a recall of 11.99 %, ClustVX reaches a precision of 98.99 % and a recall of 67.47 %, while LightExtraction achieves a precision of 98.43 % and a recall of 85.96 %. The results show that both LightExtraction and ClustVX achieve much better results than MDR. LightExtraction obtains a similarly good precision as ClustVX and even a better recall.

The reason for the missing product records (False Nagatives) of row 1 and row 10 is that these are located in a second product data region which LightExtractor is not recognizing. The product records of row 5 were not identified since LightExtractor has detected some promotion product records in the website menu contains 47 records which is a higher number of elements than the number of the "real" product records which is 24. The False Positives of row 3 and 10 appear since the elements are located in the same data region as the product records.

Table 4 presents the results of the second round of the experiment where the improved version of LightExtraction is compared to its first version as well as the best other approach from the first round which has been ClustVX.

Considering the results of Table 4 it is noticeable that the results of ClustVX as well as those of the first version of LightExtraction have deteriorated. The reason for achieving those worse results is the change of the products within the Web pages of the experimental data set. In the second round of the experiment some Web pages include product offers which differ in the structure of the HTML element quite more than in the first round. For those Web pages the results became worse than for the pages including product offers with a homogeneous element structure.

Table 4. Experimental Results II. See Table 2 for the URIs of the websites.

No.	Total	ClustVX					LightExtraction (Version 1)					LightExtraction (Version 2)				
		TP	FP	FN	Precision	Recall	TP	FP	FN	Precision	Recall	TP	FP	FN	Precision	Recall
1	20	0	7	20	0.00%	0.00%	20	0	0	100.00%	100.00%	18	7	2	72.00%	90.00%
2	50	50	0	0	100.00%	100.00%	50	0	0	100.00%	100.00%	50	1	0	98.04%	100.00%
3	15	0	3	15	0.00%	0.00%	15	0	0	100.00%	100.00%	15	0	0	100.00%	100.00%
4	12	12	0	0	100.00%	100.00%	0	9	12	0.00%	0.00%	12	0	0	100.00%	100.00%
5	24	24	0	0	100.00%	100.00%	0	48	24	0.00%	0.00%	24	1	0	96.00%	100.00%
6	45	45	0	0	100.00%	100.00%	45	38	0	54.22%	100.00%	45	40	0	52.94%	100.00%
7	45	0	7	45	0.00%	0.00%	45	0	0	100.00%	100.00%	45	6	0	88.24%	100.00%
8	24	0	1	24	0.00%	0.00%	0	24	24	0.00%	0.00%	0	13	24	0.00%	0.00%
9	15	3	0	12	100.00%	20.00%	15	0	0	100.00%	100.00%	15	0	0	100.00%	100.00%
10	24	0	2	24	0.00%	0.00%	20	0	4	100.00%	83.33%	24	4	0	85.71%	100.00%
Total:	*274*	*134*	*20*	*140*	*87.01%*	*48.91%*	*210*	*119*	*64*	*63.83%*	*76.64%*	*248*	*72*	*11*	*77.50%*	*90.51%*
11	47	40	0	7	100.00%	85.11%	40	0	7	100.00%	85.11%	44	1	3	97.78%	93.62%
Total:	*321*	*174*	*20*	*147*	*89.69%*	*54.21%*	*250*	*119*	*71*	*67.75%*	*77.88%*	*292*	*73*	*14*	*80.00%*	*90.97%*

In the case of the Web page coozina.gr (row no. 8 of the result tables) there are two different kinds of product offers which have also a diverse element structure. Coozina.gr includes special product offers (special price offers) as well as common product offers. In the case of the common product offers the single offers can be separated properly by its HTML elements, but in the case of the special product offers the product description and the product prices of a product offer cannot be extracted continuously. The reason for this issue is the structure of the product view where four product offers together build a row. The description of each four product offers are embedded into the row of a table whereas the prices of the four product offers are included into the following row of the table. Thus, the product descriptions and the corresponding prices cannot be extracted together by a general approach. Since the experimental dataset includes a Web page of coozina.gr containing only special product offers we also added a Web page of coozina.gr including common product offers (see row 11 of the result tables) to demonstrate that LightExtraction is able to properly extract product records from such Web pages.

The results of Table 4 show that we could improve the LightExtraction algorithm in its precision as well as in its recall. The precision of the improved version of LightExtraction is still not as good as the precision of ClustVX as it returns more False Positives. However, we consider the recall as the more important value as LightExtraction is able to find quite more of the product records within a Web page than ClustVX. Additionally, the identification of False Positives in the data record extraction step is not a real difficulty as the False Positives can be filtered out in a following processing step. We describe such a further step in [13].

6 Conclusions

This paper proposes a novel approach called LightExtraction for the automated identification and extraction of product offer records from e-shop Web pages. LightExtraction uses a filtering technique in combination with a tag path clustering method for identifying the product offers within a Web page in order to extract them. The paper presents the creation of the first version of LightExtraction as well as improvements in a second version of LightExtraction.

In an experiment the first version of LightExtraction is compared to the existing approaches MDR and ClustVX. The results of the experiment show that LightExtraction achieves much better results than MDR and a better recall than ClustVX, whereas LightExtraction needs significantly less process steps than ClustVX. In a second round of the experiment the improved version of LightExtraction is compared to its first version as well as to ClustVX. The results indicate that the quality of the results of ClustVX as well as the first version of LightExtraction strongly depend on the homogeneity of the structure of the product offers within the e-shop Web pages whereas the second version of LightExtraction depends less on that circumstance. The experiment demonstrated that the second version of LightExtraction achieves a much better recall than the other tested approaches and a satisfactory precision.

Acknowledgements. The work published in this article was partially funded by the SME E-COMPASS project of the European Union's Seventh Framework Programme for research, technological development and demonstration under the grant agreement no. 315637.

References

1. Nagelvoort, B., et al.: European B2C E-commerce report. Onlien (2014). http://www.adigital.org/sites/default/files/studies/european-b2c-ecommerce-report-2014.pdf
2. Simon, H., Fassnacht, M.: Preismanagement: Strategie - Analyse - Entscheidung - Umsetzung. Gabler Verlag, Wiesbaden (2008)
3. McGovern, C., Levesanos, A.: Optimizing pricing and promotions in a digital world: from product-led to customer-centric strategies (2014). http://www.accenture.com/us-en/Pages/insight-optimizing-pricing-promotions-digital-world-summary.aspx
4. Grigalis, T.: Towards web-scale structured web data extraction. In: Proceedings of the Sixth ACM International Conference on Web Search and Data Mining, WSDM 2013, pp. 753–758 (2013)
5. Grigalis, T., Cenys, A.: Unsupervised structured data extraction from template-generated web pages. J. Univ. Comput. Sci. **20**, 169–192 (2014)
6. Liu, B., Grossman, R., Zhai, Y.: Mining data records in web pages. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2003, pp. 601–606 (2003)
7. Zhao, H., et al.: Fully automatic wrapper generation for search engines. In: Proceedings of the 14th International Conference on World Wide Web, WWW 2005, pp. 66–75 (2005)
8. Walther, M., et al.: Locating and extracting product specifications from producer websites. In: Proceedings of the 12th International Conference on Enterprise Information Systems, ICEIS 2010, pp. 13–22 (2010)

9. Liu, B.: Web Data Mining. Exploring Hyperlinks, Contents, and Usage Data, pp. 1–14. Springer, Heidelberg (2006)
10. Andreson, N., Hong, J.: Visually extracting data records from the deep web. In: Proceedings of the 22nd International World Wide Web Conference, WWW 2013, pp. 1233–1238 (2013)
11. Real, R., Vargas, J.M.: The probabilistic basis of jaccard's index of similarity. Syst. Biol. **3**, 380–385 (1996)
12. Horch, A., Kett, H., Weisbecker, A.: A lightweight approach for extracting product records from the web. In: Proceedings of the 11th International Conference on Web Information Systems and Technologies, WEBIST 2015, pp. 420–430 (2015)
13. Peters, J.F.: Topology of Digital Images. Visual Pattern Discovery in Proximity Spaces. ISRL, vol. 63, pp. 1–76. Springer, Heidelberg (2014)
14. PostNord: E-Commerce in Europpe 2014 (2014). http://www.postnord.com/globalassets/global/english/document/publications/2014/e-commerce-in-europe-2014.pdf
15. Van Rijsbergen, C.J.: Information Retrieval. Butterworth-Heinemann, New York (1979)

A Comprehensive Analysis of the First Ten Editions of the WEBIST Conference

Giseli Rabello Lopes[1](✉), Bernardo Pereira Nunes[2,3],
Luiz André P. Paes Leme[4], Terhi Nurmikko-Fuller[5], and Marco A. Casanova[2]

[1] Computer Science Department, Federal University of Rio de Janeiro,
Rio de Janeiro, RJ, Brazil
giseli@dcc.ufrj.br

[2] Department of Informatics, Pontifical Catholic University of Rio de Janeiro,
Rio de Janeiro, RJ, Brazil
{bnunes,casanova}@inf.puc-rio.br

[3] Department of Applied Informatics, UNIRIO, Rio de Janeiro, RJ, Brazil

[4] Computer Science Institute, Fluminense Federal University, Niterói, RJ, Brazil
lapaesleme@ic.uff.br

[5] Oxford E-Research Centre, Oxford University, Oxford OX1 3QG, UK
terhi.nurmikko-fuller@oerc.ox.ac.uk

Abstract. An analysis of the proceedings of the first decade of the WEBIST conference, in terms of social networking and statistical analyses, as well as bibliometrics, unearthed information regarding existing patterns in the prevalent themes and topics of the conference, shedding light on the development of the event and its community as they grew and matured. In addition to the findings of this analysis we present a queriable Web-based application, which draws from a dataset of RDF triples, enabling the recreation of the examined patterns and the further exploration of the proceedings data.

Keywords: Conference analysis · Statistical analysis · Bibliometrics · Social network analysis · Webist analysis · Linked data

1 Introduction

Knowing about the past makes the present easier to understand, enables us to make predictions about the future, and helps guide us towards appropriate actions and correct decisions. However, with the perpetual flooding of newly available information, it became increasingly difficult to keep up to date, as well as to interpret and analyse data in meaningful ways. In response, there have been rapid technological advancements to support data analysts in both handling large amounts of data and in decision-making. In the commercial sector, companies and organisations relied on such analyses to overcome competitors, to improve customer relations and to identifying specific needs. In academia, data analysis has also been useful, helping to solve and uncover a number of

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 252–274, 2016.
DOI: 10.1007/978-3-319-30996-5_13

problems in domain as diverse as Health, Management, Marketing, Engineering and Computer Science [1].

Data analysis has been previously used to detect features such as related research groups, topics of interest, impact of authors and publications in a given field. Among others, an analysis of a group of four conferences in the Human-Computer Interaction (HCI) domain was conducted by Henry et al. [2]. Based on publication metadata (such as authors and keywords), it provided valuable insights into authors' behaviours and research topics investigated in HCI over the last two decades. Blanchard [3] presented a decade-long longitudinal study, which analysed the potential of cultural biases on the Intelligent Tutoring Systems (ITS) and Artificial Intelligence in Education (AIED) strands of the American Psychology Association (APA). Chen et al. [4] presented a visual analytic approach to identify co-citation clusters, classified and used to understand how astronomical research evolved between 1994 and 1998. Another example along the same lines was conducted by Gasparini et al. [5], who were able to identify central authors, institutions, important trends and topics in the HCI field. As for Information Systems (IS), Posada and Baranauskas [6] analysed a sister-event called International Conference on Enterprise Information Systems (ICEIS), and built a roadmap of the IS domain based on paper titles and authors from the last three years in ICEIS and the last eight years of selected papers published in a Springer series on IS. Chen et al. [7] performed a citation analysis of all papers published in the International Conference on Conceptual Modeling (ER) between 1979 and 2005. These analyses opened up a wide range of new research agendas and trends, as well as showing the value of a domain's introspective analysis.

Zervas et al. [8] presented a study on research collaboration patterns via co-authorship analysis in Technology-enhanced Learning fields. Similar analyses were conducted by Procopio Jr. et al. [9] for Databases fields and by Cheong and Corbitt [10] for IS (analysing the Pacific Asia Conference on IS). The analysis of co-authorships in research communities can reveal strong research groups in the area and also enable the creation of links between different groups.

We present an in-depth analysis of the first ten editions (2005–2014) of the WEBIST conference. So far, it attracted 2,867 researchers and professionals from several institutions, as well as published 1,449 papers, which in turn are being cited. The conference currently has five main tracks: *Internet Technology*, *Web Interfaces and Applications*, *Society, e-Business and e-Government*, *Web Intelligence* and *Mobile Information Systems.*

The analysis presented in this paper relies on techniques borrowed from social network analysis [11], bibliometrics and traditional statistical measures. In addition to presenting these analyses, we published the results in a format where they can be replicated and reused in further analysis. For this, we borrowed Batista and Loscio's approach [12] and used Linked Data (LD) principles. We also created a Web-based application that enables users to interactively explore data through a SPARQL endpoint.

In this paper, Sect. 2 overviews metrics and measures used in the analysis. Section 3 details the extraction, enrichment and publication process of raw WEBIST data into RDF data and presents a visualisation tool specifically created to manipulate and possibly assist users in finding new research groups, topics and insights. Section 4 presents several analysis conducted with the WEBIST tool. Finally, Sect. 5 concludes the work with remarks and future directions.

2 Background

This section provides the necessary background information required to understand the analysis conducted with the data. We review metrics and methods of statistical analysis, social network analysis and bibliometric indices.

2.1 Classical Statistical Measures

Standard deviation (σ) is a common measure of dispersion used to describe the central tendency of a distribution. Standard deviation [13] is defined as the square root of its variance, as shown in Eq. 1. Thus, considering a population X of N data points x_i, having average $\bar{X}$, σ is defined as:

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{X})^2}, \quad \text{where} \quad \bar{X} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{1}$$

Note that a low σ value indicates that the data points has a high central tendency, i.e., tend to be very close to the average, whereas a high σ value indicates that the data points are dispersed over a large range of values.

The *Pearson's correlation coefficient* [14], often denoted by the letter r, measures the strength and direction of the linear correlation between two variables X and Y. Pearson's coefficient (see Eq. 2) is defined as the covariance of the variables divided by the product of their standard deviations to measure their dependence:

$$r = \frac{\sum_{i=1}^{N}(x_i - \bar{X})(y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{X})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{Y})^2}} \tag{2}$$

An r value between $+1$ and -1 indicates the degree of linear dependence between X and Y: r=1 indicates a total positive correlation between the two variables; and r=-1 indicates a total negative (inverse) correlation. For instance, as X values increase, Y values linearly decrease.

The *Lorenz curve* [15] represents the cumulative distribution of a probability density function. Such a function is built as a ranking of the members of the population disposed in ascending order of the amount being studied. The percentage of individuals is plotted on the x-axis and the percentage of the variable values on the y-axis. The distribution is perfectly equalitarian when every individual has the same variable value; a 45-degree line represents the perfect equality. On the other hand, the perfectly unequal distribution is that in which only one individual has all the variable value, the curve is $y = 0$ for all $x < 100\,\%$, and $y = 100\,\%$ when $x = 100\,\%$, known as the perfect inequality line. This curve was initially created to study the social inequality of wealth and income distributions for a population, but it can be applied to analyse other distributions [16]. We used the Lorenz curve (Sect. 4) to study the distribution of papers by author.

The *Gini coefficient* [15] is a measure of statistical dispersion indicating the inequality among values of a frequency distribution. It is graphically represented as the area between the perfect equality line and the observed Lorenz curve.

The *Robin Hood index* [17], also called Hoover index, is used to measure the fraction of the total variable value that must be redistributed over the population to become a uniform distribution. It is graphically represented as the longest vertical distance between the Lorenz curve and the perfect equality line.

2.2 Social Network Analysis

Before introducing social network metrics and concepts [11,18–22], we recall that we may represent a social network as a graph $G = (N, E)$, where N is the set of nodes, where $n_i \in N$ represents an actor of the network, and E is the set of edges, where $e_i \in E$ represents a relational tie between a pair of actors.

The *Density* of a graph is defined as the number of the existing edges of the graph, divided by the maximum number of edges the graph can have. A density value equal to 1 indicates an entirely connected network, while 0 indicates a disconnected network. Considering an undirected graph, where the possible number of connections between each two nodes is 1, the density is defined as:

$$D = \frac{2|E|}{|N|\,(|N| - 1)} \tag{3}$$

where $|E|$ is the cardinality of the set of edges and $|N|$ is the cardinality of the set of nodes.

Modularity is a measure of the structure of networks and estimates the strength of division of a network into communities (groups). It is often used in optimisation methods for detecting community structure in networks. A high modularity value indicates a network having dense connections between the nodes within the communities, but sparse connections between nodes in different communities. Modularity is defined as [23]:

$$Q = \sum_i (e_{ii} - a_i^2) \tag{4}$$

where e_{ij} is the number of edges connecting nodes from the community i to nodes from the community j; $a_i = \sum_j e_{ij}$ is the number of edges with at least one node from the community i. Each edge contributes only once to the count (the contribution must be divided by half, one halve for e_{ij} and the other for e_{ji}).

A *Connected Component* of an undirected graph is a subgraph in which any two nodes are connected to each other by paths, and in which their nodes are not connected to any other nodes in the supergraph.

A *Giant Component* of a graph (also named *main component*) is the connected component which contains most of the nodes in the graph.

The *Giant Coefficient* of a graph is based on the size of the giant component G' of a graph G. It is defined as the number of nodes N' in the giant component divided by the total number of nodes N in the entire graph:

$$GC = \frac{|N'|}{|N|}, \quad \text{where} \quad N' \subseteq N \tag{5}$$

Diameter is associated with graph distance. It is defined as the maximum value among all shortest paths between two nodes of the graph (i.e., the longest distance between any pair of nodes belonging to the graph).

The *Average Clustering Coefficient* is a measure of the degree to which nodes in a graph tend to cluster together (connectivity of neighbours). It is defined as the average of the clustering coefficients of all the nodes in the graph:

$$\bar{C} = \frac{1}{|N|} \sum_{i=1}^{|N|} C_i \tag{6}$$

where C_i is the clustering coefficient of a node n_i and is calculated as the number of existing edges between the direct neighbours of n_i divided by the total number of possible edges directly connecting all neighbours of n_i.

2.3 Bibliometric Indices

This section introduces two common bibliometric indices often used to measure the impact, in terms of popularity, of researchers, scientific publications, conferences and journals.

The *h-index* was proposed to measure both the number of publications and the number of citations per publication of a scientist. According to Hirsch [24], a scientist has index h if h of his/her N_p papers have at least h citations each, and the other $(N_p - h)$ papers have no more than h citations each. This index is also applied to estimate the productivity and impact of conferences.

The *i10-index* indicates the number of publications of a scientist having at least ten citations[1].

3 WEBIST Workflow - from Raw to RDF Data

3.1 Overview of the Process

This section overviews the process of data acquisition, involving extraction, enrichment, preparation and consolidation, adopted to create the *WEBIST Dataset* and its use by the *WEBIST Analytics* tool. Figure 1 depicts the whole process.

[1] http://googlescholar.blogspot.com.br/2011/.

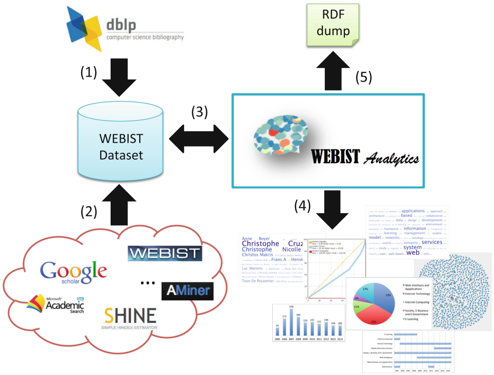


Fig. 1. WEBIST workflow.

Initially, we created an interlinked open dataset, called *WEBIST Dataset*, available in RDF, following the Linked Data principles [25], about the 10 editions of WEBIST conference. This dataset was created by aggregating data extracted from different data sources. The initial core of the data about WEBIST was extracted from DBLP (Digital Bibliography & Library Project)[2] (Step 1). Then, the data was enriched using data crawled from different Web sources such as Google Scholar Citations[3] (Step 2).

Based on the information loaded in the *WEBIST Dataset* (Step 3), the proposed Web application, called *WEBIST Analytics*, provides different functionalities such as exploratory search, and several analysis over the data, presented through different graphical visualisations (Step 4).

Moreover, using the *WEBIST Analytics* interface, the RDF dump of the *WEBIST Dataset* is available for download (Step 5). The *WEBIST Dataset* creation and *WEBIST Analytics* functionalities are detailed in the next subsections.

3.2 WEBIST Dataset

Data Acquisition. Over the last ten years, WEBIST conference data, such as paper acceptance or organisation committee, was published. Thus, to create a tool to seamlessly make sense of the data, we aggregated data extracted from different data sources, being aware of the possible necessity of initially submitting the data to deduplication [26] techniques.

The initial core of the data about WEBIST was extracted, in December 2014, from DBLP, a digital library about computer science publications. We were not able to find an updated source of DBLP data in RDF format (containing all editions of WEBIST conference). Thus, we had to extract the data directly from the XML version of DBLP available. This XML data also contained information

[2] http://www.informatik.uni-trier.de/~ley/db/.

[3] http://scholar.google.com/citations.

about the name disambiguation of the authors (different spellings of the name representing the same author in XML version of DBLP). Thus, the authors name disambiguation [27] was facilitated in this initial core. In summary, we collected information about the published papers and authors of WEBIST, reaching a total of 1,449 papers and 2,867 authors.

Data Enrichment. Data enrichment serves as a means to extending the initial data from additional data sources. For this, we developed a focused crawler to obtain this additional information. In this step, information from Google Scholar Citations and Google Scholar were used to obtain bibliometric indices of WEBIST authors. Specifically, the key of authors in Google Scholar Citations and the authors indices (*h*-index, i10-index and number of citations) were extracted from Google Scholar[4] and Google Scholar Citations, respectively. The crawling process used the name of the authors to perform the searches. Using this strategy, 748 authors profiles were found in Google Scholar Citations, representing 26.09 % of the total WEBIST authors. Other complementary information about some publications citations was crawled from Google Scholar. We collected the number of citations for the presumed most cited papers. The candidates to be most cited papers were obtained by the topmost ranked WEBIST papers presented in SHINE (Simple H-INdex Estimator)[5], Arnetminer[6] and Microsoft Academic Search[7]. Additional information about the main research areas and program committee (members and their affiliations) of each edition of WEBIST were extracted from each conference Web site[8] Moreover, other information about each conference edition, such as location, number of submissions, number of countries with submissions and paper acceptance rates (for full papers and oral presentations), were extracted from the forewords of the WEBIST proceedings available at SCITEPRESS digital library[9].

Data Transformation. Another crucial step is data transformation, carried out after data acquisition involving the preparation and enrichment steps, requiring a common format for the data. For this, we followed the Linked Data principles [25] that encourage data publishers to expose their data through HTTP mechanism and to use RDF as the data description language. According to these guidelines, the publishers should name things using HTTP URIs and provide appropriate clipping of data in RDF when users follow the URIs. All the data about WEBIST, obtained in the two previous steps, were first loaded in a relational database. After that, we used a relational-to-RDF framework (D2RQ) [28] that dynamically transforms relational data into RDF graphs. It provides an HTML browser for relational databases as well as a SPARQL interface to

[4] http://scholar.google.com.

[5] http://shine.icomp.ufam.edu.br.

[6] http://arnetminer.org/.

[7] http://academic.research.microsoft.com/.

[8] http://www.webist.org/ [*2005-2011:* WEBIST*$year$*; *2012-2014:*?y=*$year$*].

[9] http://www.scitepress.org.

query the database. This framework also provides a mapping language to define rules for transforming relational data and schema into RDF graphs.

Data Publication. The successful completion of these previous steps ensured that the dataset was available to others (both in terms of users and applications) that want to use it for different purposes. The RDF dump of the *WEBIST dataset* is available for download from the *WEBIST Analytics* interface.

3.3 WEBIST Analytics Application

WEBIST Analytics, a Web-based application, was created to provide multiple perspectives of the data produced by WEBIST conferences over the 10 editions. In addition to providing the *WEBIST dataset*, the proposed application is also composed of analytics tools, graphical visualisations and a simple search engine that assists users in finding, uncovering and making sense of the information available. *WEBIST Analytics* application can be accessed at: http://lab.ccead.puc-rio.br/webist_analytics/.

Based on the information loaded in the *WEBIST Dataset*, the proposed Web application provides different functionalities as both exploratory search and several analyses over the data, presented through different graphical visualisations. Free text search is available over two different WEBIST graphs, the co-authorships graph (among authors) and a more complete graph composed by co-authorships and authoring relations (among authors and publications). It allows users to search and retrieve related information about WEBIST conferences, including an interactive visualisation of networks. Other exploratory search is allowed via tag cloud visualisations. In this case, the terms in the tag cloud can be selected and the associated publications retrieved, which in turn assists users in finding papers related to each research topic.

4 Analysis and Results

This section presents and discusses the results of the analysis available in *WEBIST Analytics*. We observe that the results reported in this section were computed using the methods and metrics presented in Sect. 2.

4.1 WEBIST Overview

Table 1 overviews the last ten editions of the WEBIST conference with respect to the paper acceptance rate and the venue information. Since the first edition of the WEBIST conference, the full paper acceptance rate decreased and became stable under 15 % of all submitted papers. The low number of full papers accepted by WEBIST may suggest the level of rigorousness of the reviewers as well as the level of quality expected by the conference. On the other hand, the high acceptance rate for short papers (see oral presentations rates) may indicate an inclination of WEBIST towards bringing together researchers with work in progress and researchers with consolidated work, possibly offering opportunities for knowledge transfer and discussion.

In addition to the paper acceptance rate, Table 1 provides information about the location of each WEBIST edition. Note that although WEBIST is an international conference, with the exception of its first edition that took place in USA, all editions were held in Europe, mostly Spain and Portugal. As the number of submitted papers from all over the world has roughly remained the same, independently of where the conference took place (USA, Germany, Netherlands, Spain or Portugal), the change of place could bring extra benefits such as new collaborations with local universities and researchers.

Table 1. Conference stats.

Year	Location	#submitted papers	#countries with submissions	% of accepted papers	
				Full papers	Oral pres.[a]
2005	Miami, USA	110	37	22 %	49 %
2006	Setubal, Portugal	218	more than 40	16 %	50 %
2007	Barcelona, Spain	367	more than 50	14 %	44 %
2008	Funchal, Madeira, Portugal	238	more than 40	13 %	40 %
2009	Lisbon, Portugal	203	47	13 %	36 %
2010	Valencia, Spain	205	46	12 %	36 %
2011	Noordwijkerhout, Netherlands	156	43	9 %	33 %
2012	Porto, Portugal	184	41	13.6 %	44.6 %
2013	Aachen, Germany	143	43	19 %	39.9 %
2014	Barcelona, Spain	153	49	15.03 %	41.83 %

[a] Oral presentation including full papers and short papers.

4.2 General Analysis

An initial analysis of all WEBIST conferences was conducted with regard to its authors and publications. In this analysis we gathered 1,449 publications, which included all full papers, short papers, posters and selected papers. Figure 2 depicts the distribution of the papers over the conference editions. The number of accepted papers reached its peak in 2007, where 270 papers were accepted to a single conference, a figure almost twice the average number of papers accepted to other editions. This peak number of publications may be an indication of the rapid increase in the popularity of WEBIST and its reaching a certain level of maturity over the years, settling on a stable conference-size and community.

A rough analysis of the community can be carried out based on the number of authors of a scientific publication. The number of authors of a paper gives us a hint of the average size of the community and research groups. Across the 10 editions of WEBIST, there have been contributions from 2,867 authors, which gives an average of 2.91 authors per publication (with a standard deviation (σ) of 1.35, the maximum number of authors being 14 per paper and the minimum 1). Figure 3 shows the distribution of the average number of authors per year.

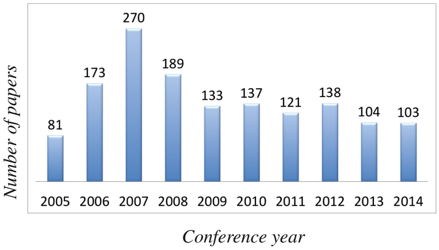


Fig. 2. Number of papers published per year.

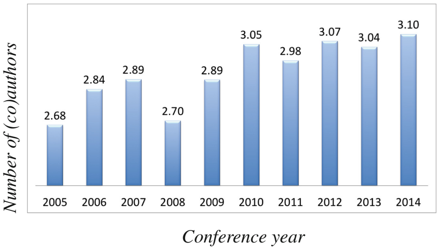


Fig. 3. Average number of (co)authors per paper over the conference years.

The list of topmost authors of WEBIST may reveal not only prolific authors, but possible experts and supporters for future editions of the conference. The engagement of researchers in a specific community could be initially measured by the number of papers they have had accepted in the earlier editions of the conference. The assumption is that, if they had over a specific number of papers, they might be eligible to make part of the program committee. After 10 editions, a total of 29 authors had more than 6 papers. The most active researcher had 15 published papers and the second had 12 papers. Figure 4 shows the top authors as a tag cloud[10]. The size of the names represents how active a research is in the WEBIST conference.

Figure 5 presents the Lorenz curve[11] along with an analysis based on the Gini coefficient and the Robin Hood Index (see Sect. 2). The Gini coefficient resulted in 25.99 % of inequality, while the Robin Hood Index was 23.06 %. The results show that the Lorenz Curve is closer to the equality than to the inequality line. This is an expected result for peer-reviewed conferences, where only high quality papers are accepted for publication. Although a few authors have more than 6 papers in WEBIST editions, the Lorenz Curve and the Robin Hood Index show that no redistribution is necessary, i.e., there is no bias in accepting papers from a research group or another, but simply merit. A high Robin Hood Index would indicate a possible need for further analysis in some publications.

4.3 Co-Authorships Network

Social Network Analysis (SNA) techniques were applied to the obtained information about the co-authorships in the WEBIST conference. The analysis was

[10] http://tagcrowd.com.

[11] http://www.peterrosenmai.com/lorenz-curve-graphing-tool-and-gini-coefficient-calculator.

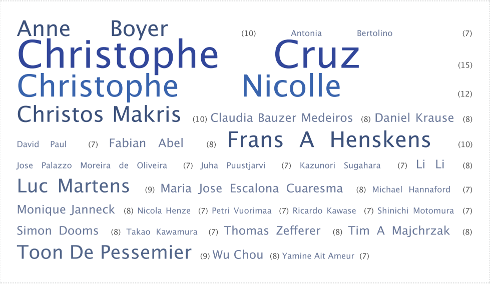


Fig. 4. Top authors with more than 6 papers.

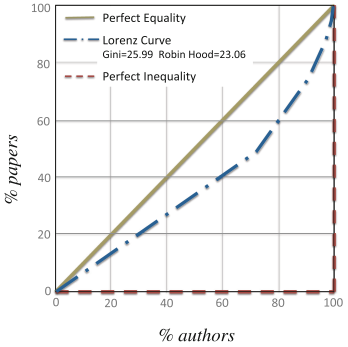


Fig. 5. Lorenz curve for the number of papers per author distribution.

conducted over an undirected graph G (defined in Sect. 2), where the nodes represent the authors and the edges represent a co-authorship between researchers. The WEBIST co-authorships network is comprised of 2,867 authors and 4,235 pairs of authors (edges) having at least one co-authored paper.

Table 2 shows an analysis of the co-authorship network using SNA measures. The analysis considers all WEBIST authors in the last 10 years. Briefly, we have:

– *Average Degree* shows that the authors, on the average, have co-authored papers with 2.9 other authors.
– *Density* shows a low proportion of co-authorships in the network relative to the total number possible (situation where all authors co-authored at least one paper with all others), only 0.1 %. It represents a weakly connected network. This shows an expected result in a conference network, where there are different groups of authors working in different papers. The measured modularity and the number of communities, as explained below, can reinforced this result.
– *Modularity* shows a high value representing the strength of division of the network into modules (also called groups, clusters or communities). Thus, WEBIST co-authorships network has co-authorships between the authors within the communities but none between authors in different communities.
– *Number of Communities* detected based on the modularity, was 803, being exactly the same as the **Number of Connected Components**. This shows that, in the analysed network, there are isolated communities that have not co-authorships in WEBIST with the authors of the other communities.

The following analysis takes into account only the giant component of the WEBIST network. Again, briefly, we have:

– *Giant Coefficient* represents the percentage of authors in the Giant Component of the WEBIST co-authorships network, being approximately 1.57 % (45 authors) of the total number of authors that published in all WEBIST conferences. These authors have 108 co-authorships between them (2.55 % of the total possible co-authorships, i.e., if each of these authors co-authored on at least one paper with all others).

- *Diameter* represents the longest of all the shortest paths between two authors in the Giant Component, being estimated as 8. This shows that the farthest authors in the Giant Component have more than six degrees of separation, based on co-authorship in WEBIST papers. This reveals that the Giant Component probably results from a hierarchical structure, which is natural when research groups of different institutions are involved. The different research groups (subgroups) are connected by "hub" authors (probably research group leaders or professors) that collaborate in different research projects amongst the subgroups, while some researches (probably students) developed more specific tasks (sometimes related to only one paper).
- *Clustering Coefficient* measures the average degree to which authors in the network tend to cluster together, being approximately 93.4 %. This shows that many authors belonging the Giant Component worked with other authors that also worked together in at least one paper.

Table 2. Social networks analysis from the WEBIST co-authorships network.

Measure	Value
Average Degree	2.954
Density	0.001
Modularity	0.995
Number of Communities	803
Number of Connected Components	803
Giant Coefficient[a]	0.0157
Diameter[a]	8
Average Clustering Coefficient[a]	0.934

[a] Estimated considering the Giant Component.

4.4 Authors Indices

In this section, we consider different bibliometric indices to analyse the profiles of WEBIST authors. As previously stated (Sect. 3), we identified and extracted Google Scholar Citations profiles for 26.09 % of the WEBIST authors. Thus, the analysis presented in this section is related only to this subset of the authors.

The bibliometric indices from WEBIST authors were firstly analysed in terms of the Average and the Standard Deviation (σ) (see results in Table 3). The bibliometric indices, obtained from Google Scholar Citations data, were separated into global indices, estimated considering all the years of the citations, and the same indices estimated considering only the citations since 2009. On the average, the authors presented a considerable total number of citations and i10-index values greater than their *h*-index. However, the Standard Deviation was quite

Table 3. Average and standard deviation of number of citations and bibliometric indices from authors.

Measure	Average	σ
overall citations	1,634.49	4,087.46
citations *since 2009*	988.95	2,565.17
overall h-index	14.30	12.17
h-index *since 2009*	11.54	8.98
overall i10-index	28.16	54.32
i10-index *since 2009*	19.94	42.03

high, showing that the community, as expected in good conferences, is formed of both young and senior researchers, as further discussed in what follows.

To better understand the profile of the WEBIST authors, we performed further analyses by splitting the authors into two groups, named A and B. We assigned to Group A those authors who had an *overall* h-index greater than the h-index *since 2009* and assigned to Group B those authors who had a *overall* h-index equal to the h-index *since 2009*. This classification assumes that the authors whose *overall* h-index consisted solely of citations made after 2009 were researchers who had started their careers more recently than those whose *overall* h-index included citations from before 2009.

Table 4 presents the results using this classification. This table shows, for each conference year, the percentage of authors and the respective average of the h-index per class. The results evidence that, in all conference editions, the number of authors in Group A is greater than those in Group B. Also, the results show that, in all conference editions, the average h-index of authors in Group A is greater. Note that the average of h-index is 18.35 for authors in Group A considering all editions of WEBIST conference.

4.5 Program Committees Analysis and Indices

Program committee (PC) members of the first ten editions of the WEBIST conference were examined for potential information regarding discernible patterns or possible emerging social networks around particularly interconnected nodes. We looked at 569 individual researchers from 49 distinct countries. Figure 6 illustrates the dispersion of these PCs across a world map - the darker the color, the higher the number of participating institutions (countries which appear white had none). The topmost countries were found to be Italy, United States (USA), Germany, United Kingdom, Greece and Spain, representative of the international but not necessarily global reach of the WEBIST network of participating researchers. For all these countries, the number of participations of researchers as PC members (in the analysed period, each researcher could have participated in a maximum of 10 editions) was greater than 100. Cross-referencing these findings with those from Sect. 4.1 (USA, Germany, Netherlands, Spain and Portugal),

Table 4. Percentage and average of h-index of scholars in groups A and B.

Year	Percentage		Average of h-index	
	group A	group B	group A	group B
2005	84.62 %	15.38 %	19.77	9.50
2006	78.65 %	21.35 %	18.03	8.84
2007	80.59 %	19.41 %	18.58	7.24
2008	63.73 %	36.27 %	18.38	6.95
2009	67.86 %	32.14 %	20.39	8.15
2010	67.03 %	32.97 %	18.64	7.00
2011	67.06 %	32.94 %	20.16	5.54
2012	57.02 %	42.98 %	15.65	5.39
2013	53.03 %	46.97 %	20.74	6.52
2014	55.56 %	44.44 %	19.72	4.90
All	65.64 %	34.36 %	18.35	6.58

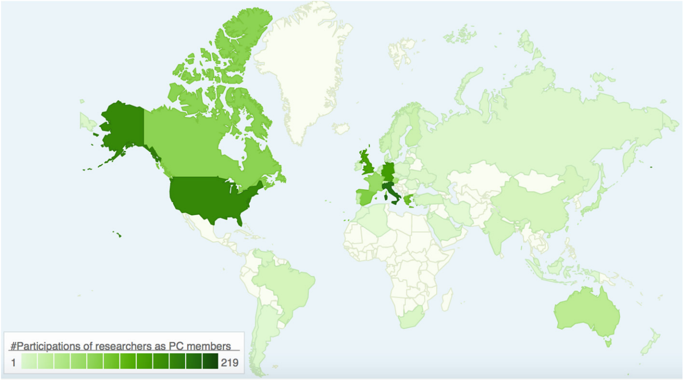


Fig. 6. Intensity of participations of PC members at institutions from the countries.

can provide some helpful suggestions in terms of potential future locations for conferences. These are in particular Italy (where WEBIST 2016 will be held), United Kingdom and Greece. Portugal was the most frequent location across previous conference sessions, but with 25 PC participants, it ranks as the 12th country overall.

The number of PC members is depicted in Table 5 (second column). On the average, the number of program committee members by conference year was approximately 175. To better illustrate the variation of researchers participating in the program committees, Fig. 7 shows a distribution based on the number of conference editions and how many researchers participated in that number of editions. Twelve researchers participated as a PC member in all of the ten editions of the WEBIST conference which form the dataset. Around a fifth, (20.21 %) of the researchers participated as PC members for at least 50 % of the considered editions (at least five editions). Figure 8 depicts all the most active PC members (there are 34) who participated in at least 80 % of the conference editions as a tag cloud. This tag cloud represents the names of the researchers followed by their number of participations in the WEBIST PC in parentheses.

Table 5 also shows the percentage of new PC members (third column). This category consists of researchers who have not attended WEBIST in the capacity of a PC member before the corresponding conference edition; the percentage of variation in each program committee, as compared to the edition that immediately preceeds it is shown in the fourth column. On the average, the program committees had 27 % new members and 34 % of each committee had not participated as a PC in the previous year. This analysis shows that the WEBIST program committees have been composed of experienced researchers (the "core" of the PC) but that it is also constantly renewed and refreshed with the addition of new members.

Table 5. Number of program committee members over the conference edition.

year	#PC members	%new PC members	%different PC members related to the previous year
2005	129	-	-
2006	116	28.45 %	28.45 %
2007	200	49.00 %	49.00 %
2008	185	11.35 %	11.89 %
2009	151	15.23 %	23.84 %
2010	158	31.01 %	42.41 %
2011	116	26.72 %	43.10 %
2012	245	50.20 %	63.27 %
2013	221	12.22 %	19.46 %
2014	184	19.02 %	26.09 %
Avg	175.11	27.02 %	34.17 %

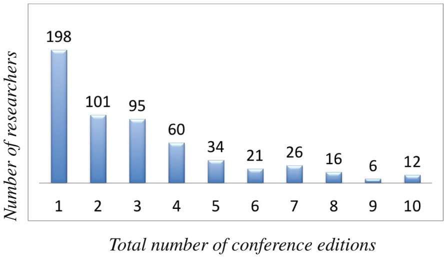
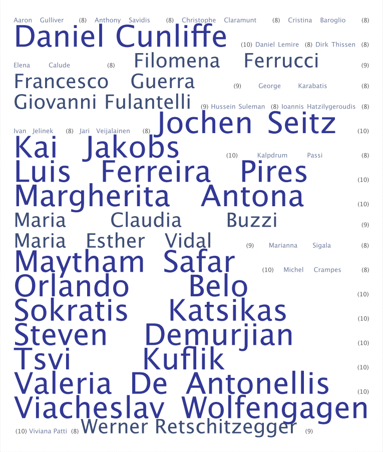


Fig. 7. Number of PC members participating in each total number of editions.

The following stage aimed to identify those PC members who also published in at least some WEBIST conference edition. In this analysis, the names of the authors (as extracted from DBLP) and the names of PC members (as extracted from WEBIST websites) were normalized (disregarding accents and not being case sensitive). A process of disambiguation was then carried out, by comparing the normalized versions of authors names to the normalized versions of PC members names. We were able to identify 114 equalities indicating that at least 20.03 % of the PC members are also authors in some WEBIST edition. Recall from Fig. 4 (Sect. 4.2) that 29 authors had published more than six papers in the first ten editions of WEBIST conference. Among them we identified 6 authors (20.69 % of the total) who were also PC members in some WEBIST edition (see Fig. 9). This reinforces our earlier conclusions regarding the most active authors. Moreover, none of the most active PC members (see Fig. 8) are amongst the topmost WEBIST authors (see Fig. 9) and all PC members published, on the average, only 2.31 papers in WEBIST. This reinforces the hypothesis of unbiased reviewing process (previously commented in Sect. 4.2) and one which is not favoring any group of authors, whether or not they are PC members.



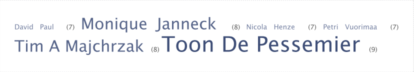


Fig. 8. Top researchers with more than 7 participations in program committees.

Fig. 9. Top PC members with more than 6 papers published in WEBIST.

We estimated the number of citations and bibliometric indices (h-index and i10-index) from the PC members that published papers in some WEBIST conference in terms of the Average (see results in Table 6). To facilitate a comparison, we replicated the values previously presented in Table 3 (these can be seen in the second column of Table 6). On the average, PC members showed a considerably higher total number of citations, i10-index, and h-index than that obtained for all WEBIST authors across all editions. These results are coherent since it is to be expected that the program committees are composed of a selected group of experienced and qualified researchers.

Table 6. Average of number of citations and bibliometric indices from PC members.

Measure	Avg from Authors	Avg from PC Members	%Increase
overall citations	1,634.49	2,787.28	70.53 %
citations *since 2009*	988.95	1,528.32	54.54 %
overall h-index	14.30	20.48	43.22 %
h-index *since 2009*	11.54	15.68	35.88 %
overall i10-index	28.16	47.08	67.19 %
i10-index *since 2009*	19.94	30.33	52.11 %

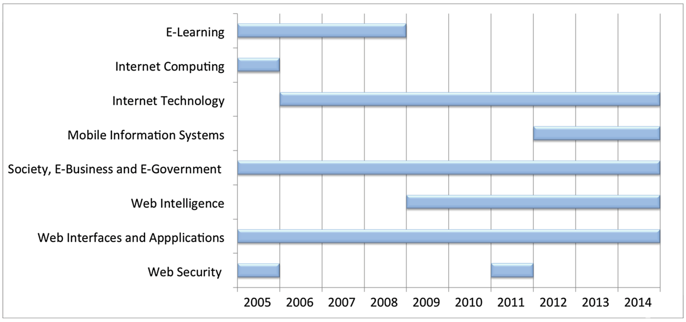


Fig. 10. Main conference areas per conference year.

4.6 Topics and Conference Areas

In this section, we analyse the topics of the papers published over the 10 years of WEBIST conference and their relation to the predefined main conference areas. Firstly, Fig. 10 presents, in alphabetical order, the main conference areas over the different conference editions. Some areas appear in all conference editions, such as *Society, E-Business and E-Government* and *Web Interfaces and Applications*. The third most frequent area is *Internet Technology*, which appeared from the second edition to the last one, probably as an expansion of *Internet Computing* (which appears only in the first conference edition). *Web Intelligence* and *Mobile Information Systems* appear more recently, in 2009 and 2012, respectively. *E-Learning* appears only in the first four editions of WEBIST conference. This phenomenon can be explained by the fact that the WEBIST conference, from 2009 to 2014, was held in conjunction with CSEDU (The International Conference on Computer Supported Education), a conference focused in innovative technology-based learning strategies and institutional policies on computer supported education (e-learning). *Web Security* appears only in specific editions (2005 and 2011).

Another analysis was performed over the topics covered by the papers published in WEBIST conferences. Figure 11 shows a tag cloud generated from the terms presented in the titles of the papers. This tag cloud represents the terms

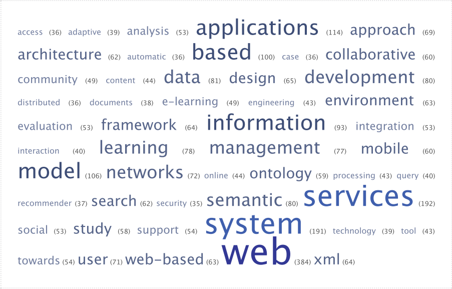


Fig. 11. Top 50 terms of years 2005–2014.

followed by their total frequencies in parentheses. Moreover, the term size in the graphic is proportional to its frequency. Terms such as *web*, *systems*, *services*, *applications*, *model* and *information* are the most frequent. These terms are aligned with the research focuses of WEBIST conference that are technological advances and business applications of web-based information systems. Briefly, we have:

For a more detailed analysis, we considered the evolution of main conference areas and terms presented in titles of WEBIST papers per conference year (tag clouds from top 50 terms of each conference year are available at *WEBIST Analytics*). Specifically, we verified what happened to the frequency of particular terms that are directly related to updates in the main conference areas.

– *e-Learning* area was eliminated in 2009. *E-learning* term was a frequent top term in titles between 2005 and 2008, but this was not true in the following years (2009–2014).
– *Web Intelligence* area was included in 2009. Terms related to topics such as information filtering and retrieval, Web mining and classification appeared in different conference years (including years prior to 2009).
– *Web Security* area appears in editions from 2005 to 2011. The *security* term appears in the tag cloud of 2005 but not in 2011. We decided to investigate the quantity of papers published in 2011 that were directly associated with this main research area and discovered that only two short papers and one poster were published. This was probably the underlying reason which led to the deletion of this main research area in the following year.
– *Mobile Information Systems* area was included in 2012. The *mobile* term appears among the top 50 terms in 2012 (previously the term already appeared in the first conference editions, but became prominent only after the inclusion of the *Mobile Information Systems* area in 2012).

We also studied the evolution of the top 50 terms in the titles over a decade of WEBIST conferences. Table 7 presents the average and the standard deviation (σ) of the frequency of the top 50 terms. In the first editions of the conference, with the exception of 2005, both the average and σ were high, leading us to conclude that there are likely to be terms that are related to major topics, as well as marginal topics in the accepted papers. In the most recent conference editions, the terms have a more equal distribution (greater equality frequency), showing that even whilst manifesting some peripheral change over the years, the conference found a core that is equally evolving. When analyzed in conjunction, the average and standard deviation demonstrate that the frequency of the top 50 terms (and consequently the relative frequency of the conference topics) is becoming more homogeneous. Moreover, a high diversity (dispersion) was observed, i.e., there were many terms (topics) covered by the conference over its 10 years.

The Pearson's correlation coefficient was estimated between the frequency of top 50 terms group from each conference edition (see results in Table 8). The sequence of the conference editions (underlined values in Table 8), except between 2006–2007, maintained a consistency within the group of top 50 terms: terms from one year correlated with the group of terms from the following year (Pearson's correlation coefficient is positive). Moreover, the correlation between the groups of top 50 terms from years 2008–2009 increased considerably compared with all the previous years (2005–2006; 2006–2007 and 2007–2008). This probably happened because, in this period, the main research areas were updated, with the removal of *E-learning* and the inclusion of *Web Intelligence.*

Finally, Table 8 shows an evolution on the research topics, considering the correlation between the top 50 terms of each conference edition and of all the others. The edition of 2010 presented, on the average, the highest Pearson's correlation coefficients between its top 50 terms and all others (being positive for all cases). Moreover, recall from Fig. 10 that WEBIST 2010 had as main research areas *Internet Technology*, *Society, E-Business and E-Government*,

Table 7. Average and standard deviation from frequency of top 50 terms per conference edition.

Year	Average	σ
2005	4.58	3.59
2006	9.08	6.90
2007	14.74	11.68
2008	10.50	9.12
2009	7.64	6.14
2010	7.18	5.37
2011	6.72	5.32
2012	7.12	4.53
2013	5.26	3.14
2014	5.08	3.02
All	70.30	55.66

Table 8. Pearson's correlation between the frequency of top 50 terms from each conference edition.

	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014
2005		0.211	0.219	−0.012	0.128	0.178	0.105	−0.004	0.082	0.080
2006			−0.035	0.390	0.241	0.205	0.059	0.253	0.294	0.170
2007				0.174	0.178	0.259	0.084	0.189	0.178	0.140
2008					0.341	0.289	0.088	−0.007	0.118	0.005
2009						0.036	0.206	0.250	0.203	0.013
2010							0.325	0.316	0.404	0.122
2011								0.175	0.245	−0.103
2012									0.135	0.106
2013										0.395
2014										

Web Intelligence and *Web Interfaces and Applications*, which are the only areas that occur in the majority of conference editions (the "core" of research areas).

4.7 Paper Citation Analysis

In this section, we performed an analysis related to the WEBIST topmost cited papers (recall for Sect. 3 how these topmost papers were obtained) and estimated the h-index for the WEBIST conference series. The h-index obtained was 18, indicating that there are at least 18 papers with at least 18 citations. Thus, Fig. 12 presents the percentage of top 18 most cited papers per type of publication. The results show that the most cited papers are mostly full papers (more than 50 %, corresponding to 10 papers).

Figure 13 presents the top 18 most cited papers based on the percentage per main research areas. It can be seen that the *Web Interfaces and Applications* and *Internet Technology* areas had the highest number of most cited papers in the top 18 (around 33 % each). Surprisingly, *E-Learning*, which appeared only in the first four editions of WEBIST, had a higher percentage (around 17 %) of the most cited papers than *Society, E-Business and E-Government* (around 6 %) which appeared in all conference editions. As expected, the most recent main research areas do not have papers in the top 18 (2010 was the latest year with a paper in the top 18).

5 Discussion and Outlook

We described the *WEBIST Dataset* and the *WEBIST Analytics* Web application. The former aggregates data from different sources and follows the Linked Data principles, while the latter provides different functionalities for the searching, analysing, and visualising the dataset.

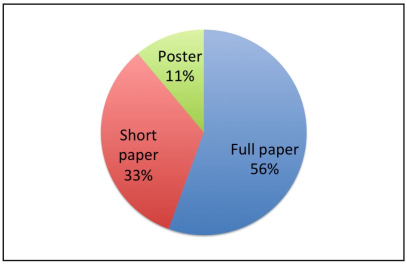


Fig. 12. Top 18 most cited papers per type of publication.

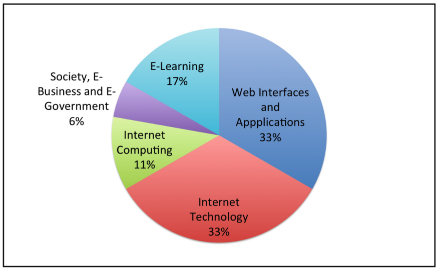


Fig. 13. Top 18 most cited papers per main research area.

A comprehensive analysis of the first ten editions of WEBIST illustrated the rapid growth in popularity achieved by WEBIST in 2007 and its maturation in subsequent years, reaching a stable conference-size, paper acceptance rate, community of IS experts, discernible research topics and supporters. The analysis highlighted the unbias of the reviewing process and how it contributed to the fast advancement of IS and the generation of knowledge: the WEBIST community plays a key role in knowledge transfer and impact in its domain (h-index = 18).

The *Web Interfaces and Applications* and *Internet Technology* tracks have been crucial to the development and popularity of WEBIST and they have accumulated the most cited papers. An important point to note is that the extinct *E-Learning* track, which appeared only four times as a main track, obtained a proportion of top cited papers which is higher than those of the *Society, E-Business and E-Government* track, although the latter appeared in all conference editions. Although the conference topics have became increasingly homogeneous, a higher diversity of topics and terms was observed. It is possible that a wider range of conference locations could bring about benefits, such as new collaborations with local universities and researchers.

The main contributions of this paper are the generated dataset and the Web application, which serve as a baseline for future analysis, including the extension of the proposed workflow to analyse multiple conferences and researchers from different fields.

Acknowledgements. This work was partly funded by CNPq under grants 444976/2014-0, 303332/2013-1, 442338/2014-7 and 248743/2013-9, by FAPERJ under grants E-26/101.382/2014 and E-26/201.337/2014 and by CAPES under grant 1410827.

References

1. Ott, R., Longnecker, M.: An Introduction to Statistical Methods and Data Analysis. Available 2010 Titles Enhanced Web Assign Series. Cengage Learning, Boston (2008)
2. Henry, N., Goodell, H., Elmqvist, N., Fekete, J.D.: 20 years of four HCI conferences: a visual exploration. Int. J. Hum. Comput. Interact. **23**, 239–285 (2007)

3. Blanchard, E.G.: On the weird nature of ITS/AIED conferences. In: Cerri, S.A., Clancey, W.J., Papadourakis, G., Panourgia, K. (eds.) ITS 2012. LNCS, vol. 7315, pp. 280–285. Springer, Heidelberg (2012)
4. Chen, C., Zhang, J., Vogeley, M.S.: Visual analysis of scientific discoveries and knowledge diffusion. In: Proceedings of 12th International Conference on Scientometrics and Informetrics (ISSI 2009), pp. 874–885 (2009)
5. Gasparini, I., Kimura, M.H., Pimenta, M.S.: Visualizando 15 anos de IHC. In: Proceedings of 12th Brazilian Symposium on Human Factors in Computing Systems, IHC 2013, SBC, pp. 238–247 (2013)
6. Posada, J.E.G., Baranauskas, M.C.C.: A study on the last 11 years of ICEIS conference - as revealed by its words. In: Proceedings of 16th International Conference on Enterprise Information Systems, vol. 3, pp. 100–111. SciTePress (2014)
7. Chen, C., Song, I.Y., Zhu, W.: Trends in conceptual modeling: citation analysis of the er conference papers (1975–2005). In: Proceedings of 11th International Conference on the International Society for Scientometrics and Informatrics, CSIC, pp. 189–200 (2007)
8. Zervas, P., Tsitmidelli, A., Sampson, D.G., Chen, N.S.: Kinshuk: studying research collaboration patterns via co-authorship analysis in the field of TeL: the case of educational technology & society journal. Educ. Technol. Soc. **17**(4), 1–16 (2014)
9. Procopio Jr., P.S., Laender, A.H.F., Moro, M.M.: Análise da rede de coautoria do Simpósio Brasileiro de Bancos de Dados. In: Brazilian Symposium on Databases - SBBD Posters (2011)
10. Cheong, F., Corbitt, B.J.: A social network analysis of the co-authorship network of the pacific asia conference on information systems from 1993 to 2008. In: PACIS, AISeL 23 (2009)
11. Wasserman, S., Faust, K.: Social Network Analysis: Methods and Applications. Cambridge University Press, Cambridge (1994)
12. Batista, M.G.R., Loscio, B.F.: OpenSBBD: Usando linked data para publicação de dados abertos sobre o SBBD. In: Brazilian Symposium on Databases - SBBD 2013, Short Papers (2013)
13. Bland, M.M., Altman, D.G.: Statistics notes: measurement error. BMJ **313**, 744 (1996)
14. Rodgers, J.L., Nicewander, A.W.: Thirteen ways to look at the correlation coefficient. Am. Stat. **42**, 59–66 (1988)
15. Gini, C.W.: Variability and mutability, contribution to the study of statistical distributions and relations. Studi Economico-Giuridici della R. Universita de Cagliari (1912)
16. Lopes, G.R., da Silva, R., Moro, M.M., de Oliveira, J.P.M.: Scientific collaboration in research networks: a quantification method by using gini coefficient. IJCSA **9**, 15–31 (2012)
17. Hoover, E.M.: Interstate redistribution of population, 1850–1940. J. Econ. Hist. **1**, 199–205 (1941)
18. Freeman, L.C.: Centrality in social networks: conceptual clarification. Soc. Netw. **1**, 215–239 (1979)
19. Hoser, B., Hotho, A., Jäschke, R., Schmitz, C., Stumme, G.: Semantic network analysis of ontologies. In: Sure, Y., Domingue, J. (eds.) ESWC 2006. LNCS, vol. 4011, pp. 514–529. Springer, Heidelberg (2006)
20. Marsden, P.V.: Egocentric and sociocentric measures of network centrality. Soc. Netw. **24**, 407–422 (2002)
21. Newman, M.E.J.: Scientific collaboration networks: I. network construction and fundamental results. Phys. Rev. E **64**, 016131 (2001)

22. Newman, M.E.J.: The structure and function of complex networks. SIAM Rev. **45**, 167–256 (2003)
23. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E **69**, 026113 (2004)
24. Hirsch, J.E.: An index to quantify an individual's scientific research output. In: Proceedings of National Academy of Sciences of the United States of America, vol. 102, pp. 16569–16572 (2005)
25. Berners-Lee, T.: Linked Data. In: Design Issues. W3C (2006)
26. Elmagarmid, A.K., Ipeirotis, P.G., Verykios, V.S.: Duplicate record detection: a survey. IEEE Trans. Knowl. Data Eng. **19**, 1–16 (2007)
27. Borges, E.N., de Carvalho, M.G., Galante, R., Gonçalves, M.A., Laender, A.H.F.: An unsupervised heuristic-based approach for bibliographic metadata deduplication. Inf. Process. Manage. **47**, 706–718 (2011)
28. Bizer, C., Seaborne, A.: D2RQ - treating Non-RDF databases as virtual RDF graphs. In: Proceedings of 3rd International Semantic Web Conference (2004)

Translation of Heterogeneous Databases into RDF, and Application to the Construction of a SKOS Taxonomical Reference

Franck Michel(✉), Loïc Djimenou, Catherine Faron-Zucker, and Johan Montagnat

University of Nice Sophia Antipolis, CNRS, I3S (UMR 7271), Nice, France
fmichel@i3s.unice.fr

Abstract. While the data deluge accelerates, most of the data produced remains locked in deep Web databases. For the linked open data to benefit from the potential represented by this huge amount of data, it is crucial to come up with solutions to expose heterogeneous databases as linked data. The xR2RML mapping language is an endeavor towards this goal: it is designed to map various types of databases to RDF, by flexibly adapting to heterogeneous query languages and data models while remaining free from any specific language. It extends R2RML, the W3C recommendation for the mapping of relational databases to RDF, and relies on RML for the handling of various data formats.

In this paper we present xR2RML, we analyse data models of several modern databases as well as the format in which query results are returned, and we show how xR2RML translates any result data element into RDF, relying on existing languages such as XPath and JSONPath when necessary. We illustrate some features of xR2RML such as the generation of RDF collections and containers, and the ability to deal with mixed data formats. We also describe a real-world use case in which we applied xR2RML to build a SKOS thesaurus aimed at supporting studies on History of Zoology, Archaeozoology and Conservation Biology.

Keywords: Linked data · RDF · R2RML · NoSQL · History of zoology

1 Introduction

The web of data is now emerging through the publication and interlinking of various open data sets in RDF. Initiatives such as the W3C Data Activity[1] and the Linking Open Data (LOD) project[2] aim at Web-scale data integration and processing, assuming that making heterogeneous data available in a common machine-readable format should create opportunities for novel applications and services. Their success largely depends on the ability to reach data from the deep web [1], a part of the web content consisting of documents and databases hardly linked with other data sources and hardly indexed by standard

[1] http://www.w3.org/2013/data/.

[2] http://linkeddata.org/.

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 275–296, 2016.
DOI: 10.1007/978-3-319-30996-5_14

search engines. The deep web keeps growing as data is continuously accumulated in ever more heterogeneous databases. In particular, NoSQL systems have gained a remarkable success during recent years. Driven by major web companies, they have been developed to meet requirements of web 2.0 services, that relational databases (RDB) could not achieve (flexible schema, high throughput, high availability, horizontal elasticity on commodity hardware). Thus, NoSQL systems should now be considered as potential heavy contributors of linked open data. Other types of databases have been developed over time, either for generic purpose or specific domains, such as XML databases (notably used in edition and digital humanities), object-oriented databases or directory-based databases.

Significant efforts have been invested in the definition of methods to translate various kinds of data sources into RDF. R2RML [2], for instance, is the W3C recommendation to describe RDB-to-RDF mappings. RML extends R2RML for the integration of heterogeneous data formats [3]. To our knowledge though, no method has been proposed yet to tackle NoSQL-to-RDF translation.

In this paper, we present xR2RML, a mapping language designed as an extension of R2RML and RML. Besides relational databases, xR2RML addresses the mapping of a large and extensible scope of non-relational databases to RDF. It is designed to flexibly adapt to various data models and query languages, it can translate data with mixed formats and generate RDF collections and containers. xR2RML is exemplified and validated through a real-world use case. A large-scale SKOS thesaurus aimed at supporting studies on History of Zoology, Archaeozoology and Conservation Biology is thus generated.

In the rest of this section we draw a picture of other works pertaining to the translation of various data sources to RDF, and we scope the objectives of xR2RML. Section 2 explores in more details the capabilities required to reach these goals. In Sect. 3 we recall the main characteristics of R2RML and RML, and in Sect. 4 we describe xR2RML specific extensions. Section 5 presents a working implementation of the language. Section 6 describes the experimentation we ran to create a large taxonomical reference in SKOS. Finally Sects. 7 and 8 discuss xR2RML applicability in different contexts and concludes by outlining some perspectives.

1.1 Related Works

Wrapper-based data integration systems like Garlic [4] and SQL/MED [5] generally have similar architectures: a global data model is described using specific modelling languages (e.g. Garlic's GDL), a query federation engine handles user queries expressed in terms of a global data model and determines a query plan, a per-data source wrapper implements a specific wrapper interface and performs the mapping with the data source schema. No guideline is provided as to how a wrapper should describe and implement the mapping.

The same global architecture holds in data integration systems based on semantic web technologies. Existing works focus on efficient query planning and distribution, such as FedX [6], Anapsid [7] and KGRAM-DQP [8]. The global data model is expressed by domain ontologies using common languages, e.g. RDFS or OWL. User queries, expressed in terms of the domain ontologies, are written in SPARQL. SPARQL is also used as the wrapper interface. Each data source wrapper is a SPARQL endpoint that performs the schema mapping with the source schema. Our work, as well as most related works listed below, focuses on the mapping step: the rationale is to standardize the schema mapping description, so that a mapping description can be written once and applied with different wrapper implementations.

RDB-to-RDF mapping has been an active field of research during the last ten years [9–11]. Several mapping methods and languages have been proposed over time, based either on the materialization of RDF data sets or on the SPARQL-based access to relational data. Published in 2012, R2RML, the W3C RDB-to-RDF mapping language recommendation, has reached a notable consensus[3].

Similarly, various solutions exist to map XML data to RDF. The XSPARQL query language [12] combines XQuery and SPARQL for bidirectional transformations between XML and RDF. Several other solutions are based on the XSLT technology such as XML Scissor-lift [13] that describes mapping rules in Schematron XML validation language, and AstroGrid-D [14]. SPARQL2XQuery [15] applies XML Schema to RDF/OWL translation rules.

Much work has already been accomplished regarding the translation of CSV, TSV and spreadsheets to RDF. Tools have been developed such as XLWrap [16] and RDF Refine[4]. The Linked CSV[5] format is a proposition to embed metadata in a CSV file, that makes it easy to link on the Web and eventually to translate to RDF or JSON. However this approach assumes that CSV data is made compliant with the format in the first place, before it can be translated to RDF. The CSV on the Web W3C Working Group[6], created in 2014, intends to propose a recommendation for the description of and access to CSV data on the Web. In this context, RDF is one of the formats targeted either to represent metadata about CSV data, or as a format to translate CSV data into.

Several tools are designed as frameworks for the integration of sources with heterogeneous data formats. XSPARQL, cited above, provides an R2RML-compliant extension. Thus it can simultaneously translate relational, XML and RDF data to XML or RDF. TARQL[7] is a SPARQL-based mapping language that can convert from RDF, CSV/TSV and JSON formats to RDF, but it does not focus on how the data is retrieved from different types of databases. Datalift [17] provides an integrated set of tools for the publication in RDF of raw structured data (RDB, CSV, XML) and the interlinking of resulting data sets.

RML [3,18] is an extension of R2RML that tackles the mapping of data sources with heterogeneous data formats such as CSV/TSV, XML or JSON. Most approaches create links between data sets after they were translated to RDF, e.g. using properties `rdfs:seeAlso` or `owl:sameAs`. This is sometimes not adequate as logical resources having different identifiers in different data sets

[3] http://www.w3.org/2001/sw/rdb2rdf/wiki/Implementations.

[4] http://refine.deri.ie/.

[5] http://jenit.github.io/linked-csv/.

[6] http://www.w3.org/2013/csvw/wiki.

[7] https://github.com/cygri/tarql/wiki/TARQL-Mapping-Language.

cannot easily be reconciled. RML creates linked data sets at mapping time by enabling the simultaneous mapping of multiple data sources, thus allowing for cross-references between resources defined in various data sources. However, RML does not investigate the constraints that arise when dealing with different types of databases. It proposes a solution to reference data elements within query results using expressive languages such as XPath and JSONPath. But it does not clearly distinguish between such languages and the actual query language of a database. In some cases they might be the same, e.g. XPath can be used to query an XML native database, and later on to reference data elements from query results. But in the general case, the query language and the language used to reference elements within query results must be dissociated, e.g. NoSQL document stores use proprietary query languages, while results are JSON documents that can be evaluated against JSONPath expressions. Furthermore, RML explicitly refers to known evaluation languages (ql:JSONPath, ql:XPath). In this context, supporting a new evaluation language requires to change the mapping language definition. To achieve more flexibility, we believe that such characteristics should be implementation-dependent, leaving the mapping language free from any explicit dependency.

1.2 Objectives of This Work

The works presented in Sect. 1.1 address various types of data sources. Some of them could be extended to new data sources by developing ad-hoc extensions, although they are generally not designed to easily support new data models and query languages. Only RML comes with this flexibility as its design aims at adapting to new data models. Our goal with xR2RML is to define a generic mapping language able to equally apply to most common relational and non-relational databases. We make a specific focus on NoSQL and XML native databases, and we argue that our work can be generalized to some other types of database, for instance object-oriented and directory (LDAP) databases. In Sect. 2 we explore the capabilities required by xR2RML to reach these goals.

Moreover, we describe a validation of xR2RML made in the context of a real-world use case: the translation of data from a MongoDB NoSQL document store into a large-scale RDF-based thesaurus aimed at supporting studies on History of Zoology, Archaeozoology and Conservation Biology.

2 xR2RML Language Requirements

Different kinds of databases typically differ in several aspects: the query language used to retrieve data, the data model that underlies the data structures retrieved and the cross-data referencing scheme, if any. Below we explore in further details the capabilities that we want xR2RML to provide.

Query Languages. The landscape of modern database systems shows a vast diversity of query languages. Relational databases generally support ANSI SQL, and most native XML databases support XPath and XQuery. By contrast, NoSQL is a catch-all term referring to very diverse systems [19,20]. They have heterogeneous access methods ranging from low-level APIs to expressive query languages. Despite several propositions of common query language (N1QL[8], UnQL[9], SQL++ [21], ArangoDB QL[10], CloudMdsQL [22]), no consensus has emerged yet, that would fit most NoSQL databases. Therefore, until a standard eventually arises, xR2RML must be agile enough to cope with various query languages and protocols in a transparent manner.

Data Models. Similarly to the case of query languages, we observe a large heterogeneity in data models of modern databases. To describe their translation to RDF, a mapping language must be able to reference any data element from their data models. Below we list most common data models, we shortly analyse formats in which data is retrieved and figure out how a mapping language can reference data elements within retrieved data.

Relational databases comply with a row-based model in which column names uniquely reference cells in a row. NoSQL extensible column stores[11] also comply with the row-based model, with the difference that all rows do not necessarily share the same columns. For such systems, referencing data elements is simply achieved using column names. Other non-relational systems, such as XML native databases, NoSQL key-value stores, document stores or graph stores, have heterogeneous data models that can hardly be reduced to a row-based model:

- In databases relying on a specific data representation format like JSON (notably in NoSQL document stores) and XML, data is stored and retrieved as documents consisting of tree-like compound values. Referencing data elements within such documents can be achieved thanks to languages such as JSONPath and XPath.
- Object-oriented databases conventionally provide methods to serialize objects, typically as key-value associations: keys are attribute names while values are objects (composition or aggregation relationship), or compound values (collection, map, etc.). Serialization is typically done in XML or JSON, thus here again we can apply XPath or JSONPath expressions.
- A directory data model is organised as a tree: each node has an identifier and a set of attributes represented as `name=value`. Each entry retrieved from an LDAP request is named using an LDAP path expression, e.g. `cn=Franck Michel, ou=cnrs, o=fr`. Referencing data elements within such entries can be simply achieved using attribute names.
- In graph databases, the abstract data model basically consists of nodes and edges. Query capabilities generally allow to retrieve either values matching specific patterns (like the SPARQL SELECT clause), or a set of nodes and edges representing a result graph (like the SPARQL CONSTRUCT clause). Whatever the type of result though, graph databases commonly provide APIs

[8] http://www.couchbase.com/communities/n1ql.

[9] http://unql.sqlite.org/index.html.

[10] http://docs.arangodb.org/Aql/README.html.

[11] aka. column family store, column-oriented store, etc.

to manipulate query results. For instance a SPARQL SELECT result set has a row-based format: each row of a result set consists of columns typically named after query variable names. The Neo4J graph database provides a JDBC interface to process a query result, and its REST interface returns result graphs as JSON documents. Thus, although a graph may be a somehow complex data structure, query results can be fairly easy to manipulate using well-known formats: a row-column model, a serialization in JSON or some other representation syntax, etc.

Finally, the way a mapping language can reference data elements within query results depends more on the API capabilities than the data model itself. To be effective, xR2RML must transparently accept any type of data element reference expression. This includes a column name (applicable not only to row-based data models but also to any row-based query result), JSONPath, XPath or LDAP path expressions, etc. An xR2RML processing engine must be able to evaluate such expressions against query results, but the mapping language itself must remain free from any reference to specific expression syntaxes.

Collections. Many data models support the representation of collections: these can be sets, arrays or maps of all kinds (sorted or not, with or without duplicates, etc.). Although the RDF data model supports such data structures, to the best of our knowledge, existing mapping languages do not allow for the production of RDF collections (`rdf:List`) nor RDF containers (`rdf:Bag`, `rdf:Seq`, `rdf:Alt`), except TARQL that is able to convert a JSON array into an `rdf:List`. In all other cases, structured values such as collections or key-value associations are flattened into multiple RDF triples. Listing 1.1 is an example XML collection consisting of two "movie" elements.
Its translation into two triples is illustrated in Listing 1.2. Assuming that the order of "movie" elements implicitly represents the chronological order in which movies were shot, triples in Listing 1.2 lose this information. Using an RDF sequence may be more appropriate in this case, as illustrated in Listing 1.3.

```
<director name="Woody Allen">
    <movie>Annie Hall</movie>
    <movie>Manhattan</movie>
</director>
```

Listing 1.1. Example of XML collection.

```
<http://example.org/dir/Woody\%20Allen>
  ex:directed "Annie Hall".
<http://example.org/dir/Woody\%20Allen>
  ex:directed "Manhattan".
```

Listing 1.2. Translation to multiple RDF triples.

```
<http://example.org/dir/Woody\%20Allen>
ex:movieList [ a rdf:Seq;
  rdf:_1 "Annie Hall"; rdf:_2 "Manhattan"].
```

Listing 1.3. Translation to an RDF sequence.

Consequently, to map heterogeneous data to RDF while preserving concepts such as collections, bags, alternates or sequences, xR2RML must be able to map data elements to RDF collections and containers.

Cross-References. Cross-references are commonly implemented as foreign key constraints in relational data models, or aggregation and composition relationships in object-oriented models. Cross-referencing is even the primary goal of graph-based databases. More generally, it is possible to cross-reference logical entities in any type of database. For instance, a JSON document of a NoSQL document store may refer to another document by its identifier or any other field that identifies it uniquely, even if this is generally not recommended for the sake of performances.

A cross-referenced logical resource may be mapped alternatively as the subject or the object of triples. This may entail joint queries between tables or documents. Therefore, xR2RML must (i) allow a modular description so that the mapping of a logical resource can be written once and easily reused as a subject or an object, and (ii) allow the description of joint queries to retrieve cross-referenced logical resources.

Summary. Finally, we draw up the list of key capabilities expected from xR2RML as follows:

1. It enables to describe the mapping of various relational and non-relational databases to RDF.
2. It is flexible enough to allow for new databases, query languages and data models in an agile manner: supporting a new system, query language and/or data model only requires changes in the implementation (adaptor, plug-in, etc.), but no changes are required in the mapping language itself.
3. It enables to generate RDF collections (`rdf:List`) or containers (`rdf:Seq`, `rdf:Bag`, `rdf:Alt`) from one-to-many relations modelled as compound values or as cross-references. RDF collections and containers can be nested.
4. It enables to perform joint queries following cross-references between logical resources, and it allows the modular reuse of mapping definitions.

Additionally, data sources to be mapped to RDF using xR2RML should provide a **declarative** query language. If not, it must be possible to fetch the whole data at once, like a CSV or XML file returned by a Web service. There must exist technical means to parse query results, ranging from simple column names to expressive languages like XPath. In case of large data sets, the database interface should provide ways to iterate on query results, similarly to SQL cursors in RDBs.

To help in the design of xR2RML we chose to leverage R2RML, a standard, well-adopted mapping language for relational databases. R2RML already provides some of the requirements listed above: modularity, management of cross-references, as well as rich features such as the ability to define target named graphs. To facilitate its understanding and adoption, xR2RML is designed as a backward compatible extension of R2RML. Besides, to address the mapping of heterogeneous data formats such as CSV/TSV, XML and JSON, we leverage propositions of RML that is itself an extension of R2RML.

3 R2RML and RML

R2RML is a generic language meant to describe customized mappings that translate data from a relational database into an RDF data set. An R2RML mapping is expressed as an RDF graph written in Turtle syntax[12]. An R2RML mapping graph consists of *triples maps*, each one specifying how to map rows of a logical table to RDF triples. A triples map is composed of exactly one *logical table* (property `rr:logicalTable`), one *subject map* (property `rr:subjectMap`) and any number of *predicate-object maps* (property `rr:predicateObjectMap`). A logical table may be a table, an SQL view (property `rr:tableName`), or the result of a valid SQL query (property `rr:sqlQuery`). A predicate-object map consists of *predicate maps* (property `rr:predicateMap`) and *object maps* (property `rr:objectMap`). For each row of the logical table, the subject map generates a subject IRI, while each predicate-object map creates one or more predicate-object pairs. Triples are produced by combining the subject IRI with each predicate-object pair. Additionally, triples are generated either in the default graph or in a named graph specified using *graph maps* (property `rr:graphMap`).

Subject, predicate, object and graph maps are all R2RML *term maps*. A term map is a function that generates RDF terms (either a literal, an IRI or a blank node) from elements of a logical table row. A term map must be exactly one of the following: a *constant-valued term map* (property `rr:constant`) always generates the same value; a *column-valued term map* (property `rr:column`) produces the value of a given column in the current row; a *template-valued term* map (property `rr:template`) builds a value from a template string that references columns of the current row.

When a logical resource is cross-referenced, typically by means of a foreign key relationship, it may be used as the subject of some triples and the object of some others. In such cases, a *referencing object map* uses IRIs produced by the subject map of a (parent) triples map as the objects of triples produced by another (child) triples map. In case both triples maps do not share the same logical table, a joint query must be performed. A join condition (property `rr:joinCondition`) names the columns from the parent and child triples maps, that must be joined (properties `rr:parent` and `rr:child`).

Below we provide a short illustrative example. Triples map `<#R2RML_Directors>` uses table `DIRECTORS` to create triples linking movie directors (whose IRIs are built from column `NAME`) with their birth date (column `BIRTH_DATE`).

```
<#R2RML_Directors>
 rr:logicalTable [ rr:tableName "DIRECTORS" ];
 rr:subjectMap [
    rr:template "http://example.org/dir/{NAME}" ];
 rr:predicateObjectMap [
   rr:predicate ex:bithdate;
   rr:objectMap [
     rr:column "BIRTH_DATE"; rr:datatype xsd:date ] ].
```

[12] http://www.w3.org/TR/turtle/.

RML is an extension of R2RML that targets the simultaneous mapping of heterogeneous data sources with various data formats, in particular hierarchical data formats. An RML logical source (property `rml:logicalSource`) extends R2RML logical table and points to the data source (property `rml:source`): this may be a file on the local file system, or data returned from a web service for instance. A reference formulation (property `rml:referenceFormulation`) names the syntax used to reference data elements within the logical source. As of today, possible values are `ql:JSONPath` for JSON data, `ql:XPath` for XML data, and `rr:SQL2008` for relational databases. Data elements are referenced with property `rml:reference` that extends `rr:column`. Its object is an expression whose syntax matches the reference formulation. Similarly, the definition of property `rr:template` is extended to allow such reference expressions to be enclosed within curly braces ('{' and '}'). Below we provide an RML example. It is very similar to the R2RML example above, with the difference that data now comes from a JSON file "directors.json".

```
<#RML_Directors>
 rml:logicalSource [
   rml:source "directors.json";
   rml:referenceFormulation ql:JSONPath;
   rml:iterator "$.*";  ];
 rr:subjectMap [
   rr:template "http://example.org/dir/{$.*.name}" ];
 rr:predicateObjectMap [
   rr:predicate ex:bithdate;
   rr:objectMap [
     rml:reference "$.*.bithdate"; rr:datatype xsd:date ] ].
```

4 The xR2RML Mapping Language

In this section we briefly describe the elements of the xR2RML language. A complete specification is provided in [23]. We illustrate the descriptions with a running example: Listing 1.4 shows JSON documents stored in a MongoDB database, in two collections: a "directors" collection with documents on movie directors, and a "movies" collection in which movies are grouped in per-decade documents. Listing 1.5 shows an xR2RML mapping graph to translate those documents into RDF. Director IRIs are built using director names, while movie IRIs use movie codes. We assume the following namespace prefix definitions:

```
@predfix xrr: <http://www.i3s.unice.fr/ns/xr2rml#>.
@predfix rr: <http://www.w3.org/ns/r2rml#>.
@predfix rml: <http://semweb.mmlab.be/ns/rml#>.
@predfix xsd: <http://www.w3.org/2001/XMLSchema#>.
@predfix ex: <http://example.com/ns#>.
```

4.1 Describing a Logical Source

To reach its genericity objective, xR2RML must avoid explicitly referring to specific query languages or data models. Keeping this in mind, we define logical sources as a mean to represent a data set from any kind of database. In conformance with R2RML principles, we keep database connection details out of the scope of the mapping language. In RML on the other hand, a logical source points to the data to be mapped typically using a file URL (property `rml:source`). This difference makes it difficult for xR2RML to extend RML's logical source concept. Instead, xR2RML extends the R2RML logical source while commonalities are addressed by using or extending some RML properties (`rml:referenceFormulation`, `rml:query`, `rml:iterator`).

xR2RML triples maps extend R2RML triples maps by referencing a *logical source* (property `xrr:logicalSource`) which is the result of a request applied to the input database. It is either an *xR2RML base table* or an *xR2RML view*. The xR2RML base table extends the concept of *R2RML table or view* to tabular databases beyond relational databases (extensible column store, CSV/TSV, etc.). It refers to a table by its name (property `rr:tableName`). An xR2RML view represents the result of executing a query against the input database. It has exactly one `xrr:query` property that extends RML property `rml:query` (which itself extends `rr:sqlQuery`[13]). Its value is a valid expression with regards to the query language supported by the input database. No assumption is made whatsoever as to the query language used.

```
Collection "directors":
{"name": "Woody Allen", "directed":
    ["Manhattan", "Interiors"]},
{"name": "Wong Kar-wai", "directed":
    ["2046", "In the Mood for Love"]}

Collection "movies":
{"decade": "2000s", "movies": [
    {"name": "2046", "code": "m2046",
    "actors": ["T. Leung", "G. Li"]},
    {"name": "In the Mood for Love", "code": "Mood",
    "actors": ["M. Cheung"]} ] }
{"decade": "1970s":, "movies": [
    {"name": "Manhattan", "code": "Manh",
    "actors": ["Woody Allen", "Diane Keaton"]}
    {"name": "Interiors", "code": "Int01",
    "actors": ["D. Keaton", "G. Page"]} ] }
```

Listing 1.4. Example Database.

Reference Formulation. Retrieving values from a query result set requires evaluating *data element references* against the query result. Relational database APIs (such as JDBC drivers) support the evaluation of a column name against the current row of a result set. Conversely, some databases come with simple APIs that provide lower level evaluation features. For instance, APIs of most

[13] rml:query also subsumes rml:xmlQuery and rml:queryLanguage, although none of those properties are described or exemplified in the RML language specification and articles at the time of writing.

NoSQL document stores return JSON documents but hardly support JSONPath. Therefore, the xR2RML processing engine is responsible for evaluating such data element references. To do so, it needs to know which syntax is being used. To this end, RML introduced the reference formulation concept (property `rml:referenceFormulation` of a logical source) to name the syntax of data element references. As underlined above, xR2RML adheres to R2RML's principle that database-specific details be kept out of the scope of the mapping language. We also want the mapping language to remain free from explicit reference to specific syntaxes. As a result, we amend the R2RML processor definition as follows: *an xR2RML processor must be provided with a database connection and the reference formulation applicable to results of queries run against the connection. If the reference formulation is not provided, it defaults to column name, in order to ensure backward compatibility with R2RML.*

Iteration Model. In R2RML, the row-based iteration occurs on a set of rows read from a logical table. xR2RML applies this principle to other systems returning row-based result sets: CSV/TSV files, extensible column stores, but also some graph databases as underlined in Sect. 2, e.g. a SPARQL SELECT result set is a table in which columns are named after the variables in the SELECT clause. In the context of non row-based result sets, the model is implicitly extended to a *document-based iteration model*: a document is basically one entry of a result set returned by the database, e.g. a JSON document retrieved from a NoSQL document store, or an XML document retrieved from an XML native database. In the case of data sources whose access interface does not provide built-in iterators, e.g. a web service returning an XML response at once, then a single iteration occurs on the whole retrieved document.

Yet, some specific needs may not be fulfilled. For instance, it may be needed to iterate on explicitly specified entries of a JSON document or elements of an XML tree. To this end, we leverage the concept of iterator introduced in RML. An iterator (property `rml:iterator`) specifies the iteration pattern to apply to data read from the input database. Its value is a valid expression written using the syntax specified in the reference formulation. The iterator can be either omitted or empty when the reference formulation is a column name.

Listing 1.5 presents two logical source definition examples. Both consist of a MongoDB query (property `xrr:query`). We assume that the JSONPath reference formulation is provided along with the database connection. In collection "directors" (Listing 1.4), each document describes exactly one director. By contrast, in collection "movies" each document refers to several movies grouped by decade. To avoid mixing up multiple movies of a single document, an iterator with JSONPath expression `$.movies.*` is associated with triples map `<#Movies>`: thus, the triples map applies separately on each movie of each document.

4.2 Referencing Data Elements

In Sect. 3 we have seen that RML properties `rml:reference` and `rr:template` both allow data element references expressed according to the reference formulation

```
<#Movies>
  xrr:logicalSource [
    xrr:query "db.movies.find({decade:{$exists:true}})";
    rml:iterator "$.movies.*";
  ];
  rr:subjectMap [
    rr:template "http://example.org/movie/{$.code}" ];
  rr:predicateObjectMap [
    rr:predicate ex:starring;
    rr:objectMap [
      rr:termType xrr:RdfBag;
      xrr:reference "$.actors.*";
      xrr:nestedTermMap [ rr:datatype xsd:string ] ] ].

<#Directors>
  xrr:logicalSource [ xrr:query "db.directors.find()" ];
  rr:subjectMap [
    rr:template "http://example.org/dir/{$.name}" ];
  rr:predicateObjectMap [
    rr:predicate ex:directed;
    rr:objectMap [
      rr:parentTriplesMap <#Movies>;
      rr:joinCondition [
        rr:child "$.directed.*";
        rr:parent "$.name" ] ] ].
```

Listing 1.5. xR2RML Example Mapping Graph.

(column name, XPath, JSONPath). xR2RML uses these RML definitions as a starting point to a broader set of use cases.

In real world use cases, databases commonly store values written in a data format that they cannot interpret. For instance, in key-value stores and in most extensible column stores, values are stored as binary objects whose content is opaque to the system. A developer may choose to embed JSON, CSV or XML values in the column of a relational table, for performance issues or due to application design constraints. We call such cases *mixed content.*

xR2RML proposes to apply the principle of data element references defined in RML, and extend it to allow referencing data elements within mixed content. An xR2RML *mixed-syntax path* consists of the concatenation of several path expressions, each path being enclosed in a *syntax path constructor* that explicits the path syntax. Existing constructors are: Column(), CSV(), TSV(), JSONPath() and XPath(). For example, in a relational table, a text column `NAME` stores JSON-formatted values containing people's first and last names, e.g.: {`"First":"John", "Last":"Smith"`}. Field `FirstName` can be referenced with the following mixed-syntax path: `Column(NAME)/JSONPath($.First)`. An xR2RML processing engine evaluates a mixed-syntax path from left to right, passing the result of each path constructor on to the next one. In this example, the first path retrieves the value associated with column `NAME`. Then the value is passed on to the next path constructor that evaluates JSONPath expression "$.First" against the value. The resulting value is finally translated into an RDF term according to the current term map definition.

xR2RML defines property `xrr:reference` as an extension of RML property `rml:reference`, and extends the definition of property `rr:template`. Both properties accept either simple references (illustrated in Listing 1.5) or mixed-syntax path expressions.

4.3 Producing RDF Terms and (Nested) RDF Collections/Containers

In a row-based logical source, a valid column name reference returns zero or one value during each triples map iteration. In turn an R2RML term map generates zero or one RDF term per iteration. By contrast, JSONPath and XPath expressions used with properties `xrr:reference` and `rr:template` allow addressing multiple values. For instance, XPath expression `//movie/name` returns all `<name>` elements of all `<movie>` elements. Therefore, reference-valued and template-valued term maps can return multiple RDF terms at once. This difference entails the definition of two strategies with regards to how triples maps combine RDF terms to build triples: the product strategy, and the collection/container strategy.

Product Strategy. During each iteration of an xR2RML triples map, triples are generated as the product between RDF terms produced by the subject map and each predicate-object pair. Predicate-object pairs result of the product between RDF terms produced by the predicate maps and object maps of each predicate-object map. Like any other term map, a graph map may also produce multiple terms. The product strategy equally applies in that case, therefore triples are produced simultaneously in all target graphs corresponding to the multiple RDF terms produced by the graph map.

Collection/Container Strategy. Multiple values returned by properties `xrr:reference` and `rr:template` are combined into an RDF collection or container. This is achieved using new xR2RML values of the `rr:termType` property: a term map with term type `xrr:RdfList` generates an RDF term of type `rdf:List`, term type `xrr:RdfSeq` corresponds to `rdf:Seq`, `xrr:RdfBag` to `rdf:Bag` and `xrr:RdfAlt` to `rdf:Alt`. Listing 1.5 illustrates this case: instead of generating multiple triples relating each movie to one actor, triples map `<#Movies>` relates each movie to a bag of actors starring in that movie. For instance:

```
<http://example.org/movie/m2046> ex:starring
    [ a rdf:Bag; rdf:_1 "Tony Leung"; rdf:_2 "Gong Li" ].
```

At this point, two important needs must still be addressed in the collection/container strategy: (i) like in a regular term map, it must be possible to assign a term type, language tag or data type to the members of an RDF collection or container; and (ii) it must be possible to nest any number of RDF collections and containers inside each-other. Both needs are fulfilled using *xR2RML Nested Term Maps*. A nested term map (property `xrr:nestedTermMap`) very much resembles a regular term map, with the exception that it can be defined only in the context of a term map that produces RDF collections or containers. In a column-valued or reference-valued term map, a nested term map describes how to translate values read from the logical source into RDF terms, by specifying optional properties `rr:termType`, `rr:language` and `rr:datatype`. Similarly, in a template-valued term map, a nested term map applies to values produced by applying the template string to input values. In Listing 1.5, triples map `<#Movie>` uses a nested term map to assign a `xsd:string` datatype to names of names starring in a movie. For instance:

```
<http://example.org/movie/m2046> ex:starring [ a rdf:Bag;
rdf:_1 "Tony Leung"^^xsd:string; rdf:_2 "Gong Li"^^xsd:string ].
```

Finally, properties `xrr:reference` and `rr:template` can be used within a nested term map to recursively parse structured values while producing nested RDF collections and containers.

4.4 Reference Relationships Between Logical Sources

A cross-referenced logical resource usually serves as the subject of some triples and the object of other triples. In R2RML, this is achieved using a referencing object map. xR2RML extends R2RML referencing object maps in two ways. Firstly, when a joint query is needed (i.e. the parent and child triples map do not share the same logical source), properties `rr:child` and `rr:parent` of the join condition contain data element references (4.2), possibly including mixed-syntax paths. As underlined in Sect. 4.3, such data element references may produce multiple terms. Consequently, the equivalent joint query of a referencing object map must deal with multi-valued child and parent references. More precisely, a join condition between two multi-valued references should be satisfied if at least one data element of the child reference matches one data element of the parent reference. This is described in Definition 1 using an SQL-like syntax and first order logic for the description of WHERE conditions.

Definition 1: If a referencing object map has at least one join condition, then its equivalent joint query is:

$$\begin{gathered}\textit{SELECT * FROM (child-query) AS child, (parent-query) AS parent}\\ \textit{WHERE}\\ \exists c_1 \in eval(child, \{child - ref_1\}),\ \exists p_1 \in eval(parent, \{parent - ref_1\}), c_1 = p_1\\ \textit{AND}\\ \exists c_2 \in eval(child, \{child - ref_2\}),\ \exists p_2 \in eval(parent, \{parent - ref_2\}), c_2 = p_2\\ \textit{AND ...}\end{gathered}$$

where "{child-ref i}" and "{parent-ref i}" are the child and parent references of the i^{th} *join condition, and "eval(child, {ref})" and "eval(parent, {ref})" are the result of evaluating data element reference "{ref}" on the result of the child and parent queries.*

Listing 1.5 depicts a simple example: in triples map `<#Directors>`, the object map uses movie IRIs generated by parent triples map `<#Movies>`. When processing director "Wong Kar-wai", the child reference ($.directed.*) returns values

"2046" and "In the Mood for Love", while the parent reference ($.name) returns a single movie name. The join condition is satisfied if the parent reference returns one of "2046" or "In the Mood for Love". Generated triples use movie codes to build movie IRIs, such as:

```
<http://example.org/dir/Wong%20Kar-wai>
   ex:directed <http://example.org/movie/m2046>.
```

Secondly, the objects produced by a referencing object map can be grouped in an RDF collection or container, instead of being the objects of multiple triples. To do so, an xR2RML referencing object map may have a `rr:termType` property with value `xrr:RdfList`, `xrr:RdfSeq`, `xrr:RdfBag` or `xrr:RdfAlt`. Results of the joint query are grouped by child value, i.e. objects generated by the parent triples map, referring to the same child value, are grouped as members of an RDF collection or container. An interesting consequence of this use case is the ability, in the case of a regular relational database, to build an RDF collection or container reflecting a one-to-many relation.

5 Implementation

To evaluate the effectiveness of xR2RML, we have developed Morph-xR2RML, an open source prototype implementation available on Github[14]. It is written in Scala and is an extension of Morph-RDB [24], an R2RML implementation.

In a first step, we upgraded Morph-RDB to support xR2RML features in the context of relational databases. This included the support of logical sources, mixed contents (JSON, XML, CSV or TSV data embedded in cells) and RDF collections/containers. In a second step, we developed a connector to the MongoDB document store, to translate MongoDB JSON documents into RDF. A MongoDB query string is specified in each triples map logical source. The connector executes the query and iterates over result documents returned by the database. Subsequently, results are passed to the xR2RML processor that applies the optional iterator (`rml:iterator`) and evaluates JSONPath expressions in each `xrr:reference` and `rr:template` property of all term maps. The support of RDF collections/containers was validated, including in the case of referencing object maps that entail a joint query between two JSON documents.

We evaluated the prototype using two simple databases: a MySQL relational database and a MongoDB database with two collections. In both cases, the data and associated xR2RML mappings were written to cover most mapping situations addressed by xR2RML: strategies for handling multiple RDF terms, mixed-syntax paths with mixed contents (relational, JSON, XML, CSV/TSV), cross-references, RDF collection/containers, UTF-8 encoding. Both databases as well as the example mappings are available on the GitHub repository. The current status of the prototype applies the data materialization approach, i.e. RDF data is generated by sequentially applying all triples maps. The query rewriting approach (SPARQL to database specific query rewriting) may be considered in

[14] https://github.com/frmichel/morph-xr2rml/.

future work as suggested in Sect. 8. At the time of writing the prototype has two limitations: (i) only one level of RDF collections and containers can be generated (no nested collections/containers), and (ii) the result of a joint query in a relational database cannot be translated into an RDF collection or container.

6 Validation: Construction of a SKOS Zoological and Botanical Reference Thesaurus

The Zoomathia research network[15] studies the transmission of zoological knowledge throughout historical periods. It intends to leverage the Semantic Web technologies to annotate and link together various resources such as rich medieval compilation literature on Ancient zoological knowledge, archaeozoological data from excavation reports, iconographic material and modern conservation biology knowledge. This challenging goal can be addressed through the use of controlled and widely accepted semantic references. In this context, the TAXREF [25] zoological and botanical taxonomy has been chosen to build a SKOS thesaurus[16] supporting the integration of these heterogeneous data sets. SKOS, the Simple Knowledge Organization System, is a W3C standard designed to represent controlled vocabularies, taxonomies and thesauri. It is extensively used to bridge the gap between existing knowledge organisation systems and the Semantic Web and Linked Data. In this section we shortly present TAXREF and we describe how we used xR2RML for the creation of a SKOS vocabulary faithfully representing TAXREF. More detailed information about TAXREF's content and structure, the SKOS modeling and the data sources to be integrated, can be found in [26].

TAXREF is the French national taxonomic reference for fauna, flora and fungus of metropolitan France and overseas departments and collectivities. It registers 452.106 taxa of living beings from the Palaeolithic until now, covering continental and marine environments. Each taxon is provided along with its scientific name, upper taxon in the classification, synonyms, vernacular names, authority (author name and publication year), taxonomical rank (order, family, gender, species...), type of habitat (marine, terrestrial...) and a biogeographical status (present, endemic, extinct, etc.) for each considered geographical area. TAXREF is developed, maintained and distributed by the French *National Museum of Natural History* (MNHN). It can be browsed, downloaded in TSV format, or queried through a Web service.

In a first step we defined the SKOS modelling of TAXREF[17]. In brief, a SKOS concept is created for each taxon, along with a SKOS label for each scientific (preferred label) name and synonym (alternate label). The SKOS broader property is used to model the relationships between a taxon and the upper taxon in the classi?cation. To ensure proper linkage with well-adopted data sets, in particular within the Linking Open Data cloud, we identified relevant ontologies such

[15] http://www.cepam.cnrs.fr/zoomathia/.

[16] http://www.w3.org/2009/08/skos-reference/skos.html.

[17] Changes were made since the description presented in [26].

as the NCBI taxonomic classification[18], the GeoSpecies ontology[19], the ENVO[20] environment ontology and Biodiversity Information Standards promoted by the Taxonomic Databases Working Group[21]. Links were made either by using appropriate properties and classes, or by aligning taxa with their equivalent in other ontologies.

Listing 1.6 shortly illustrates the SKOS representation of a dolphin species, the taxon “Delphinus delphis”, generated by xR2RML using the Turtle RDF syntax. This consists of a SKOS concept for the taxon and two SKOS labels, one for the reference name and one for the synonym “Delphinus tropicalis”.

In a second step we retrieved a full TAXREF JSON dump from the TAXREF Web service, and imported it into a MongoDB instance. We wrote the xR2RML mappings that describe how to map the result of queries to the MongoDB instance into RDF triples, according to the SKOS modelling. This step requires a good knowledge of the xR2RML language, but it is quite straightforward for users who are already familiar with R2RML.

To perform the translation of TAXREF into SKOS, we ran Morph-xR2RML, the prototype implementation of xR2RML described in Sect. 5, on a laptop equipped with a 3 GHz Intel Core i7 processor and 8 GB of RAM. The resulting RDF graph consists of more than 5 million triples. The translation process required approximately 6 h to complete, which can be considered surprisingly long. We analyze this issue in the next paragraph. Nonetheless, this execution time span should not be considered as an hurdle in the context of TAXREF. Indeed, insofar as TAXREF is updated once a year approximately, the traditional Extract, Transform and Load (ETL) approach is relevant. This is the approach we intend to follow in the future: each time TAXREF is updated, the corresponding SKOS thesaurus is generated. The resulting graph is loaded into a public triple store accessible using either a SPARQL endpoint or the HTTP dereferencing method.

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix dct: <http://purl.org/dc/elements/1.1/> .
@prefix skc: <http://www.w3.org/2004/02/skos/core#>.
@prefix skx: <http://www.w3.org/2008/05/skos-xl#>.
@prefix tc: <http://lod.taxonconcept.org/ontology/txn.owl#>.
@prefix nt: <http://purl.obolibrary.org/obo/ncbitaxon#> .
@prefix dwc: <http://rs.tdwg.org/dwc/terms/> .
@prefix taxr: <http://inpn.mnhn.fr/taxref/> .

<http://inpn.mnhn.fr/taxref/8.0/taxon/60878> a skc:Concept;
   skx:altLabel <http://inpn.mnhn.fr/espece/cd_nom/60881>;
   skx:prefLabel <http://inpn.mnhn.fr/espece/cd_nom/60878>;
   skc:broader <http://inpn.mnhn.fr/taxref/taxon/191591>;
   taxr:hasHabitat <http://inpn.mnhn.fr/taxref/habitat#Marine>;
   nt:has_rank <http://inpn.mnhn.fr/taxref/taxrank#Species>;
   skc:note "Delphinus delphis";
```

[18] http://www.ontobee.org/browser/index.php?o=NCBITaxon.

[19] http://datahub.io/dataset/geospecies.

[20] http://www.ontobee.org/browser/index.php?o=ENVO.

[21] http://www.tdwg.org/.

```
  taxr:bioGeoStatusIn [
   rdfs:label ”Guadeloupe”;
   dct:spatial <http://sws.geonames.org/3579143/>;
   dwc:locationId ”TDWG:LEE-GU; WOEID:23424831”;
   dwc:occurrenceStatus <http://inpn.mnhn.fr/taxref/bioGeoStat#P> ];
  taxr:bioGeoStatusIn [
   rdfs:label ”New Caledonia”;
   dct:spatial <http://sws.geonames.org/2139685/> ;
   dwc:locationId ”TDWG:NWC-OO; WOEID:23424903” ;
   dwc:occurrenceStatus <http://inpn.mnhn.fr/taxref/bioGeoStat#B> ].

<http://inpn.mnhn.fr/espece/cd_nom/60878> a skx:Label;
  skx:literalForm ”Delphinus delphis”;
  tc:authority ”Linnaeus, 1758”;
  taxr:isPrefLabelOf <http://inpn.mnhn.fr/taxref/8.0/taxon/60878>:
  taxr:vernacularName ”Short-beaked common dolphin”@en.

<http://inpn.mnhn.fr/espece/cd_nom/60881> a skx:Label;
  skx:literalForm ”Delphinus tropicalis”.
  tc:authority ”Van Bree, 1971”;
  taxr:isAltLabelOf <http://inpn.mnhn.fr/taxref/8.0/taxon/60878>;
  taxr:vernacularName ”Short-beaked common dolphin”@en.
```

Listing 1.6. Example representation of TAXREF entries in SKOS.

The xR2RML mapping graph for TAXREF consists of 90 triples maps. The high number of triples maps is a consequence of the distance between the internal structure of TAXREF and the targeted SKOS modelling. We illustrate this distance with an example. Habitats are coded in TAXREF with integer values, e.g. value ‘1’ represents the marine habitat. Translating the marine habitat into URI `<http://inpn.mnhn.fr/taxref/habitat#1>` would be straightforward: a template-valued term map could simply append the value ‘1’ read from the database to the namespace `<http://inpn.mnhn.fr/taxref/habitat#>`. Therefore a single triples map (thus a single query) would be sufficient to generate all triples related to all types of habitat. However, we wish to generate more meaningful URIs, such as `<http://inpn.mnhn.fr/taxref/habitat#Marine>`. This URI cannot be generated by a template, instead we have to write a triples map whom query filters only taxa with habitat ‘1’. Similarly, we must write one triples map for each habitat value, that is 8 triples maps. The same situation is observed for the 48 taxonomical ranks and 30 biogeographical statuses, that all result in specific dedicated triples maps. Consequently, many queries have to be run, some of them returning tens or hundreds of thousands of JSON documents. The same JSON documents are retrieved and parsed several times, but, each time, for the generation of triples with different properties. An analysis of the execution traces shows that, out of the 452.106 unique documents in the database, approximately 1.6 million documents are actually retrieved and processed during the translation, that is an approximate average of 4.600 documents treated per minute.

7 Discussion

xR2RML relies on the assumption that databases to translate into RDF provide a declarative query language, such that queries can be expressed directly in a mapping description. This complies with the equivalent assumption of R2RML that all RDBs support ANSI SQL. This is somehow restrictive since some NoSQL key-value stores, like DynamoDB and Riak, have no declarative query language, instead they provide APIs for usual programming languages to describe queries in an imperative manner. For xR2RML to work with those systems, a query language should be figured out along with a compiler that transforms queries into imperative code. Interestingly, this is already the case of some systems supporting the MapReduce programming model. MapReduce is conventionally supported through APIs for programming languages, however more and more systems now propose an SQL or SQL-like query language on top of a MapReduce framework (e.g. Apache Hive). Queries are compiled into MapReduce jobs. This approach is often referred to as SQL-on-Hadoop [27].

To achieve the targeted flexibility, xR2RML comes with features that are applicable independently of the type of database used. Yet, all features should probably not be applied with all kinds of database. For instance, join conditions entail joint queries. Whereas RDBs are optimized to support joins very efficiently, it is not recommended to make cross-references within NoSQL document or extensible column stores, as this may lead to poor performances. Similarly, translating a JSON element into an RDF collection is quite straightforward, but translating the result of an SQL joint query into an RDF collection is likely to be quite inefficient. In other words, because the language makes a mapping possible does not mean that it should be applied regardless of the context (database type, data model, query capabilities). Consequently, mapping designers should be aware of how databases work in order to write efficient mappings of big databases to RDF.

Like R2RML, xR2RML assumes that well-defined domain ontologies exist beforehand, whereof classes and properties will be used to translate a data source into RDF triples. In the context of RDBs, an alternative approach, the Direct Mapping, translates relational data into RDF in a straightforward manner, by converting tables to classes and columns to properties [10,28]. The direct mapping comes up with an ad-hoc ontology that reflects the relational schema. R2RML implementations often provide a tool to automatically generate an R2RML direct mapping from the relational schema (e.g. Morph-RDB [29]). The same principles could be extended to automatically generate an xR2RML mapping for other types of data source, as long as they comply with a schema: column names in CSV/TSV files and extensible column stores, XSD or DTD for XML data, JSON schema[22] or a JSON-LD[23] description for JSON data. Nevertheless, such schemas do not necessarily exist, and some databases like

[22] http://json-schema.org/.

[23] http://www.w3.org/TR/json-ld/.

the DynamoDB key-value store are schema-less. In such cases, automatically generating an xR2RML direct mapping should involve different methods aimed at learning the database schema from the data itself.

More generally, how to automate the generation of xR2RML mappings may become a concern to map large and/or complex schemas. There exists significant work related to schema mapping and matching [30]. For instance, Clio [31] generates a schema mapping based on the discovery of queries over the source and target schemas and a specification of their relationships. Karma [32] semi-automatically maps structured data sources to existing domain ontologies. It produces a Global-and-Local-As-View mapping that can be used to translate the data into RDF. xR2RML does not directly address the question of how mappings are written, but can be complementary of approaches like Clio and Karma. In particular, Karma authors suggest that their tool could easily export mapping rules as an R2RML mapping graph. A similar approach could be applied to discover mappings between a non-relational database and domain ontologies, and export the result as an xR2RML mapping graph.

8 Conclusion and Perspectives

In this paper we presented xR2RML, a language designed to describe the mapping of various types of databases to RDF, by flexibly adapting to heterogeneous query languages and data models. We analysed data models of several modern databases as well as the format in which query results are returned, and we showed that xR2RML can translate any data element within such results into RDF, relying when necessary on existing languages such as XPath and JSONPath. We illustrated some features of xR2RML such as the generation of RDF collections and containers, and the ability to deal with mixed data formats, e.g. when the column of a relational table stores data formatted in another syntax like XML, JSON or CSV.

Principles of the xR2RML mapping language were validated in a prototype implementation supporting several RDBs and the MongoDB NoSQL document store. The prototype was used in a real-world use case to perform the translation of a taxonomical reference of more than 450.000 taxa, represented as JSON documents stored in a MongoDB instance, into a SKOS thesaurus. The data materialization approach we implemented proved to be effective, although the use case underlined scaling limitations with regards to execution time span and memory consumption. In particular, this approach cannot scale to big data sets. Dealing with big data sets requires the data to remain in legacy databases, and that the translation to RDF be performed on demand through the xR2RML-based rewriting of SPARQL queries into the source database query language. In this regard, existing works related to RDBs should be leveraged [24,33].

References

1. He, B., Patel, M., Zhang, Z., Chang, K.C.C.: Accessing the deep web. Commun. ACM **50**, 94–101 (2007)
2. Das, S., Sundara, S., Cyganiak, R.: R2RML: RDB to RDF mapping language (2012)
3. Dimou, A., Sande, M.V., Slepicka, J., Szekely, P., Mannens, E., Knoblock, C., Walle, R.V.: Mapping hierarchical sources into RDF using the RML mapping language. In: Proceedings of ICSC 2014, pp. 151–158. IEEE (2014)
4. Roth, M.T., Schwartz, P.: Don't scrap it, wrap it! A wrapper architecture for legacy data sources. In: Proceedings of VLDB 1997, pp. 266–275 (1997)
5. Melton, J., Michels, J.E., Josifovski, V., Kulkarni, K., Schwarz, P.: SQL/MED: a status report. ACM SIGMOD Rec. **31**, 81–89 (2002)
6. Schwarte, A., Haase, P., Hose, K., Schenkel, R., Schmidt, M.: FedX: optimization techniques for federated query processing on linked data. In: Aroyo, L., Welty, C., Alani, H., Taylor, J., Bernstein, A., Kagal, L., Noy, N., Blomqvist, E. (eds.) ISWC 2011, Part I. LNCS, vol. 7031, pp. 601–616. Springer, Heidelberg (2011)
7. Acosta, M., Vidal, M.-E., Lampo, T., Castillo, J., Ruckhaus, E.: ANAPSID: an adaptive query processing engine for SPARQL endpoints. In: Aroyo, L., Welty, C., Alani, H., Taylor, J., Bernstein, A., Kagal, L., Noy, N., Blomqvist, E. (eds.) ISWC 2011, Part I. LNCS, vol. 7031, pp. 18–34. Springer, Heidelberg (2011)
8. Gaignard, A.: Distributed knowledge sharing and production through collaborative e-science platforms. PhD thesis (2013)
9. Spanos, D.E., Stavrou, P., Mitrou, N.: Bringing relational databases into the semantic web: a survey. Semant. Web J. **3**, 169–209 (2012)
10. Sequeda, J., Tirmizi, S.H., Corcho, S., Miranker, D.P.: Survey of directly mapping SQL databases to the semantic web. Knowl. Eng. Rev. **26**, 445–486 (2011)
11. Michel, F., Montagnat, J., Faron-Zucker, C.: A survey of RDB to RDF translation approaches and tools Research report. ISRN I3S/RR 2013–04-FR (2014)
12. Bischof, S., Decker, S., Krennwallner, T., Lopes, N., Polleres, A.: Mapping between RDF and XML with XSPARQL. J. Data Semant. **1**, 147–185 (2012)
13. Fennell, P.: Schematron - more useful than you'd thought. In: Proceedings of the XML London 2014 Conference, pp. 103–112 (2014)
14. Breitling, F.: A standard transformation from XML to RDF via XSLT. Astron. Not. **330**, 755 (2009)
15. Bikakis, N., Tsinaraki, C., Stavrakantonakis, I., Gioldasis, N., Christodoulakis, S.: The SPARQL2XQuery interoperability framework. CoRR abs/1311.0536 (2013)
16. Langegger, A., Wöß, W.: XLWrap – querying and integrating arbitrary spreadsheets with SPARQL. In: Bernstein, A., Karger, D.R., Heath, T., Feigenbaum, L., Maynard, D., Motta, E., Thirunarayan, K. (eds.) ISWC 2009. LNCS, vol. 5823, pp. 359–374. Springer, Heidelberg (2009)
17. Scharffe, F., Atemezing, G., Troncy, R., Gandon, F., Villata, S., Bucher, B., Hamdi, F., Bihanic, L., Képéklian, G., Cotton, F., et al.: Enabling linked data publication with the Datalift platform. In: Proceedings of the AAAI Workshop on Semantic Cities (2012)
18. Dimou, A., Vander Sande, M., Colpaert, P., Verborgh, R., Mannens, E., Van de Walle, R.: RML: a generic language for integrated RDF mappings of heterogeneous data. In: Proceedings of the 7th LDOW Workshop (2014)
19. Hecht, R., Jablonski, S.: NoSQL evaluation: a use case oriented survey. In: Proceedings of CSC 2011, pp. 336–341. IEEE Computer Society (2011)

20. Gajendran, S.K.: A survey on NoSQL databases (technical report) (2013)
21. Ong, K.W., Papakonstantinou, Y., Vernoux, R.: The SQL++ unifying semi-structured query language, and an expressiveness benchmark of SQL-on-Hadoop, NoSQL and NewSQL databases (submitted). CoRR abs/1405.3631 (2014)
22. Kolev, B., Valduriez, P., Jimenez-Peris, R., Martìnez-Bazan, N., Pereira, J.: Cloud-MdsQL: querying heterogeneous cloud data stores with a common language. In: Proceedings of the BDA 2014 Conference (2014)
23. Michel, F., Djimenou, L., Faron-Zucker, C., Montagnat, J.: xR2RML: Relational and non-relational databases to RDF mapping language (2014). Research report. ISRN I3S/RR 2014–04-FR v3
24. Priyatna, F., Corcho, O., Sequeda, J.: Formalisation and experiences of R2RML-based SPARQL to SQL query translation using Morph. In: Proceedings of WWW 2014 (2014)
25. Gargominy, P., Tercerie, S., Régnier, C., Ramage, T., Schoelinck, C., Dupont, P., Vandel, E., Daszkiewicz, P., Poncet, L.: TAXREF v8.0, référentiel taxonomique pour la France: méthodologie, mise en oeuvre et diffusion. In: Rapport SPN 2014 - 42 (2014)
26. Callou, C., Michel, F., Faron-Zucker, C., Martin, C., Montagnat, J.: Towards a shared reference thesaurus for studies on history of zoology, archaeozoology and conservation biology. In: ESCW 2015, Workshop Semantic Web For Scientific Heritage (SW4SH), Portoroz, Slovenia (2015)
27. Floratou, A., Minhas, U.F., Ozcan, F.: SQl-on-hadoop: Full circle back to shared-nothing database architectures. Proc. VLDB Endowment **7**, 1295–1306 (2014)
28. Arenas, M., Bertails, A., Prud'hommeaux, E., Sequeda, J.: A direct mapping of relational data to RDF (2012)
29. de Medeiros, L.F., Priyatna, F., Corcho, O.: MIRROR: automatic R2RML mapping generation from relational databases. In: Cimiano, P., Frasincar, F., Houben, G.-J., Schwabe, D. (eds.) ICWE 2015. LNCS, vol. 9114, pp. 326–343. Springer, Heidelberg (2015)
30. Shvaiko, P., Euzenat, J.: A survey of schema-based matching approaches. In: Spaccapietra, S. (ed.) Journal on Data Semantics IV. LNCS, vol. 3730, pp. 146–171. Springer, Heidelberg (2005)
31. Fagin, R., Haas, L.M., Hernández, M., Miller, R.J., Popa, L., Velegrakis, Y.: Clio: schema mapping creation and data exchange. In: Borgida, A.T., Chaudhri, V.K., Giorgini, P., Yu, E.S. (eds.) Conceptual Modeling: Foundations and Applications. LNCS, vol. 5600, pp. 198–236. Springer, Heidelberg (2009)
32. Knoblock, C.A., Szekely, P., Ambite, J.L., Goel, A., Gupta, S., Lerman, K., Muslea, M., Taheriyan, M., Mallick, P.: Semi-automatically mapping structured sources into the semantic web. In: Simperl, E., Cimiano, P., Polleres, A., Corcho, O., Presutti, V. (eds.) ESWC 2012. LNCS, vol. 7295, pp. 375–390. Springer, Heidelberg (2012)
33. Sequeda, J.F., Miranker, D.P.: Ultrawrap: SPARQL execution on relational data. Web Semant.: Sci. Serv. Agents WWW **22**, 19–39 (2013)

Pre-trip Ratings and Social Networks User Behaviors for Recommendations in Touristic Web Portals

Silvia Rossi[1](✉), Francesco Barile[2], Antonio Caso[3], and Alessandra Rossi[3]

[1] Dipartimento di Ingegneria Elettrica e Tecnologie Dell'Informazione,
Universita' Degli Studi di Napoli "Federico II", Napoli, Italy
silvia.rossi@unina.it

[2] Dipartimento di Matematica e Applicazioni,
Universita' Degli Studi di Napoli "Federico II", Napoli, Italy
francesco.barile@unina.it

[3] Dipartimento di Fisica, Universita' Degli Studi di Napoli "Federico II",
Napoli, Italy
{antonio.caso,alessandra.rossi}@unina.it

Abstract. Decision-making activities in planning a city visit typically include a pre–visit hunt for information. Hence, users spend the most of the time consulting web portals in the pre–trip phase. The possibility of obtaining social media data and providing user-generated content are powerful tools for help users in the decision process. In this work, we present our framework for profiling both single users and group of users that relies on a not intrusive analysis of the users' behaviors on social networks/media. Moreover, the analysis of the behavior of small close groups on social networks may help an automatic system in the merge of the different preferences the users may have, simulating somehow a decision process similar to a natural interaction. Such data can be used to provide POI filtering techniques on city touristic portals.

1 Introduction

Decision–making and information searching to plan a touristic trip are difficult processes [1]. Information technology has shaped the way in which travel–related information are founded [2], since online content is now a primary source of travel information. Almost all processing and decision-making models include pre-visit hunt for information as one of the key components. When tourists plan their vacation, they look for transports, accommodations, cultural sites, restaurants, events and so on. In most cases, they have to refer to several web-applications, at least one for each service, or to specialized destination travel service portals and websites (such as www.Tripadvisor.com). These online services organize and present travel information on various topics, including flights, hotels, attractions, linking their data to customer reviews.

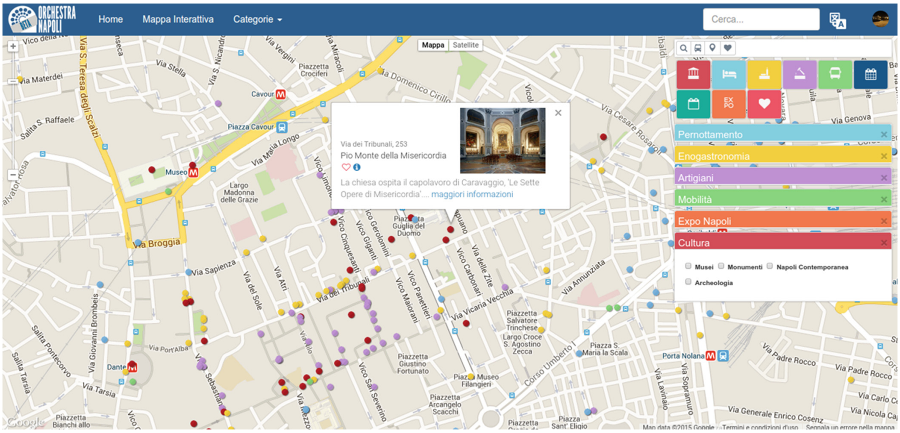
In this context, we would like to provide a unified window to the city of Naples which gathers all information and services, and shows them on a map

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 297–317, 2016.
DOI: 10.1007/978-3-319-30996-5_15

as POI, in order to support tourists in travel organization process. Since the number of the available POI is high and since many tourists visit a city only for few days, it is not possible to visit and evaluate every POI: the tourist has to make a selection of what he/she believes to be the most valuable POI. Social media and user-generated content provide powerful tools for help users in the decision process. However, their possible impact on the decision making process depends on trust [3].

In this work, we describe a general framework that relies on the automatic analysis of both single user behaviors and group relationships, using the same social network, in order to provide a POI filtering technique that can work for both. In detail, the developed framework is based on an automatic user profiling system that, without intruding the users with questionnaires, provides recommendations and decision support facilities for tourist users. In the proposed system, we use recommendation generated from users' profiles both to filter the POI to visualize and in order to help the user in the creation of a personalized itinerary. However, in this domain, it is difficult to extract explicit signals from the users about their interests. Typically tourists interact with the system only in preparation (or during) the trip, while user profiling techniques depend on in-depth analysis of users' traveling behavior and preferences. In the proposed system, we chose to use social networks as external sources for constructing user profiles on the basis of detailed observations of users' interaction on the social network. Recent studies have shown that, by using data drawn from social networks, it is possible to improve the quality of a recommendation system [4,5] while obtaining useful indirect information to profile occasional users. We address the cold-start problem, to properly evaluate the similarity between users, by shifting such evaluation on the domain of the social network.

Moreover, one of the main features in the planning of a city tour is the simultaneous presence of multiple users, usually aggregated in small groups (e.g., families or groups of friends), each with her/his own preferences and inclinations, which rarely want to separate or isolate themselves during the journey. In touristic application domains, group profiles have been taken into account [6], however mainly as an optimization problem among POI. Moreover, in [7] intra-group relationships, such as children and the disabled were contemplated, while [8,9] provided mechanisms to help groups in deciding common attributes and features for their holidays. Approaches that deal with small groups within museums focus on content personalization and on the possibility to enhance the group interaction during and after the visit [10], and assume a free navigation of each user within the museum space. On the contrary, outdoor planning of a city tour has to take into account that the group (not a single tourist) jointly selects the activities to perform together in order to maximize the group satisfaction. The same automatic analysis of the user behaviors on social networks can be used to evaluate social relationships among users in a group that can help in the creation of group recommendations. Here, we describe how to obtain an automatic analysis of group relationships using the same social network to provide a POI filtering technique that can work also for groups. In particular,



Via dei Tribunali, 253
Pio Monte della Misericordia
La chiesa ospita il capolavoro di Caravaggio, 'Le Sette Opere di Misericordia'.... maggiori informazioni

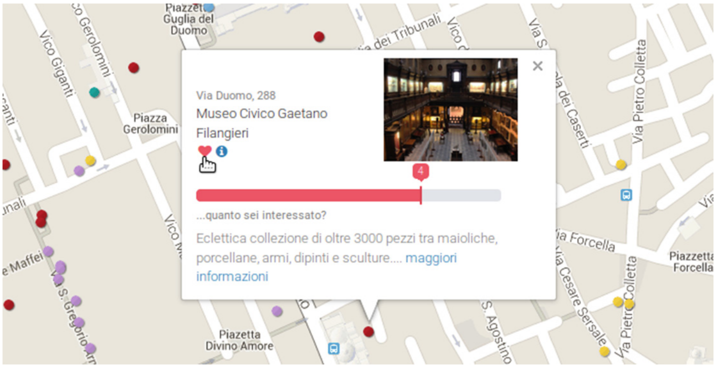
Fig. 1. Map view of the web portal.

we are interested in the analysis of the behavior of small close groups (as a representation of people that spend vacation time together) and in the definition of an automatically obtained measure of dominance. This analysis may help an automatic system in the merge of the different preferences the users may have, simulating somehow a decision process similar to a natural interaction. The use of this common framework to address both problems (single users and groups), up to our knowledge, was never addressed before in literature.

2 A Web Portal for Smart Tourism

A Touristic Web Portal can be considered a window on a city for tourists and citizens. It gathers the references to different kinds of information about the city: touristic places, restaurants, accommodations, local transports (buses, taxi, car sharing, bike sharing, etc.), events, thematic layers (like the map of movies scenes filmed in the city or the map of the best dishes) and so on. Indeed, the main goal of our developed portal is to provide all the information that the user needs in a single interface. The user can show all these POI on a map and she/he can select the POI she/he prefers.

By providing all the necessary information in one single portal, the user is free to decide her/his preferred items/activities without the need of consulting different information sources. However, this kind of interface introduces the typical information overload problem: too much information to show and to manage. Hence, we introduced two approaches to facilitate user navigation inside the Web Portal: the common possibility of browsing POI through categories and sub-category, and an automatic filtering based on the user profile. With the first approach, the user can filter every kind of POI by selecting a category and applying filters (e.g., she/he can show all three stars hotels), and then save all the POI she/he prefers in a favorites list. Preferred POI are shown on a predefined



Via Duomo, 288

Museo Civico Gaetano Filangieri

4

...quanto sei interessato?

Eclettica collezione di oltre 3000 pezzi tra maioliche, porcellane, armi, dipinti e sculture.... maggiori informazioni

Fig. 2. Balloon with add to favorites button.

layer called “Favorites/Recommended”. Simultaneously, by applying a filtering approach, all the information is automatically ordered and filtered according to the user profile (see Sect. 3): if the user do not use categories, not all the POI are shown (since they may be hundreds), but only those that are appropriate for that user profile. Hence, it is very important, for a smart tourist system, to include an automatic user profiling mechanism that, without intruding the user with questionnaires, learns her/his preferences and uses a Recommendation System to provide recommendations for the selection of preferred POI, the creation of a personalized itinerary, or simply to facilitate the navigation among the information. Obviously, during the city tour, the tourist is able to consult, any time, all the information contained on the portal and her/his preferred ones through her/his smart-phone or computer. In Fig. 1, the map view of portal is showed.

2.1 Pre-trip Ratings

Our Web Portal is designed to efficiently provide users with useful tools that can help them planning their trips in the city. Hence, the most of these tools concerns operations to be performed before the trip. Like in several trip–planner web applications[1], logged users can bookmark their favorites POI. This feature allows the users to retrieve quickly the POI she/he is interested in and to manage them. The user can add a POI to favorites simply clicking on the heart symbol situated into the balloon that appears clicking on the POI marker on the map or into the POI detailed card, as showed in Fig. 2. Unlike traditional approach, in our system, when the user bookmarks a POI, she/he can indicate also an interest value with a pre–trip rating between 1 and 5. It is necessary to underline that this rating is not a review, but it represents the POI importance in a potential trip. Recall that our users spend most of their time interacting with the portal before the actual trip, while they eventually provide post–visit rating on other

[1] www.tripomatic.com, www.gogobot.com, www.stay.com.

social media. The favorites POI and relative pre–trip ratings, evaluated by other users, will be used in the recommendation mechanism to build the set of POI to recommend to a user (see Sect. 3).

3 Single User Filtering

Generally speaking, the aim of a Recommendation System (RS) is to predict the relevance and the importance of items that the user never evaluated. A RS can be used both to proactively propose new items to the user, and to filter irrelevant items on a list, in order to only show the items considered the more interesting for the user (e.g., to select the k-best items, as in our case). In fact, in our system, we use recommendation both to filter the POI to visualize and in order to help the user in the creation of a personalized itinerary.

In formal terms, given a user u_i and a set of m POI $P = \{p_1, \ldots, p_m\}$, the recommendation system, for each user i, aims at building a *Preference Profile* or a ranking R_i of the user i over P. Such preference profile is the set $R_i = \{r_{i,1}, \ldots, r_{i,m}\}$, with $r_{i,x} \in \mathbb{R}$, which represents a partial order over P. Our goal is not to guess the exact value of $r_{i,j}$ the user i would assign to the item j, but to properly select the k-best items in the preference profile (the ones with the highest rating). The set P is finite and constitutes all the possible items to recommend within a spatial region and for a specific class of objects (e.g., tourist POI, restaurants, recreational activities and so on), and it does not depend on a specific user.

The most common approach used in RSs to generate a user preference profile is based on Collaborative Filtering techniques [11]. This approach suggests items to the user (or defines a rating for an item) by taking into account the preferences of similar users; this similarity is evaluated by considering the common items that they rated. However, this kind of technique suffers from two problems: $cold - start$ and *sparsity*. The cold-start problem concerns the issue that the system, at the beginning, has not yet sufficient information about a user, because she/he rated too few items; so, it cannot properly evaluate the similarity between users. The sparsity problem regards especially systems where the set of items is extremely large. In fact, in this case, most of the users only rated a small subset of the overall. Many studies dealt with these two problems: for example, in [12,13] the Authors propose some approaches to alleviate the sparsity problem, while in [14,15] the Authors suggest methods to solve the cold-start problem.

In our system, like in [16], we choose to use social networks as external sources to obtain users' information and to overcome the above–mentioned problems. In detail, we use the most popular social networks: www.Facebook.com, which is an online social network with 1.317 billion monthly active users and that stores more than 300 petabytes of user data. Recent studies, [4,5], have shown that, by using data drawn from social networks, it is possible to improve the quality of a RS. In our system, like in [16], we extract users' preferences from the contents that they published, in order to derive their preferences. The aim of this approach is to examine all cross-domain information, from a user profile, to obtain, then,

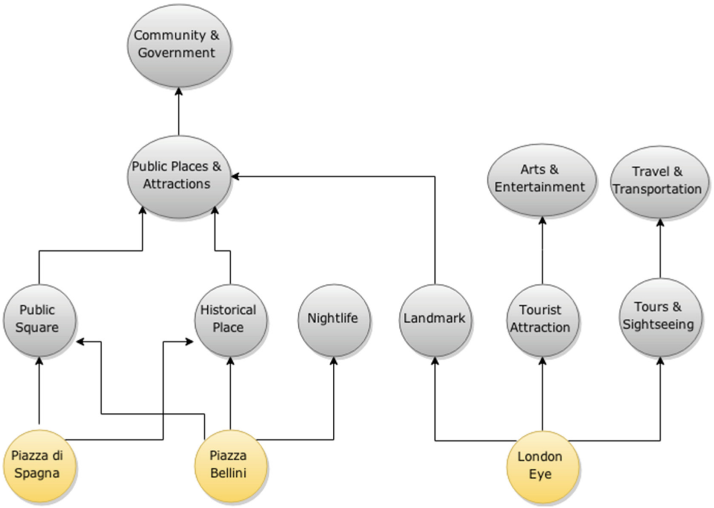
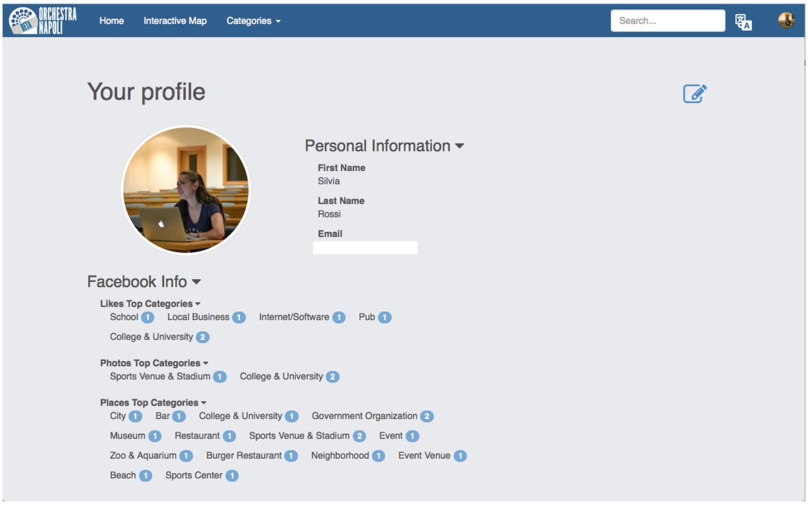


Fig. 3. An example of categories organization.

a recommendation in a specific domain (e.g., touristic preferences). Note that a typical RS approach, with a social network connection, is to gather useful information on a specific user directly from her/his peers. However, we did not choose this kind of approach also taking in account that, with the newest version of Facebook API, we cannot consider the links between all users' friends, because it is possible only to obtain the list of a person's friends which are also using the specific application and not of all of them. Instead, with this technique, we compare user preferences with all other users of the system and not with her/his personal friends.

In detail, our method, analyzing user's likes, tags, check-in and photos on www.Facebook.com, collects data from users' profile in every possible domains (age, education level, music, movies, check-in places, etc.) and uses them to evaluate the similarity between the current user and other users of the system. To evaluate such similarity, we do not consider only the specific items that are liked by user (e.g. the Rolling Stones' page or a check-in at Colosseum), but we evaluate their category (e.g. musician, rock band, history museum, Chinese restaurant, etc.). Figure 4 shows an example of profile data. Indeed, the rate of a like on an item is propagated to its parent category and then to all its hierarchy. In this procedure, like in [17], we use the logarithm to lessen the rate propagated to the parents. This kind of approach is essential because of the sparsity of the possible items, and so, we analyze the user's generic cross domain categories preferences to evaluate the user similarity. To evaluate this kind of similarity, we use an approach similar to [17], where authors propose a user similarity calculation based on a location category hierarchy extracted from the social network *Foursquare*. In our case, we build a category hierarchy graph that reproduces the hierarchy of categories of Facebook items in all kinds of domains (pages that



Your profile

Personal Information

First Name
Silvia

Last Name
Rossi

Email

Facebook Info

Likes Top Categories
School 1 Local Business 1 Internet/Software 1 Pub 1
College & University 2

Photos Top Categories
Sports Venue & Stadium 1 College & University 2

Places Top Categories
City 1 Bar 1 College & University 1 Government Organization 2
Museum 1 Restaurant 1 Sports Venue & Stadium 2 Event 1
Zoo & Aquarium 1 Burger Restaurant 1 Neighborhood 1 Event Venue 1
Beach 1 Sports Center 1

Fig. 4. An example of user profile.

user likes, locations, artists, movies, etc.); therefore the obtained graph consists of two kinds of nodes: specific nodes (that represent unique and specific items) and category nodes (that represent the specific categories of items or categories in a generic level of the hierarchy). An example of facebook categories organization in showed in Fig. 3. Like in [17], we first calculate a score on specific nodes, but whereas in [17] the authors use the number of visit on a location, in our case a score of a node can represents both a like on a page and a check-in in a specific place. Later we propagate the score from the specific nodes to category nodes using a propagation rate and then calculate the similarity like in [17]. An example of a user profile is shown in Fig. 4 with general categories evaluations.

Finally, the prediction of the preference of items in our specific domain (cultural sites and other POI of the city) is obtained using the explicit pre-trip ratings or saved itineraries produced on the Web Portal by the most similar users. In detail, the system use the ratings the favorites POI of the k-neighborhood (obtained with the procedure explained in the previous paragraph) to calculate a subset of n POI to suggest to the current user.

4 Group Filtering

In the previous section, we described how the proposed system provides recommendations for a single user, retrieving information about her/his interests from the Online Social Network (OSN) *Facebook.com* and using them to determine POI that can be of interest for the user. However, people usually organize travel in groups, and the group's members jointly select the activities to perform and the POI to visit on the basis of their personal preferences and the needs of each group's member. Hence, our system must provide support to this group decision making process, by implementing a group recommendation system.

The problem of providing recommendation to groups has been widely analyzed in recent years. The diversity and dynamics of inter-group relationships make it a very challenging problem [18], and it is widely recognized that one of the main issues to take into account in the design and implementation of these systems is the type of control over the group decision-making process [19]. Hence, recommendation systems for groups need to capture both preferences of the group members but also key factors in the group decision process [18]. For example, in some cases, the group's members may find an agreement following a democratic process, but, in the most cases, group's members have different influences on the others, and there are key persons, a human leader for example, that have more influence in the final decision. Real small group interactions take into account intra-group roles and influence hierarchies, and the implemented system must take into account these social dynamics.

Generally, there are two possible approaches used to design a group recommendation system. The first uses the users' profiles (one for each of the group's member) and it merges them in order to obtain a single profile for the whole group. Then, it uses a single user recommendation system on this profile to find the recommendations for the group. The second approach firstly uses a single user recommendation system on each user's profile, determining recommendations for all group's members, and then it merges these recommendations using some group decision strategy. For our specific context we need to have the maximum flexibility in the group formation, and the identity of the group members has to be dynamically determined since the actual members of a group can be established only according to the activity to perform. For these motivations, we decide to use the second approach. In this way, single user's profiles and recommendations are built independently from the group membership. This allows the system to dynamically account for group relationships at the time of providing the group recommendations because the users' recommendations are merged only once the group is formed. Besides, during the process of aggregation of user's preferences, we can estimate the importance of each user with respect to the other group's members and determine a sort of dominance value for each user. This value is then used as weight in the aggregation process.

In the following sections, the evaluation of such dominance value is detailed (see Sect. 3.2), and subsequently the aggregation functions used by the system is defined (see Sect. 4.3).

4.1 Social Strength Analysis in Online Social Networks

According to social scientists [20–22], social strength, or tie strength, can generally be said to be a metaphor that quantifies relationships between people. Peter et al., addressed the problem of measuring social strength by using multiple dimensions such as closeness and duration [20]. [23] defined seven dimensions for predicting social strength: intensity, duration, intimacy, reciprocal services, structural, emotional support, and social distances. These seven dimensions have been applied for predicting relationship tiers as being either strong or weak, mainly by using manual efforts.

To understand users' relationships and roles inside a group, the analysis of interactions in OSNs among the group's members can be used. In detail, this kind of analysis can be considered a useful way to obtain (without intruding the users with questionnaires, but simply observing their communication habits and frequency) information about these social relationships and activities among the group of visitors that can be used in helping to take decisions. The attempt to infer meaningful relationships from social networks connectivity is often criticized from sociology researchers [24]; however, analysis of the interaction graphs in controlled situations (small and close groups) may provide useful insight.

The analysis of relationship through social networks is a complex activity that requires a deep analysis of the individual profiles and the types of interaction between members of a group. Social Network Analysis evaluates the relationships and flows between people, organizations, groups, etc., organized in graphs. In the most cases, these entities are mapped into the nodes of a graph in which the edges show relationships. By analyzing these graphs it is possible to identify the location of actors and extract the various groupings and roles. Many mathematical techniques, inherited from graph theory, are available to evaluate this kinds of networks. The most common approaches involve a cluster computation, with the identification of the dominant central cluster and the periphery clusters, and the classification of the different kinds of nodes (hubs, bridges, isolates, etc.). Several centrality measures exist in literature, the most recurring are those formalized in [25], that are degree centrality, closeness centrality and betweenness centrality. However, the basic definitions of these measures are only designed for binary network and are based on unweighted and undirected graphs. Hence, many social networks analysis approaches assume binary and symmetric relationships of equal value between all directly connected users, while, in reality, an individual has relationships of varying quality [26].

In order to provide effective group recommendation on our web portal, we evaluate not only the strength, but also the "direction" of a specific relationship, defining a "function" that does not use semantic textual features. Our aim is to use the strength of such directional ties to define a measure of dominance/popularity for each member of the group that could be used as a weight of each user in the decision process. Moreover, social networks analysis may lead to a misinterpretation on popularity as dominance that sometimes are high correlated, but sometimes they are not. It was shown that cohesiveness of a group determines the correlation between these two concepts [27]. Hence, the cohesiveness of a group is a requirement for providing help in the decision process. In a close group, users' self-needs can be sacrificed for the wellness of the whole group.

4.2 A PageRank–Based Evaluation for Weighted Networks

There are a number of attempts to generalize the node centrality measures to weighted networks. For example, [28] maps a weighted network to an unweighted multigraph and adapts standard techniques for unweighted graph to these multigraph. [29], instead, proposes a generalization that combines tie weights and number of ties, considering also the case of direct networks.

Here, to compute the users' centrality, we use a variation of the famous *PageRank* algorithm [30], used by the Authors to rank web pages, firstly introduced in [31]. We followed this choice for creating a simple, but effective, algorithm, with the aim of evaluating the rank of a person, interpreted as its indirect rank in a group of people, and of obtaining a value that can be considered an index of popularity in a small group of friends. It should be recalled that the two concepts of popularity and dominance are correlated in small and close groups [27]. A similar approach was used in [32], where the authors use a modified version of PageRank to define a new centrality measure. While the original PageRank formula of Brin and Page is based on directed and unweighted graphs, the version proposed in [32] is adapted for the undirected and weighted graphs. Instead, in this work, we present another variant that uses directed and weighted graphs. In our opinion, both the degree of activity of a person and the direction of specific communication activities are essential to obtain information about the social relationships among members of a group.

Our ranking function is defined as follows:

$$R(x) = \frac{1-d}{|F|} + d\sum_{i \in F} \frac{w(i,x)}{w(i)} R(i) \tag{1}$$

where, $|F|$ is the total number of friends in the group and d (with $0 \leq d \leq 1$) is a dampening factor set to 0.85 (this value is often considered the default value for PageRank calculations [33]). In the second part of Eq. 1, the user x inherits a portion of popularity from the other i group's members. In detail, this proportion is calculated by considering both the i-th friend's popularity and the weight of the communication activity of the i-th friend towards the user x ($w(i,x)$), normalized with respect to the total communication activity of the i-th friend with all the members of the group ($w(i)$). The rationale of this choice is that the frequency of directed communication (or interaction) from the user i towards the user x is an index of the strengths of the directed tie i-x (which can have a different value with respect to the tie x-i, and, hence, have a different impact on the evaluation of the x's popularity within the group).

Such weights are calculated by considering some of the communication activities between couple of users on the OSN *Facebook.com*, collecting a combination of data arising from [23]. Referring to the activity graph of friends' relationship, $w(i,x)$ evaluate the edges from the user i to the user x, which represent an activity with i as source and x as receiver. In detail, regarding the ONS facebook.com, the considered activities are:

- 1 basic activity derived from the existence of the friend's relationship between i and x;
- $\#F(i,x)$ is the number of feeds (posts and links) published on the wall of the user x by the user i;

- $\#F_c(i,x)$ is the number of comments from the user i on feeds published by the user x;
- $\#F_l(i,x)$ is the number of likes from the user i on the posts published by the user x;
- $\#F_t(i,x)$ is the number of tags of user x inserted by i;
- $\#P_c(i,x)$ is the number of comments from the user i on photos published by the user x;
- $\#P_l(i,x)$ is the number of likes from the user i on photos published by the user x;
- $\#P_t(i,x)$ is the number of tags of user x inserted by i on photos.

Hence,

$$w(i,x) = 1 + \#F(i,x) + \#F_c(i,x) + \#F_l(i,x) + \\ + \#F_t(i,x) + \#P_c(i,x) + \#P_l(i,x) + \#P_t(i,x) \tag{2}$$

The obtained $w(i,x)$ value is normalized with $w(i)$, that can be calculated with the same type of data of the user i, but with respect to the relationships with all users of the group and not only with the user x:

$$w(i) = \sum_{j \in F} w(i,j) \tag{3}$$

Note that the friend's contribution is normalized with respect to its global activity on the whole group (as in PageRank). However, PageRank assumed that there is only a single link between two pages x and i, hence, web page i contributes equally to the centrality of all web pages it points to, while, here, we represent the weight of the directed connection from i to x determining the level of one-side communication.

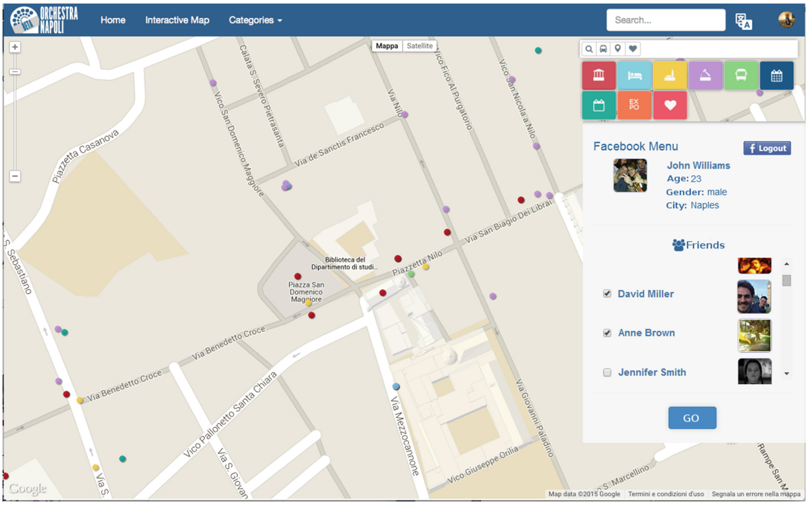
Like the classic PageRank, the Eq. 1 iterates until the values will converge.

4.3 An Aggregation Mechanism for Group Recommendation

Figure 5 shows the portal section that support users in the selection of the group. Initially, when user connects to the portal, her/his profile is used to show, on the map, POI that can interest her/him. Furthermore, the user can select a set of friends, and the system uses the Group Recommendation function to suggest POI for the whole group.

As stated above, the dominance measure, as defined in Eq. 3, can be used as a weight in the process of merging single user's recommendations. In this way, we give an importance to the recommendation of a user proportional to her/his influence/dominance on the others in the group.

Figure 6 shows the architecture of our recommendation system; single users' profiles are used to obtain the single recommendations, and the information about the interactions on the social network are used to compute the Popularity (Dominance) rankings. Both these information are used from the Group Recommendation System to provide the final choices for the whole group.



John Williams
Age: 23
Gender: male
City: Naples

☑ David Miller
☑ Anne Brown
☐ Jennifer Smith

Fig. 5. Interface for group recommendation.

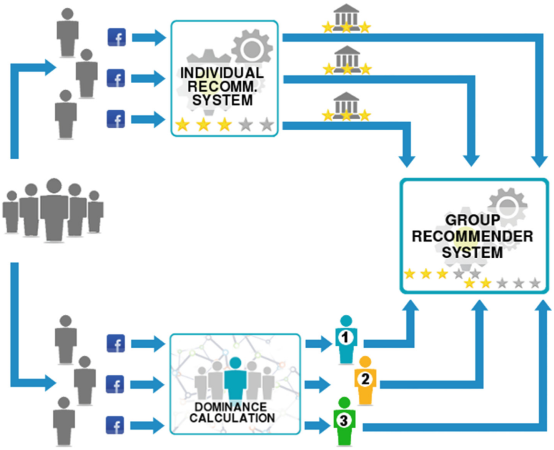


Fig. 6. System architecture for single user and group recommendation.

To evaluate the group $r_F(x)$ rating for the POI x we use the following strategy, introduced in [34]:

$$r_{avg,x} = \frac{1}{n}\sum_{i=1}^{n}(R(i) \cdot r_{i,x}) \tag{4}$$

where, n is the number of users in the group, $R(i)$ is the dominance value of user i, calculated as defined in Eq. 1. Hence, Eq. 4 is a function that evaluates the average of all the i users rankings $r_{i,x}$ of the item x, weighted by the i-th dominance value $R(i)$.

The set $\succ_{avg} = \{r_{avg,1}, \ldots, r_{avg,m}\}$, which is the set of group's rankings computed for each item, is then used to get the final decision: the first k activities x (with k equals to the number of activities to propose) with the highest $r_{avg,x}$ values are selected for the recommendation.

5 A Pilot Study

We conducted two types of pilot studies with real users, one on single users to evaluate the use of pre–visit rating mechanism, and the other on users group to evaluate the preferences aggregation process. Each participant was asked to plan a trip in the city of Naples, choosing the POIs they were interested to visit. All the useful information were gathered in order firstly to compare social network relationships vs. face-to-face interactions, and to evaluate the usefulness of the mechanism of saving the points of interest.

5.1 A Single User Study

Actors. We have analyzed the answers of 32 people (18 male and 14 female), aged between 21–40 (with an average age of 29), interacting with the Portal. Most of the participant declared in their Facebook profile to have a high education level, 20 people (62.5 % of 32 users) have a degree, 3 a doctoral degree (Ph.D.), and 2 an high-school degree; the remaining 7 users did not make any statement.

Procedure. We asked the participants to image to plan a trip in the city of Naples, selecting the points of interest (cultural, gastronomic) they intend to visit. Users were instructed to log in to the Portal using their Facebook account and to surf the categories and the interactive map in order to choose the preferred POIs. They could freely choose which and how many points of interest they liked, adding them as their favorites and given them an evaluation of liking. When they were satisfied with the planning, we asked them to fill in a questionnaire valuating the utility of the pre–visit rating mechanism.

We proposed to the participants a questionnaire composed by fifteen questions. We asked them to answer to the questions, giving at each one an evaluation of agreement in a range between zero and five (5=Great, 4=Good, 3=Average, 2=A little, 1=Not at all, 0=Not applicable), and eventually to leave a comment. We want to know if our mechanism is intuitive, easily to learn and meets people's expectations, if the task of planning a trip saving the favorites is easily achieved, and how much it can be helpful or even needful for organizing a holiday.

Results. Table 1 reports the results in terms of minimum, maximum and average values for each question.

Considering the average values, we can notice the users were quite satisfied of our mechanism according each category investigated by us (bold text), meeting a medium evaluation of 4.11 (more than 4=Good).

Table 1. Users evaluations of the mechanism for saving points of interest.

Question	Min	Max	Average
Learnability of the Interaction			
"I quickly learnt to save my favorites."	3	5	4.56
"I easily saved my favorites."	3	5	4.68
"I thought help messages were very helpful."	0	5	3.09
Easy of Use			
"I easily use it."	2	5	4.15
"I needed few steps to save my favorites."	2	5	4.53
"I believe most people can quickly learn to save their favorites."	2	5	4.09
Utility			
"I believe it is very useful to plan my trip activities."	0	5	4.15
"I can better manage my trip activities."	2	5	3.84
"I can easier reach my goals using it."	1	5	3.90
"I can save time using it."	1	5	3.93
Satisfaction			
"I found my expectations were met."	2	5	3.84
"I am satisfied."	3	5	4.34
"I will recommend it to friends."	3	5	4.40
"I would like to use it often."	2	5	4.03
"I like to use it."	2	5	4.21

Since the pre–visit rating mechanism is not common in web portal as well as in travel social media, we were particularly interested in the answers given about the utility of the process. The average values of these evaluations were a little lower than the other categories. However, we can notice that our proposed mechanism found an appreciation greater than the average value in the majority of the cases, but with a greater standard deviation (minimum values are lower in this category). We foresee that these lower results were achieved because the testers were involved only in saving their preferences without evaluating the complete recommendation system. The complete evaluation will be conducted once a sufficient number of interaction data will be available.

The participants left few comments. Only one user expressed their preference for a more traditional and textbook management of they travel activities. Everyone else expressed their opinions about the graphical details.

5.2 A Group Study

Actors. In this study, we involved 14 groups composed, in the average, of 3.4 people. The number of the total users that took part in the experimentation was 46 (26 male and 20 female). The average age was 27.3 with a graduate education. During the recruitment process, in the half of the groups, all the members of each group were directly contacted by us and involved in the experiment; in the other cases, we asked a single person to create a group and to explain the rules of the experiments to the other members. Hence, in this second case, this specific person acted as a mediator in the recruitment process. Users were ranked, within a group, according to their respective dominance values according to Eq. 1. All the analyzed data (feeds, photos, comments, tags and likes) from *facebook.com* are summarized in Table 1, where we reported the total number of analyzed data and the average value for each group (Table 2).

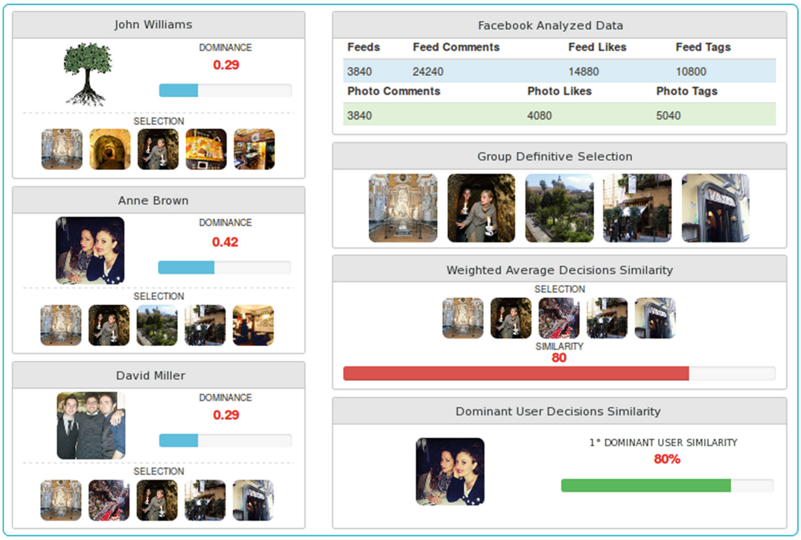
Table 2. Facebook analyzed data.

	Feeds	F. Comm	F. Likes	F. Tags	Photo Comm	Photo Likes	Photo Tags
Total	414720	391320	763440	266040	343800	639600	955680
Average	29623	27951	54531	19003	24557	45686	68263

Procedure. Each person was asked to register on the website using her/his own credentials; once registered, they were asked to imagine to plan a one-day visit to the city. In detail, they were asked to select from ten items, shown on our website, only three activities (e.g., places to visit) for the day, and one restaurant for lunch and one for dinner (from a check list of eight). Since we do not want the user to be involved in strategic reasoning, we did not ask the users to express ratings and preferences among the selected choices. The group was, then, asked to discuss, face-to-face, in order to obtain a shared and unique decision for the entire group (which represents the groups' ground truth r_{GT}).

Results. Figure 7 shows a summary of the results of a single experiment; in detail, we reported the single users' selections, the analyzed facebook data number, the selection obtained from the group members' discussion r_{GT}, the similarity evaluation between the user with the higher dominance value (dominant user) and r_{GT}, and the similarity between the decision obtained using the weighted average function and r_{GT}. In both cases, the similarity is calculated simply counting the number of common choices between the two selections. In detail, in the experiment reported in Fig. 7, we have a 3 people group with an 80 % of similarity with the dominant user, which is the user with $R(i) = 0.42$.

Table 3 summarizes the cumulative data of all groups involved in the experiment. For each group, the following data are calculated: the similarity percentage between the choices of the dominant and r_{GT} (*Dominant*); the similarity percentage between the choices of the mediator (if applicable) and r_{GT} (*Mediator*);



John Williams

DOMINANCE

0.29

SELECTION

Anne Brown

DOMINANCE

0.42

SELECTION

David Miller

DOMINANCE

0.29

SELECTION

Facebook Analyzed Data

Feeds	Feed Comments	Feed Likes	Feed Tags
3840	24240	14880	10800

Photo Comments	Photo Likes	Photo Tags
3840	4080	5040

Group Definitive Selection

Weighted Average Decisions Similarity

SELECTION

SIMILARITY

80

Dominant User Decisions Similarity

1° DOMINANT USER SIMILARITY

80%

Fig. 7. A screen-shot of the results of the experiment with one group.

Table 3. Cumulative results in the pilot study.

% Similarity	Dominant	Average	Mediator
Average	61 ± 17	59 ± 11	63 ± 13

Table 4. Results with and without mediation.

% Sim	with Med	without Med	Low STD
Avg	53 ± 15	73 ± 10	75 ± 10

the similarities average percentage of the choices of each users in a group and r_{GT} (*Average*).

From the amount of analyzed interactions, with a very high standard deviation, we can conclude that the groups' behaviors on the OSN were very different and with a good value of cohesion ($Average = 59\,\%$). Considering the aggregated data, the average similarity value of the dominant user choices (*Dominant*) is on average 61 %, which is comparable with the *Average* similarity, and the *Mediator* similarity (63 %) with the final decision of the Group r_{GT}.

Apart from the aggregated data that shows similar results on the average, what is interesting, from our point of view, is to compare the behavior of groups with a mediator with groups without this specific role. Table 4 summarizes the results of this analysis. We observed that in the case of a member of the group acting as mediator the similarity of the group decision w.r.t. the dominant user was on average 53 % (*with Mediator*); instead, in the second case, the similarity

Table 5. Similairty results with and withoud dominance weights.

% Similarity	$r_{st.avg}$	r_{avg}
Average	64 ± 16	74 ± 12

with the dominant user was, on average, equal to 73 % (*without Mediator*). In our opinion these values support our choice to use a ranking function (as defined in Eq. 1) to differently weight the most dominant users in the group consensus functions. The p-value, calculated on these two sets, is 0.0058, which means that such difference is not due to the case (Table 5).

Finally, we analyzed the standard deviation of the dominance values (according to Eq. 1) and subdivided the groups without a mediator in two sets (with low and high standard deviation). Surprisingly, the groups with low standard deviation, which can also be interpreted as a measure of cohesion and similarity in the behaviors of the group members on the social network, showed a similarity of the dominant user choices with the group final decision of 75 % (*with Low STD*). However, what we want to highlight is that it is not the dominance value *per se* to be of importance in the group decision making process (recall that such values are normalized in order to sum to one), but the relative user ordering. Moreover, the case of users with approximately the same behavior on the social network (e.g., with similar dominance values), in accordance with [27], better identify close group in which popularity is connected with dominance. Hence, we can infer that, in case there is not a mediator, the dominance evaluation got a much more important role in the consensus making, especially in close groups where the popularity index, we evaluated, better identifies a possible dominant user.

Finally, the similarity of the proposed weighted version of the average satisfaction function (r_{avg}) with respect to the groups' ground truth (r_{GT}) was evaluated. Such similarity is computed as a percentage of the r_{avg} choices that were already selected in the group final choices r_{GT}. We also evaluated the similarity of the groups' ground truth with respect to the standard implementation of such function ($r_{st.avg,x}$) on the same data, defined as follow:

$$r_{st.avg,x} = \frac{1}{n}\sum_{i=1}^{n} r_{i,x} \tag{5}$$

With respect to their standard implementation, the function that takes into account social relationships perform slightly better (74 % w.r.t. 64 %). The r_{avg} consensus function often guesses 4 on 5 activities. The difference among the obtained results was evaluated as statistically significant using a t-test ($p < .05$, $t = 3.6$, $df = 13$).

6 Conclusion and Future Works

In this paper, we describe a general framework for user profiling/recommendation that relies on the automatic analysis of both a single user behavior and his/her intra-group relationships on online social networks, hence, without intruding the users with questionnaires. We will use the presented recommendation mechanism in a web portal for smart tourism, with the aim to help users' decision-making process in the organization of their touristic tours.

Touristic web portals show unified windows to cities, which gather all information and services, showing them on a map as POI. However, since the interactions of a user with the portal can be few, and most of them in the pre–trip information haunting phase, we gather information about his/her preferences and behavior from the social network facebook.com, and we use this information to evaluate both similarities with other users on a cross domain context, and his/her social value within a small group of users. Then, we decided to introduce the possibility to express pre-trip interest ratings on POI, and to use the ones, as well as the saved itineraries, produced on the Web Portal by the most similar users, in a Collaborative Filtering recommendation mechanism, to generate possible POI ratings for the current user and to build the set of POI to recommend. Moreover, we showed that it is possible to derive a simple measure of a user dominance within a group through an intra-group ranking mechanism. Such value is obtained from the analysis of the interaction on the same social network, and we started using this measure of user's dominance for weighting users' preferences in an aggregation function, in order to provide recommendations to groups of users. Our long-term goal is to use this measure of users dominance in the definition of more different and customizable aggregation functions.

Finally, we presented two pilot studies, one for evaluating the utility of the pre-visit rating mechanism, and a second to evaluate the preferences aggregation process and the importance of the role of dominant user's in the group decision-making process. In the first study, we asked the participants to image to plan a trip in the city of Naples using the web portal, and then to fill in a questionnaire evaluating the use of the pre-visit rating mechanism. Results show a good user satisfaction with respect to this mechanism. However, since the system does not provide yet the final recommendations, the utility of the pre–trip ratings mechanism was not totally understood. In the second study, we use a small number of alternatives for planning only a single day group activities in a delimited neighborhood of a city. The study shows that the figure of the dominant user in the group's choice is fundamental in the absence of a mediator. Moreover, the study shows that the use of our weighted social choice function can increase the performance of the group recommendation system with respect to a standard not-weighted version of the same aggregation function.

Nevertheless, in both cases, we have to further expand our evaluation. For the single user case study, for example, we have to conduct a complete evaluation of the whole single user recommendation system, once the available number of user interactions with the system will increase. For the group case, the scalability of our results in providing recommendation to groups have to be evaluated by increasing the number of available choices (i.e., the ones provided by the individual user recommender system), and including also the possibility to express the pre–visit ratings on the selected choices. Finally, we limited our groups to people that did not have any hierarchical relationships among them (e.g., they were mainly friends), while also social intra-group roles have to be taken into account.

Acknowledgement. The research leading to these results has received funding from the Italian Ministry of University and Research and EU under the PON OR.C.HE.S.T.R.A. project (ORganization of Cultural HEritage for Smart Tourism and Real-time Accessibility).

References

1. Fodness, D., Murray, B.: Tourist information search. Ann. Tourism Res. **24**, 503–523 (1997)
2. Beldona, S.: Cohort analysis of online travel information search behavior: 1995–2000. J. Travel Res. **44**, 135–142 (2005)
3. Wang, Y., Chan, S.C.F., Ngai, G., Leong, H.-V.: Quantifying reviewer credibility in online tourism. In: Decker, H., Lhotská, L., Link, S., Basl, J., Tjoa, A.M. (eds.) DEXA 2013, Part I. LNCS, vol. 8055, pp. 381–395. Springer, Heidelberg (2013)
4. Guy, I., Zwerdling, N., Carmel, D., Ronen, I., Uziel, E., Yogev, S., Ofek-Koifman, S.: Personalized recommendation of social software items based on social relations. In: Proceedings of the third ACM conference on Recommender systems, pp. 53–60. ACM (2009)
5. Said, A., De Luca, E.W., Albayrak, S.: How social relationships affect user similarities. In: Proceedings of the 2010 Workshop on Social Recommender Systems, pp. 1–4 (2010)
6. Souffriau, W., Vansteenwegen, P.: Tourist trip planning functionalities: State–of–the–art and future. In: Daniel, F., Facca, F.M. (eds.) ICWE 2010. LNCS, vol. 6385, pp. 474–485. Springer, Heidelberg (2010)
7. Ardissono, L., Goy, A., Petrone, G., Segnan, M., Torasso, P.: Intrigue: Personalized recommendation of tourist attractions for desktop and handset devices. Appl. Artif. Intell. **17**, 687–714 (2003)
8. McCarthy, K., McGinty, L., Smyth, B., Salamó, M.: The needs of the many: A case-based group recommender system. In: Roth-Berghofer, T.R., Göker, M.H., Güvenir, H.A. (eds.) ECCBR 2006. LNCS (LNAI), vol. 4106, pp. 196–210. Springer, Heidelberg (2006)
9. Jameson, A.: More than the sum of its members: Challenges for group recommender systems. In: Proceedings of the Working Conference on Advanced Visual Interfaces. AVI 2004, pp. 48–54. ACM (2004)
10. Kuflik, T., Stock, O., Zancanaro, M., Gorfinkel, A., Jbara, S., Kats, S., Sheidin, J., Kashtan, N.: A visitor's guide in an active museum: Presentations, communications, and reflection. J. Comput. Cult. Herit. **3**, 1–25 (2011)
11. Ricci, F., Rokach, L., Shapira, B.: Introduction to recommender systems handbook. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook. Springer, Heidelberg (2011)
12. Yildirim, H., Krishnamoorthy, M.S.: A random walk method for alleviating the sparsity problem in collaborative filtering. In: Proceedings of the 2008 ACM Conference on Recommender Systems, pp. 131–138. ACM (2008)

13. Huang, C.B., Gong, S.J.: Employing rough set theory to alleviate the sparsity issue in recommender system. In: International Conference on Machine Learning and Cybernetics. vol. 3, pp. 1610–1614. IEEE (2008)
14. Sahebi, S., Cohen, W.W.: Community-based recommendations: A solution to the cold start problem. In: Workshop on Recommender Systems and the Social Web, RSWEB (2011)
15. Rashid, A.M., Karypis, G., Riedl, J.: Learning preferences of new users in recommender systems: an information theoretic approach. ACM SIGKDD Explor. Newslett. **10**, 90–100 (2008)
16. Shapira, B., Rokach, L., Freilikhman, S.: Facebook single and cross domain data for recommendation systems. User Model. User-Adap. Inter. **23**, 211–247 (2013)
17. Lee, M.-J., Chung, C.-W.: A user similarity calculation based on the location for social network services. In: Yu, J.X., Kim, M.H., Unland, R. (eds.) DASFAA 2011, Part I. LNCS, vol. 6587, pp. 38–52. Springer, Heidelberg (2011)
18. Gartrell, M., Xing, X., Lv, Q., Beach, A., Han, R., Mishra, S., Seada, K.: Enhancing group recommendation by incorporating social relationship interactions. In: Proceedings of the 16th ACM International Conference on Supporting Group Work, GROUP 2010, pp. 97–106. ACM (2010)
19. Jelassi, M.T., Foroughi, A.: Negotiation support systems: An overview of design issues and existing software. Decis. Support Syst. **5**, 167–181 (1989)
20. Marsden, P.V., Campbell, K.E.: Measuring tie strength. Soc. forces **63**, 482–501 (1984)
21. Nelson, R.E.: The strength of strong ties: Social networks and intergroup conflict in organizations. Acad. Manag. J. **32**, 377–401 (1989)
22. Granovetter, M.S.: The strength of weak ties. Am. J. Sociol. **78**, 1360–1380 (1973)
23. Gilbert, E., Karahalios, K.: Predicting tie strength with social media. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. CHI 2009, pp. 211–220. ACM, New York (2009)
24. Wilson, C., Boe, B., Sala, A., Puttaswamy, K.P., Zhao, B.Y.: User interactions in social networks and their implications. In: Proceedings of the 4th ACM European Conference on Computer Systems, EuroSys 2009, pp. 205–218. ACM (2009)
25. Freeman, L.C.: Centrality in social networks conceptual clarification. Soc. Netw. **1**, 215–239 (1979)
26. Banks, L., Wu, S.: All friends are not created equal: An interaction intensity based approach to privacy in online social networks. Int. Conf. Comput. Sci. Eng. **4**, 970–974 (2009)
27. Theodorson, G.A.: The relationship between leadership and popularity roles in small groups. Am. Sociol. Rev. **22**, 58–67 (1957)
28. Newman, M.E.J.: Analysis of weighted networks. Phys. Rev. E **70**, 056131 (2004)
29. Opsahl, T., Agneessens, F., Skvoretz, J.: Node centrality in weighted networks: Generalizing degree and shortest paths. Soc. Netw. **32**, 245–251 (2010)
30. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. In: Proceedings of the Seventh International World Wide Web Conference Computer Networks and ISDN Systems, vol. 30, pp. 107–117 (1998)
31. Caso, A., Rossi, S.: Users ranking in online social networks to support poi selection in small groups. In: Posters, Demos, Late-breaking Results and Workshop Proceedings of the 22nd Conference on User Modeling, Adaptation, and Personalization (UMAP2014). CEUR Workshop Proceedings, CEUR-WS.org, vol. 1181 (2014)

32. Heidemann, J., Klier, M., Probst, F.: Identifying key users in online social networks: A pagerank based approach. In: Proceedings of the International Conference on Information Systems, ICIS 2010, Association for Information Systems, pp. 1–22 (2010)
33. Langville, A.N., Meyer, C.D.: Deeper inside pagerank. Internet Math. **1**, 335–380 (2004)
34. Rossi, S., Caso, A., Barile, F.: Combining users and items rankings for group decision support. In: Bajo, J., et al. (eds.) Trends in Prac. Appl. of Agents, Multi-Agent Sys. and Sustainability. AISC, vol. 372, pp. 151–158. Springer, Heidelberg (2015)

A Transformation Language for RDF Based on SPARQL

Olivier Corby[1(✉)] and Catherine Faron-Zucker[2]

[1] Inria, Sophia Antipolis, France
olivier.corby@inria.fr
[2] Univ. Nice Sophia Antipolis, CNRS, I3S, Nice, France

Abstract. XSLT is a language for transforming XML documents into other XML documents. Despite its 16 years long life, the RDF Semantic Web language still waits its transformation language. Some propositions have been done, relying on and extending XSLT, but none of them became widely used. In this paper, we present a radically new transformation language for RDF, called STTL. It enables to transform RDF into RDF as well as any other text format. The originality and power of STTL is that it is based on SPARQL. We designed it as a lightweight extension to SPARQL and we compile it into standard SPARQL. We present a generic transformation rule engine implementing STTL and several RDF transformers we defined for various output languages, showing STTL's expressive power.

1 Introduction

The read-write Web is now providing us with a world-wide blackboard where a hybrid society of users and software agents exchange digital inscriptions. The RDF standard [1] provides us with a general purpose graph-oriented data model recommended by the W3C to represent and interchange data on the Web. While the potential of a world-wide Semantic Web of Linked Data and schemas is now widely recognized, the transformation and presentation of RDF data is still an open issue. Among the initiatives to answer this question there are extensive works for providing RDF with several and varied syntaxes: XML, N-Triples, Turtle, RDFa, TriG, N-Quads, JSON-LD, CSV-LD, etc. But this is still a partial view of the above problem.

Just like the structured Web has been provided with the XSLT transformation language to present XML data to the user into HTML pages or to transform XML data from one XML schema into another one or from an XML schema into any non XML specific text format, for XML data interchange between agents and therefore interoperability, the Web of data now requires a transformation language to present RDF data to users and transform RDF data from one RDF schema into another or transform data from its RDF syntax into another one. Indeed, a special case of RDF data holds a very special potential: RDF data encoding other formal languages. In computer science, formal languages have been used for instance to define programming languages, query languages,

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 318–340, 2016.
DOI: 10.1007/978-3-319-30996-5_16

data models and formats, knowledge formalisms, inference rules, etc. Among them, in the early 2000's, XML has gained the status of a meta-language or syntax enabling to define so-called XML languages. In the same way, we are now assisting to the advent of RDF that will likely be more and more used as an abstract syntax to represent other languages. For instance in the domain of the Semantic Web alone, this is the case of three W3C standards: OWL 2 [2] is provided with several syntaxes, among which the functional syntax, the Manchester syntax used in several ontology editors and RDF; the Rule Interchange Format (RIF) [3] is provided with several syntaxes among which a verbose XML syntax, two compact syntaxes for RIF-BLD and RIF-PRD and an RDF syntax; SPARQL Inference Notation (SPIN) is a W3C member submission [4] to represent SPARQL rules in RDF, to facilitate storage and maintenance. Many other languages can (and will) be "serialized" into RDF. For instance [5] is an attempt to represent SQL expressions in RDF.

As a result of this emerging trend to use RDF as a "syntax" or a meta-language for other Web languages, just like XML ten years earlier, the transformation of RDF data into various output formats, various concrete syntaxes, becomes a major issue. Regarding the RDF/XML syntax of RDF, one could think that XSLT is a good candidate to declaratively express RDF transformations. However, writing XSLT templates for RDF would be based on its XML syntax and not on the graph structure and semantics of RDF model — the transformation rules would depend on the concrete XML syntax of RDF instead of its graph structure and semantics—, which would make the writing of the transformation quite difficult. Moreover, the many potential serializations of any given RDF statement would make the writing XSLT templates for RDF even more complex.

In this paper we address the latter problem of transforming RDF data, i.e., generating the concrete syntax of expressions of a given language from their RDF representation. More generally, the research question addressed in this paper is *How to transform RDF data into other languages?* We answer two sub-questions: (1) How to write declarative transformation rules from RDF to RDF and other languages? (2) How to make the approach generic, i.e. the rule language independent from the output language?

We show how SPARQL [6] can be used as a generic transformation rule language for RDF, independent from the output languages. We define an RDF transformer as a set of transformation rules processed by a generic transformation rule engine. We present SPARQL Template Transformation Language (STTL), a lightweight extension to SPARQL enabling the writing of transformation rules.

In Sect. 2 we present existing transformation languages for RDF. In Sect. 3 we present STTL, an extension of SPARQL enabling the writing of RDF transformation rules. In Sect. 4 we present the generic transformation rule engine we developed to implement STTL. In Sect. 5 we show the expressive power of STTL through several RDF transformers we defined for various output languages.

2 Related Work

XSLT [7] is a pioneer rule-based transformation language for XML. An XSLT stylesheet is a set of transformation rules, called *templates*, which enables to transform any XML document conform to a given model, i.e., to which the templates apply. An XPath expression identifies the XML subtrees (their roots) for which a template applies, and the content of the template describes the transformation and its output. XSLT could be used to process and display RDF/XML data in any output format. For instance the following XSLT template could be used to transform RDF triples into an HTML table.

```
<xsl:template match='rdf:Description[@rdf:about]'>
  <xsl:for-each select='./*'>
    <tr> <td>
      <xsl:value-of select='../@rdf:about'/>
    </td> <td>
      <xsl:value-of select='name()'/>
    </td> <td>
      <xsl:call-template name="value">
        <xsl:with-param name='v' select='.'/>
      </xsl:call-template>
    </td> </tr>
  </xsl:for-each>
</xsl:template>
```

However RDF/XML syntax is extremely versatile and less and less used and, most of all, writing XSLT templates for it would be very complex considering the many potential serializations of an RDF statement: the transformation rules would depend on the concrete XML syntax of RDF instead of its semantics.

GRDDL [8] is a mechanism for extracting RDF data from XML documents. A GRDDL profile is associated to an XSLT stylesheet and can be specified in any XML document conform to the targeted model or *dialect* to order GRDDL agents to extract RDF data from it. GRDDL could then be used to extract RDF data from RDF data in RDF/XML syntax. However this W3C recommandation has never really been adopted and, like XSLT, this solution would rely on the many concrete XML syntaxes of RDF instead of its semantics.

OWL-PL [9] is an extension of XSLT for transforming RDF/OWL into XHTML. It provides an adaptation of XSLT processing of XML trees to RDF graphs. In particular, it matches properties of resources instead of XML nodes through XPath. OWL-PL is both tied to its RDF/XML input format, like XSLT. Xenon [10] is another ontology for specifying in RDF how RDF resources should be presented to the user. It reuses many of the key ideas of XSLT, among which templates, and defines a so-called RDF Stylesheet language. Xenon's two foundational concepts are lenses and views. Lenses specify which properties of an RDF resource are displayed and how these properties are ordered; views specify how they are displayed. Both OWL-PL and Xenon are tied to a specific display paradigm and an XHTML-like output format.

Fresnel [11] is an RDF vocabulary for specifying in RDF which data contained in an RDF graph should be displayed and how. Fresnel's two foundational concepts are lenses and formats. Fresnel's formats generalize Xenon's views. Figure 1 presents a Fresnel RDF graph describing a presentation format for RDF data on persons: a lense specifies that for each person, her name, mbox and picture should be displayed and a format specifies how to display her name.

```
PersonLens a fresnel:Lens ;
  fresnel:classLensDomain foaf:Person ;
  fresnel:showProperties
    ( foaf:name foaf:mbox foaf:depiction ).
:nameFormat a fresnel:Format ;
  fresnel:label "Name" ;
  fresnel:propertyFormatDomain foaf:name .
```

Fig. 1. Fresnel RDF graph.

SPARQL is provided with a CONSTRUCT query form which enables to extract and *transform* RDF data into RDF. A CONSTRUCT query returns an RDF graph specified by a graph template in the CONSTRUCT clause of the query and built by substituting the variables in the graph template with the solutions to the WHERE clause.

[12] addresses the problem of generating XML from RDF data with an extended SPARQL query. A SPARQL query is given a template of XML document where variables are fed with the query results. The SPARQL CONSTRUCT clause is overloaded to refer to an XML template with reference to SPARQL query variables that are bound by a standard WHERE clause.

XSPARQL [13] is a combination of SPARQL and XQuery [14] enabling to query both XML and RDF data and to transform data from one format into the other. XPARQL integrates within XQuery the SPARQL WHERE clause to facilitate the selection of RDF data to transform it into XML and, conversely, the SPARQL CONSTRUCT clause to facilitate the construction of RDF graphs from some extracted XML data. For instance the XSPARQL statements in Fig. 2 enable to select RDF data on persons and transform it into an XML tree describing relations between persons.

[15] proposes an XML-based transformation language, inspired by XSLT, that mainly matches types of RDF resources. [16] proposes an XML-based stylesheet language also inspired by XLST where templates match triple patterns and generate HTML.

Finally, there are quite a wide range of RDF parsers and validators[1], some of which enable to transform RDF data from one serialization format to another. Among them, let us cite RDF Distiller[2] and RDF Translator[3]. A review of

[1] http://www.w3.org/2001/sw/wiki/Category:Tool.

[2] http://rdf.greggkellogg.net/distiller.

[3] http://rdf-translator.appspot.com/.

```
declare foaf="http://xmlns.com/foaf/0.1/";
<relations> {
  for $Person $Name from <relations.rdf>
  where {$Person foaf:name $Name}
  order by $Name
  return
    <person name = "{$Name}"> {
    for $FName
    where {$Person foaf:knows $Friend .
      $Friend foaf:name $FName}
    return <knows>{$FName}</knows> }
    </person> }
</relations>
```

Fig. 2. XSPARQL.

these RDF-to-RDF converters can be found in [17]. Another famous example of specific-purpose RDF transformer is the RDF/XML parser in OWL API[4] [18] which enable to transform OWL 2 statements in RDF/XML into the functional syntax of the language.

To sum up, the state-of-the-art solutions presented in this section to transform RDF data are all tied to either an RDF/XML input syntax or to a specific output format, or both — except Fresnel. But Fresnel focuses on the *presentation* of RDF data and does not handle the general problem of the *transformation* of RDF data. In the following, we present a *generic* approach for writing RDF transformers for any output language.

3 SPARQL Template Transformation Language

SPARQL Template Transformation Language (STTL) is a generic transformation rule language for RDF based on SPARQL. It relies on two extensions of SPARQL: an additional TEMPLATE query form to express transformation rules and extension functions to recursively call the processing of templates into another one. Section 3.1 summerizes the key features of SPARQL and Sects. 3.2 and 3.3 present the extensions of SPARQL in STTL. Section 3.4 presents STTL syntax; Sect. 3.5 presents the compilation of STTL into standard SPARQL; and Sect. 3.6 presents STTL semantics. Finally, Sect. 3.7 compares STTL to XSLT.

3.1 SPARQL

SPARQL is the query language for RDF recommended by W3C. It has a SQL-like syntax (SELECT FROM WHERE) and is a graph pattern matching language. A SPARQL query is a set of triple patterns that are RDF triples (in Turtle syntax) which may hold variables. A query may also have operators such as

[4] http://owlapi.sourceforge.net/.

filter, conjunction, union, optional, minus, etc. An example of SPARQL query searching resources with name "Olivier" which are linked to other resources with a knows property is shown below:

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT * WHERE {
  ?x foaf:name"Olivier" ;
     foaf:knows ?y .
  FILTER (?x != ?y)
}
```

SPARQL is also provided with a CONSTRUCT WHERE query form the result of which is a graph. The CONSTRUCT clause specifies a graph pattern with variables which are replaced by the values found in the solutions of the WHERE clause in order to create a graph. For instance, the SPARQL query below enables to construct the RDF graph of foaf:knows inverse relations between the resources of the original RDF graph.

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
CONSTRUCT {
  ?y foaf:knows ?x
}
WHERE {
  ?x foaf:knows ?y
}
```

3.2 SPARQL Template Query Form

STTL relies on two extensions of SPARQL: an additional TEMPLATE query form and extension functions to process a set of templates such as st:apply-templates.

A TEMPLATE query is made of a TEMPLATE clause and a standard WHERE clause. The WHERE clause is the condition part of a rule, specifying the nodes in the RDF graph to be selected for the transformation. The TEMPLATE clause is the presentation part of the rule, specifying the output of the transformation for the RDF statements matching the condition.

For instance, let us consider the OWL 2 axiom stating that the class of men and the class of women are disjoint. Here is its expression in functional syntax: DisjointClasses(a:Man a:Woman); and here it is in Turtle: a:Man owl:disjointWith a:Woman.

The following template enables to transform the above RDF statement into the corresponding statement in functional syntax.

```
TEMPLATE {
  "DisjointClasses(" ?in " " ?c ")"
}
WHERE {
  ?in owl:disjointWith ?c
}
```

The WHERE clause matches the RDF statement and enables to select the subject and object of property `owl:disjointWith` and to bind them to variables `?in` and `?c`. The TEMPLATE clause specifies the result that must be generated using the solution sequence of the WHERE clause. Variables in the TEMPLATE clause are replaced by their value displayed in the Turtle syntax.

The value of a variable may be a blank node that represents another OWL statement, e.g., a Restriction. In this case, we would like to display not the blank node itself but the result of the transformation of the target OWL statement. This can be done using other templates.

3.3 SPARQL Template Extension Functions

Let us now consider the OWL 2 axiom stating that the class of parents is equivalent to the class of individuals having a person as child. Here are its expressions in functional syntax and in Turtle:

```
EquivalentClasses(a:Parent
  ObjectSomeValuesFrom(a:hasChild a:Person))

a:Parent owl:equivalentClass
    [ a owl:Restriction ;
      owl:onProperty a:hasChild ;
      owl:someValuesFrom a:Person ]
```

The template below enables to transform the above RDF statement into the corresponding functional statement.

```
TEMPLATE {
 "EquivalentClasses("
   st:apply-templates(?in) " "
   st:apply-templates(?c)
 ")"
}
WHERE {
  ?in owl:equivalentClass ?c .
}
```

The value matching variable `?in` is `a:Parent` which is expected in the transformation output (the functional syntax of the OWL 2 statement), while the value matching variable `?c` is a blank node whose property values are used to build the expected output.

This is defined in another template to be applied on this focus node. The `st:apply-templates` extension function[5] enables this recursive call of templates, where `st` is the prefix of STTL namespace:
http://ns.inria.fr/sparql-template/

In other words, hierarchical processing is done using the `st:apply-templates` function in the TEMPLATE clause. It returns the result of the application

[5] Named in reference to XSLT `xsl:apply-templates`.

of other templates to the focus nodes. The result is concatenated in the TEMPLATE clause. Hence, *nested templates* processing is performed by dynamic calls to `st:apply-templates`.

According to the definition of `PrimaryExpression` in SPARQL grammar, both SPARQL functions and extension functions can be used in TEMPLATE clauses. The following SPARQL extension functions have been defined to process a transformation:

- `st:apply-templates(term)` calls the transformer on a focus node `term` and executes one template;
- `st:call-template(name, term)` calls a template by its name on a focus node `term`;
- `st:apply-templates-with(uri, term)` calls the transformation specified by `uri` on a focus node `term` and executes one template;
- `st:call-template-with(uri, name, term)` calls a template by its name, on a focus node with a specified transformation;
- `st:apply-templates-all(term)` calls the transformer on a focus node `term` and executes all templates; it returns the concatenation of the results.
- `st:turtle(term)` returns the Turtle form of an RDF term;
- `st:define()` and `st:process()` enable to parameterize the transformer behaviour when used in a predefined template presented in Sect. 4.4.

3.4 Syntax

Figure 3 presents STTL's grammar. It is based on SPARQL 1.1's grammar[6]. In the definition of `Template`, `Prologue`, `DatasetClause`, `WhereClause`, `SolutionModifier` and `ValuesClause` are those defined in SPARQL grammar. In the definition of `TemplateClause`, `iri` is a template name. In the definition of `Term`, `PrimaryExpression` is that defined in SPARQL grammar and `Group` is syntactic sugar for SPARQL GROUP_CONCAT aggregate, enabling an easier writing of aggregation with several arguments. `Separator` enables to define the separator of aggregates; by default, it is the space character for `Group` (see Sect. 3.5) and the newline character for `TemplateClause` (see Sect. 3.6).

3.5 Compilation into Standard SPARQL

A template can be compiled into a standard SPARQL query of the SELECT form. The compilation keeps the WHERE clause, the solution modifiers and the VALUES clause of the template unchanged and the TEMPLATE clause is compiled into a SELECT clause. Here is the compilation scheme of a TEMPLATE clause into a SELECT clause:

[6] http://www.w3.org/TR/sparql11-query/#sparqlGrammar.

```
Template ::= Prologue TemplateClause
   DatasetClause* WhereClause
   SolutionModifier ValuesClause

TemplateClause ::=
  ′TEMPLATE′ (iri VarList ?) ?
  ′{′ Term* Separator? ′}′

VarList ::= ′(′ Var+ ′)′

Term ::= PrimaryExpression | Group

Group ::= ′GROUP′ ′DISTINCT′? ′{′
   PrimaryExpression* Separator? ′}′

Separator ::= ′;′ ′separator′ ′=′ String
```

Fig. 3. STTL grammar.

```
(1) cp(TemplateClause(Term(t1), ... Term(tn), sep)) =
    SELECT (concat(cp(t1), ... cp(tn)) AS ?out)

(2) cp(Group(Term(t1), ... Term(tn), sep)) =
    group_concat(concat(cp(t1), ... cp(tn)), sep)

(3) cp(Var(v)) = st:process(v)

(4) cp(PrimaryExpression(if(e1, e2, e3))) =
    if(e1, cp(e2), cp(e3))

(5) cp(PrimaryExpression(e)) = e
```

Basically, a recursive function `cp` compiles a TEMPLATE clause by concatenating the compilation of its terms by a call to function CONCAT (1). A GROUP is syntactic sugar for GROUP_CONCAT aggregate (2); a variable `v` in a TEMPLATE clause is compiled into the `st:process(v)` function call (3); `if` function call arguments are compiled except the condition which is left unchanged (4); other primary expressions are left unchanged (5). The `st:process` function returns the Turtle format of the RDF term, it can be overloaded (see Sect. 4.4).

For instance, by applying the above scheme, the following STTL expression:

```
TEMPLATE {
  "ObjectAllValuesFrom(" ?p " " ?c ")"
}
WHERE {
  ?in a owl:Restriction ;
    owl:onProperty ?p ;
    owl:allValuesFrom ?c
}
```

is compiled into the following SPARQL query:

```
SELECT
  (concat("ObjectAllValuesFrom(",
   st:process(?p), " ",
   st:process(?c), ")") AS ?out)
WHERE {
  ?in a owl:Restriction ;
    owl:onProperty ?p ;
    owl:allValuesFrom ?c
}
```

3.6 Evaluation Semantics

Since STTL can be compiled into standard SPARQL 1.1, the evaluation semantics of a template is that of SPARQL[7]. Let Ω the solution sequence resulting from the evaluation of the SPARQL query resulting itself from the compilation of a template. If the solution sequence is empty, the template fails. Otherwise, we define the result of the evaluation of a template as the result of the `Aggregation` operator of SPARQL Algebra[8] with the following arguments:

`Aggregation((?out), group_concat, scalarvals,` $\{$`1 ->` Ω $\}$`)`

where `scalarvals` corresponds to the `sep` argument in the TEMPLATE clause. Matching the graph pattern in the WHERE clause of a template may return several solutions: Ω is a solution sequence. However, the result of the evaluation of a template is unique: the solutions in Ω are aggregated with an additional GROUP_CONCAT aggregate. The result of the template is the result of GROUP_CONCAT.

3.7 Comparison of STTL and XSLT

STTL and XSLT are quite similar in their functionalities and expressiveness. The key difference between both languages is that XSLT operates on a XML (ordered) tree whereas STTL operates on an RDF (unordered) graph. Hence XSLT/XPath queries such as *the 3rd son of a node* may not be computable with STTL (and with SPARQL) in the general case because RDF edges are not ordered. However, it is possible to simulate ordering in RDF by assigning an explicit number to each son.

Both languages share declarative template rules and named templates as well as `apply-templates` and `call-template` functions. Both share conditional statement, sorting and grouping. XSLT holds explicit repetition `xsl:for-each` whereas in STTL it is implicit: the template clause is implicitly applied to all solutions. In addition, STTL also manages a `group` statement.

A XSLT template can match several patterns in the body whereas in a STTL template, there is one WHERE clause resulting in one multiset of solutions.

[7] http://www.w3.org/TR/sparql11-query/#sparqlAlgebraEval.

[8] http://www.w3.org/TR/sparql11-query/#aggregateAlgebra.

On an other hand, STTL inherits all SPARQL 1.1 statements, including the *service* clause which enables to perform transformations on Linked Data remote graphs.

Numbering. XSLT proposes a `xsl:number` numbering instruction that generates numbers according to the place of current node in the tree. We introduce in STTL a `st:number` extension function that generates a number for each solution of the `where` clause, taking into account the ORDER BY clause. Hence, the numbers are generated after the sorting of the solution sequence.

```
TEMPLATE {
  st:number() " " ?x " " ?y
}
WHERE {
  ?x ex:link ?y
}
ORDER BY ?x ?y
```

In presence of `st:number`, the result of the template clause is a *Future* datatype value, that is a specific datatype the value of which is completely determined later (i.e. after ORDER BY occurs). The result of the template clause is:
`Future(concat(st:number(), text))`
where `text` represents the constant part of the result:
`text = concat(" ", ?x, " ", ?y).`

At the end of query processing, ORDER BY sorts solutions. At the end of template processing, the additional aggregate operation (see Sect. 3.6) concatenates the value of template clause for all solutions into one textual result. This GROUP_CONCAT aggregate eventually evaluates the *Future* datatype value. As the solutions are sorted, the value of `st:number()` in a given solution is the index of the solution in the solution sequence, starting at index 1. The extended aggregate hence computes:
`group_concat(?out) =`
`group_concat(concat(st:number(), text)).`

4 Implementation

We implemented a SPARQL Template Transformation engine within the Corese Semantic Web Factory[9] [19,20]. Basically, it is called by the `st:apply-templates` extension function, or any other transformation functions introduced in Sect. 3.3. Given an RDF graph with a focus node to be transformed and a list of templates, the transformation engine successively tries to apply them to the focus node until one of them succeeds. A template succeeds if the matching of the WHERE clause succeeds, i.e., returns a result.

[9] http://wimmics.inria.fr/corese.

4.1 Algorithm

Here is the core algorithm of the `st:apply-templates` function in pseudocode:

```
(1) Node st:apply-templates(Node node) {
(2)   for (Query q : getTemplates()) {
(3)     Mappings map = eval(q, IN, node);
(4)     Node res = map.getResult(OUT);
(5)     if (res != null) return res;
(6)   }
(7)   return st:default(node);
(8) }
```

Templates are selected (2) and tried (3) one by one until one of them returns a result (4–5). In other words, a template is searched whose WHERE clause matches the RDF graph with the binding of the focus node to variable IN. If no template succeeds, the `st:default` function is applied to the node (7). Recursive calls to `st:apply-templates` implements the graph recursive traversal with successive focus nodes: `eval` (3) runs templates that recursively call the `st:apply-templates` function.

In addition to the above pseudocode, the transformer checks loops in case the RDF graph is a cyclic graph. It keeps track of the templates applied to nodes in order to avoid recursively applying the same template on the same node twice. If no fresh template exists for a focus node, the transformer returns the value returned by a call to the `st:default` function.

A call to any other transformation functions introduced in Sect. 3.3 triggers a similar algorithm.

4.2 Dynamic Variable Binding

When matching the WHERE clause of a template with the RDF graph, the SPARQL query evaluator is called with a binding of variable IN (`?in` in the WHERE clause) with the focus node to be transformed. When processing templates, the SPARQL interpreter must then be able to perfom dynamic binding to transmit the focus node. This dynamic value binding can be implemented in SPARQL with an extension function `st:getFocusNode()` that retrieves the focus node from the environment and a SPARQL BIND clause to bind it to the `?in` variable in the WHERE clause:

```
BIND(st:getFocusNode() AS ?in)
```

The same scheme can be used for named templates with arguments.

4.3 Template Selection

By default, the transformation engine considers templates in order: given a focus node, in response to a call to the `st:apply-templates` function, it considers the first template that matches this node. Hence, the result of the transformation of the focus node is the result of this template. Named templates can be chosen to be processed by a call to the `st:call-template` function.

In some cases, it is worth writing *several* templates for a type of focus node, in particular when the node holds different graph patterns that should be transformed according to several complementary rules. Executing several templates on the focus node is done by calling the `st:apply-templates-all` function in the TEMPLATE clause. The result of the transformation is the concatenation of the results of the successful templates.

A transformer can be used to transform a whole RDF graph. For this purpose, the `st:apply-templates- with` function can be called without focus node and the transformer must then determine it. By default, the first template that succeeds is the starting point of the transformer; or a `st:start` named template can be defined to be executed first (see Sect. 4.4).

4.4 Transformer Setting

Our implementation of STTL enables to simply set a special default template selection behavior for a set of templates defining a transformation, by defining two special named templates: `st:start` and `st:default`. The `st:start` template, if any, is selected at the beginning of the transformation process when no focus node is available. In that case, it is the first template executed by the template engine. The `st:default` template, if any, is executed when all templates fail to match the focus node.

The default processing of a variable in the TEMPLATE clause consists in outputting its value in the Turtle format. A specific named template `st:profile` can be used to overload this default transformation behaviour. For example, the definition shown below specifies that processing a variable, noted `st:process(?x)`, consists in the application of `st:apply-templates` to blank nodes and `st:turtle` to URIs and literals.

```
TEMPLATE st:profile {
  st:define (st:process(?x) =
    if (isBlank(?x), st:apply-templates(?x), st:turtle(?x))
  )
}
WHERE { }
```

5 Validation

In our approach of RDF transformation based on SPARQL templates, the template processor is completely generic: it applies to any RDF data or any language or model provided with an RDF syntax. What is specific to each output language or format is the set of transformation rules defined for it. In other words, each transformer specific to an output format is a specific set of templates processed by the generic template processor implementing STTL.

In this section we present specific transformers available online[10] which validate both STTL and our implementation of a generic STTL processor. The applications of STTL are many and varied. A first family of applications deals with the

[10] http://ns.inria.fr/sparql-template.

presentation of RDF data into specific syntaxes, e.g., Turtle, (see Sect. 5.1), or presentation formats, e.g., HTML (see Sect. 5.2), or any other format answering specific needs. As a result, STTL answers all the application scenarii addressed in the related work. A second family of applications deals with the transformation of statements of a given language represented in RDF syntax, e.g., OWL (see Sect. 5.3), or any other special purpose language with an RDF syntax. A third family of applications deals with the translation of RDF into other languages, e.g., RDF-to-CSV, or any translation X-to-Y of languages with RDF syntaxes.

5.1 RDF-to-RDF/Turtle Transformer

The following single STTL template enables to output RDF data in Turtle syntax.

```
TEMPLATE {
  ?x "\n"
  GROUP { ?p " " ?y ; separator = ";\n" }
  " . "
}
WHERE { ?x ?p ?y }
GROUP BY ?x
```

In a similar way, it is easy to write a transformer for each of RDF syntaxes.

5.2 RDF to HTML

We implemented a generic transformation to translate SPARQL query results into HTML. CONSTRUCT query returns an RDF graph whereas SELECT query result is translated in RDF using W3C DAWG result-set RDF vocabulary[11] which is an RDF version of SPARQL Query Results XML format.

The template below generates table cells for variable bindings:

```
prefix rs:
<http://www.w3.org/2001/sw/DataAccess/tests/result-set#>
TEMPLATE  {
  "<td>"
  coalesce(
    st:call-template(st:display, ?val), "&nbsp;")
  "</td>" ; separator = " "
}
WHERE {
  ?x rs:solution ?in
  ?x rs:resultVariable ?var
  OPTIONAL {
    ?in rs:binding [
      rs:variable ?var ; rs:value ?val ]
  }
}
ORDER BY ?var
```

[11] http://www.w3.org/2001/sw/DataAccess/tests/result-set.

Such RDF to HTML transformation enables us to design a Linked Data Navigator[12] on top of a local dataset as well as remote datasets such as DBpedia. The transformer is embedded in a Web server, that is a SPARQL endpoint augmented with a transformation engine. The transformation engine is accessible at a specific URI on the server. Given a resource URI, the transformation retrieves a description of the resource in the dataset and generates a HTML page accordingly. In the HTML page, references to related resource URI are displayed as hypertext links to the Web server.

5.3 OWL 2 Pretty-Printer

We wrote a transformation generating OWL 2 expressions in functional syntax from OWL 2 expressions in RDF as a set of 73 STTL templates.

We validated it on the OWL 2 Primer ontology[13] containing 350 RDF triples. To validate the result of the transformation, we loaded the output produced in OWL functional syntax into Protégé and did a complete cycle of transformation (save to RDF/XML, load and transform again) and we checked that the results were equivalent. Let us note that the results are equivalent and not identical because some statements are not printed in the same order, due to the fact that Protégé does not save RDF/XML statements exactly in the same order and hence blank nodes are not allocated in the same order.

We tested this OWL/RDF transformer on several real world ontologies, among which a subset of the *Galen* ontology. The RDF graph representing it contains 33080 triples, the size of the result is 0.58 MB and the (average) transformation time is 1.75 s. We also have tested our pretty-printer on the *HAO* ontology. The RDF graph representing it contains 38842 triples, the size of the result is 1.63 MB, the (average) pretty-print time is 3.1 s.

In addition to the transformation of an RDF graph representing an OWL ontology, this transformer can also be used when querying an OWL ontology stored in its RDF syntax, to present the results to the user in OWL 2 functional syntax. This is done by calling in the SELECT clause of the query one of the extension functions executing the transformer. As an example, the following query retrieves specific classes of the ontology and displays the results in functional syntax:

```
SELECT
  (st:apply-templates-with(st:owl, ?c) as ?t)
WHERE {
  ?c a owl:Class ;
     rdfs:subClassOf* f:Human
}
```

[12] http://corese.inria.fr.

[13] http://www.w3.org/TR/owl2-primer.

5.4 SPIN Pretty-Printer

SPIN is a representation of SPARQL abstract syntax trees in RDF [4]. It can be used to manage predefined SPARQL queries, rules as well as constaints. When using SPIN, the problem arises of presenting SPIN results in SPARQL syntax instead of SPIN/RDF syntax because the latter is difficult to read for users. For this purpose, we developed a SPIN to SPARQL pretty-printer using a transformation. The example below shows a SPARQL query and its SPIN representation.

```
PREFIX ex: <http://example.org/>
SELECT * WHERE {
  ?x ex:name ?y
}

@prefix sp: <http://spinrdf.org/sp#> .
@prefix ex: <http://example.org/> .
[ a sp:Select ; sp:star true ;
 sp:where (
  [ sp:subject   _:sb0 ;
    sp:predicate ex:name ;
    sp:object    _:sb1])
]
_:sb0 sp:varName "x" .
_:sb1 sp:varName "y" .
```

The SPIN pretty-printer contains 64 templates, it processes SPARQL 1.1 Query and Update. We validated the transformation by translating W3C SPARQL 1.1 test cases queries into SPIN and back to SPARQL and then evaluate the test cases with the resulting queries. The template below translates a SPIN triple into SPARQL syntax.

```
PREFIX sp: <http://spinrdf.org/sp#> .
TEMPLATE {
  ?x " " ?p " "  ?y " . "
}
WHERE {
?in sp:subject   ?x ;
    sp:predicate ?p ;
    sp:object    ?y
}
```

5.5 Example of an Entire STTL Transformation

In this section we detail the execution of the OWL transformation[14] on the following OWL/RDF statement:

[14] http://ns.inria.fr/sparql-template/owl.

```
a:Parent owl:equivalentClass [
  a owl:Restriction ;
  owl:onProperty a:hasChild ;
  owl:someValuesFrom a:Person
]
```

The transformation engine first searches a template that matches the statement owl:equivalentClass. Here is the retrieved template:

```
TEMPLATE {
if (bound(?t), "DatatypeDefinition", "EquivalentClasses")
"("  ?in  "" ?y  ")"
}
WHERE {
  ?in owl:equivalentClass ?y
  OPTIONAL { ?y a ?t filter(?t = rdfs:Datatype) }
}
```

The TEMPLATE clause is compiled to:

```
SELECT (concat(
if (bound(?t), "DatatypeDefinition", "EquivalentClasses"),
"(",  st:process(?in), " ", st:process(?y), ")"
) as ?out)
```

The WHERE clause of this template succeeds with the following bindings, where _:b is the blank node of type owl:Restriction:

```
  ?in = a:Parent ;
  ?y  = _:b ;
```

The TEMPLATE clause then evaluates its arguments: the ?t variable is not bound, hence the first expression evaluates to "EquivalentClasses". The following SELECT clause is eventually evaluated:

```
SELECT (concat("EquivalentClasses",
  "(", st:process(?in), " ", st:process(?y), ")")
as ?out)
```

st:process(?in) with ?in bound to IRI a:Parent then returns a:Parent. st:process(?y) with ?y bound to blank node _:b of type owl:Restriction, subject of properties owl:onProperty and owl:someValuesFrom, then searches a template that matches such statements. Here is the retrieved template:

```
TEMPLATE {
  if (bound(?t), "DataSomeValuesFrom",
    "ObjectSomeValuesFrom")
  "(" ?p " " ?z ")"
}
WHERE {
  ?in owl:someValuesFrom ?z ;
      owl:onProperty ?p
  OPTIONAL {
    ?p a ?t
    FILTER (?t = owl:DatatypeProperty) }
}
```

The TEMPLATE clause is compiled to:

```
SELECT
(concat(
  if (bound(?t), "DataSomeValuesFrom",
    "ObjectSomeValuesFrom"),
  "(", st:process(?p), " ", st:process(?z), ")"
as ?out)
```

The WHERE clause of this template succeeds with the following bindings:

```
  ?in = _:b ;
  ?z  = a:Person ;
  ?p  = a:hasChild
```

The TEMPLATE clause of the above template then evaluates its arguments: variables `?t` is not bound, hence the first expression evaluates to `"ObjectSomeValuesFrom"`.
The SELECT clause below is eventually evaluated:

```
SELECT (concat("ObjectSomeValuesFrom",
  "(", st:process(?p), " ", st:process(?z), ")")
as ?out).
```

As `?p` and `?z` are both bound to URIs, the evaluation of `st:process(?p)` and `st:process(?z)` eventually returns the Turtle format of these URI. The result of the above SELECT clause and therefore of the template is then:

```
"ObjectSomeValuesFrom(a:hasChild a:Person)"
```

As there is only one result, the final `group_concat(?out)` aggregate does not change it. This result is returned to the first template as the value of `st:process(?y)` in its SELECT clause . The result of this SELECT clause and therefore of the first template is then:

```
"EquivalentClasses(a:Parent
ObjectSomeValuesFrom(a:hasChild a:Person))"
```

This is precisely the expression in OWL functional syntax of the example of OWL statement chosen as input to illustrate the STTL transformation.

5.6 Design Patterns

In this section we present examples that shows some possibilities of STTL.

Recursion. Named templates can be recursively called. Hence it is possible, for example, to generate the development of factorial function.

```
TEMPLATE st:fac(?n) {
  if (?n = 0, 1,
     concat(?n, " . ", st:call-template(st:fac, ?n - 1)))
}
WHERE {
}
```

Property Path. SPARQL Property Path statement can be used to enumerate and display the elements of a list.

```
TEMPLATE {
  ?e
}
WHERE {
  ?in rdf:rest*/rdf:first ?e
}
```

Nested Query. Nested queries can be used in templates, e.g. for aggregation purpose. The example below counts and displays the number of resources instance of classes.

```
TEMPLATE {
  ?t " : " ?c
}
WHERE {
  SELECT ?t (count(?x) as ?c)
  WHERE {
    ?x a ?t
  }
  GROUP BY ?t
  ORDER BY desc(?c) ?t
}
```

Values Clause. A template can exploit the VALUES clause. The example below, extrated from SPIN, associate a string label to operators.

```
PREFIX sp: <http://spinrdf.org/sp#> .
TEMPLATE {
  "(" ?f " " str(?lab) " " ?r ")"
}
WHERE {
  ?in a ?ope ;
      sp:arg1 ?f ;
      sp:arg2 ?r
}
VALUES (?ope ?lab) {
  (sp:lt  "<")  (sp:gt  ">")
  (sp:le  "<=") (sp:ge  ">=")
  (sp:eq  "=")  (sp:ne  "!=")
}
```

SPARQL. Several transformations can be used in a query. In the example below, an OWL class is displayed in functional syntax and in RDF/Turtle syntax.

```
SELECT ?x
  (st:apply-templates-with(st:owl,    ?x) as ?o)
  (st:apply-templates-with(st:turtle, ?x) as ?t)
WHERE {
  ?x a owl:Class
}
```

The result of a transformation can be used in a SPARQL query, for example in a BIND or a FILTER clause. In the example below, the query searches occurrences of the string *"Annotation"* in the result of the functional syntax transformation.

```
SELECT *
WHERE {
  ?x a owl:Class
  BIND (st:apply-templates-with(st:owl, ?x) as ?fs)
  FILTER (contains(?fs, "Annotation"))
}
```

A transformation can be used to generate a string key for a subgraph. In the example below, the Turtle transformation is used to recursively generate keys for OWL *Restriction* statements. Then, it groups statements with the same key and counts the statements with same key.

```
SELECT ?key (count(?r) as ?c)
WHERE {
  ?r a owl:Restriction
}
GROUP BY (st:apply-templates-with(st:turtle, ?r) as ?key)
```

Template Selection. The name of a template to be called can be computed and bound to a variable. The template below binds the `?temp` variable with the `st:person` named template for resources of type `foaf:Person` and `st:resource` for other resources. The `st:call-template` function is called with `?temp` variable as argument.

```
TEMPLATE {
  st:call-template(?temp, ?y)
}
WHERE {
  ?in ex:relation ?y
  BIND (
     if (exists { ?y a foaf:Person }, st:person, st:resource)
  AS ?temp)
}
```

The name of transformations can be retrieved in the RDF graph if resources are annotated with transformation names. In the example below, the transformation name is retrieved from the class of the resource using the `st:transform` property. Hence, transformations may be part of a Semantic Web Knowledge Base.

```
TEMPLATE {
  st:apply-templates-with(?trans, ?in)
}
WHERE {
  ?in a ?c .
  ?c st:transform ?trans
}
```

Linked Data Transformation. In the spirit of Linked Data, a transformation can query a SPARQL endpoint to get additional information, using the SERVICE clause. In the example below, the template queries DBpedia to get the latitude and the longitude of a resource.

```
PREFIX p: <http://fr.dbpedia.org/property/>
TEMPLATE {
  st:call-template(st:locate, ?in, ?lat, ?lon)
}
WHERE {
  ?in a ex:Place
  SERVICE <http://fr.dbpedia.org/sparql> {
    ?in p:latitude  ?lat ;
        p:longitude ?long
  }
}
```

6 Conclusion and Future Work

In this paper we considered two related problems: (1) the transformation of RDF to present RDF data to users, e.g., into a HTML domain or application dependant format, and (2) the transformation of RDF when it is used as a meta-model to represent on the Web other languages and their abstract graph structure. We addressed the general problem of transforming RDF into other languages. We answered this question by specifying STTL, a generic and domain independent extension to SPARQL to support the declarative representation of any special-purpose RDF transformation as a set of transformation rules. Being based on SPARQL, STTL inherits its expressivity and its extension mechanisms. This specification and the algorithms we described have been implemented and tested in a generic transformation rule engine part of the Corese Semantic Web Factory platform [19,20]. This means all these results are part of this open-source platform. We demonstrated the feasibility and genericity of our approach by providing several transformations including: RDF-to-RDF syntaxes, RDF-to-HTML, RDF OWL 2-to-OWL 2 functional syntax.

As future work, regarding the performances of our generic transformation rule engine, we intend to improve them by implementing heuristics to optimize the selection of templates. We should compare in the short term the performance of our generic transformation rule engine with that of existing tools for specific RDF transformations. For instance, we may compare the performance of our engine with that of the parser of the well known OWL API[15] for transforming large OWL 2 ontologies from RDF/XML syntax into functional syntax.

Regarding the exploitation of our generic transformation rule engine to implement RDF transformers into specific languages, we intend to augment the number of STTL transformations available by writing rule sets for other formats and domains. In the cases where the TEMPLATE clauses of the transformation rules produce RDF triples (as text), we define RDF-to-RDF transformations. In particular, we envisage implementing a special case of RDF-to-RDF transformation to anonymize RDF datasets.

Acknowledgement. We would like to thank Fabien Gandon (Inria) for enlighting discussions on the nature of SPARQL-based transformations for RDF, Fuqi Song (Inria), Alban Gaignard (CNRS) and Eric Toguem (U. of Yaoundé, Cameroun) for the set up of the Web server.

References

1. Cyganiak, R., Wood, D., Lanthaler, M.: RDF 1.1 concepts and abstract syntax. Recommendation, W3C (2014). http://www.w3.org/TR/rdf11-concepts/
2. Patel-Schneider, P.F., Motik, B.: OWL 2 Web ontology language mapping to RDF graphs (second edition). Recommendation, W3C (2012). http://www.w3.org/TR/owl-mapping-to-rdf/
3. Sandro Hawke, A.P.: RIF in RDF. Working Group Note, W3C (2012). http://www.w3.org/TR/rif-in-rdf/
4. Knublauch, H.: SPIN - SPARQL Syntax. Member Submission, W3C (2011). http://www.w3.org/Submission/2011/SUBM-spin-sparql-20110222/
5. Follenfant, C., Corby, O., Gandon, F., Trastour, D.: RDF modelling and SPARQL processing of SQL abstract syntax trees. In: Programming the Semantic Web, ISWC Workshop,Boston, USA (2012)
6. Harris, S., Seaborne, A.: SPARQL 1.1 Query Language. Recommendation, W3C (2012). http://www.w3.org/TR/sparql11-query/
7. Kay, M.: XSL transformations (XSLT) version 2.0. Recommendation, W3C (2007). http://www.w3.org/TR/xslt20/
8. Connolly, D.: Gleaning resource descriptions from dialects of languages (GRDDL). Recommendation, W3C (2007). http://www.w3.org/TR/grddl/

[15] http://owlapi.sourceforge.net/.

9. Brophy, M., Heflin, J.: OWL-PL: a presentation language for displaying semantic data on the web. Technical report, Department of Computer Science and Engineering, Lehigh University (2009)
10. Quan, D.: Xenon: An RDF stylesheet ontology. In: Proceedings of the WWW (2005)
11. Pietriga, E., Bizer, C., Karger, D.R., Lee, R.: Fresnel: a browser-independent presentation vocabulary for RDF. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 158–171. Springer, Heidelberg (2006)
12. Alkhateeb, F., Laborie, S.: Towards extending and using SPARQL for modular document generation. In: Proceedings of the 8th ACM Symposium on Document Engineering, Sao Paulo, Brasil, pp. 164–172. ACM Press (2008)
13. Bischof, S., Decker, S., Krennwallner, T., Lopes, N., Polleres, A.: Mapping between RDF and XML with XSPARQL. J. Data Semant. **1**, 147–185 (2012)
14. Robie, J., Chamberlin, D., Dyck, M., Snelson, J.: XQuery 3.0: an XML query language. Recommendation, W3C (2014). http://www.w3.org/TR/xquery-30/
15. Shapkin, P., Shumsky, L.: A language for transforming the RDF data on the basis of ontologies. In: Proceedings of the 11th International Conference on Web Information Systems and Technologies (WEBIST), Lisbon, Portugal (2015)
16. Peroni, S., Vitali, F.: RSLT: RDF stylesheet language transformations. In: Proceedings of 12th ESWC Developers Workshop, Portoroz, Slovenia (2015)
17. Stolz, A., Rodriguez-Castro, B., Hepp, M.: RDF translator: a RESTful multi-format data converter for the Semantic Web. Technical report, E-Business and Web Science Research Group (2013)
18. Horridge, M., Bechhofer, S.: The OWL API: a java API for OWL ontologies. Semant. Web **2**, 11–21 (2011)
19. Corby, O., Gaignard, A., Faron-Zucker, C., Montagnat, J.: KGRAM versatile data graphs querying and inference engine. In: Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence, Macau (2012)
20. Corby, O., Faron-Zucker, C.: The KGRAM abstract machine for knowledge graph querying. In: IEEE/WIC/ACM International Conference, Toronto, Canada (2010)

Mobile Information Systems

A Sustainable Architecture for Secure and Usable Mobile Signature Solutions

Thomas Zefferer(✉)

Institute for Applied Information Processing and Communications,
Graz University of Technology, Inffeldgasse 16a, 8010 Graz, Austria
thomas.zefferer@iaik.tugraz.at
http://www.iaik.tugraz.at

Abstract. Electronic signatures are a crucial building block of transactional e-services. This especially applies to the European Union, where so-called qualified electronic signatures are legally equivalent to their handwritten pendant. For many years, signature solutions, which enable users to create electronic signatures, have been designed for classical end-user devices such as desktop computers or laptops. In most cases, these solutions cannot be easily applied on mobile end-user devices such as smartphones or tablet computers, due to the special characteristics of these devices. This complicates a use of transactional e-services on mobile devices and excludes a growing number of users, who prefer mobile access to services. To tackle this problem, this paper provides a basis for mobile signature solutions that are compatible to and applicable on mobile end-user devices. Possible architectures for these solutions are systematically derived from an abstract model first. Then, the best alternative is determined by means of systematic assessments. In particular, the aspects security and usability are considered in detail. This finally yields an implementation-independent and technology-agnostic architecture that can be used as basis for concrete implementations. By keeping the proposed solution on a rather abstract architectural level, its validity is assured, even if available mobile technologies and the current state of the art change. This way, the proposed architecture represents a sustainable basis for future mobile signature solutions and paves the way for transactional e-services on mobile end-user devices.

Keywords: Mobile signature solution · Mobile devices · Assessment · Architecture · Security · Usability

1 Introduction

Driven by the continuously increasing relevance of Internet-based procedures, electronic signatures have evolved to a fundamental building block of transactional electronic services (e-services). Relying on asymmetric cryptographic methods such as RSA [26] or ECDSA [5], electronic signatures provide integrity, authenticity, and non-repudiation. These capabilities make electronic signatures

V. Monfort et al. (Eds.): WEBIST 2015, LNBIP 246, pp. 343–364, 2016.
DOI: 10.1007/978-3-319-30996-5_17

the ideal choice in various areas of application. For instance, electronic signatures are frequently used in the e-government domain to obtain written consent from remote users. Furthermore, they are also employed by e-banking solutions to facilitate a remote authorization of financial transactions. The potential of electronic signatures has been especially recognized in the European Union (EU). There, EU laws such as the EU Signature Directive [28] or the EU eIDAS Regulation [29] provide a legal foundation for electronic signatures. Concretely, these laws define the concept of Qualified Electronic Signatures (QESs). QESs represent a special class of electronic signatures that need to fulfill specific requirements. Essentially, QESs are defined to be legally equivalent to handwritten signatures. This makes QESs especially relevant for transactional e-services that require legally binding written consent from remote users. The relevance of QESs raises the need for signature solutions that enable users to create legally binding electronic signatures in online procedures. During the past years, such solutions have been developed in various European countries. Examples are smart card based solutions, which have for instance been introduced and deployed in Austria [22], Belgium [17], or Portugal [2]. Other signature solutions available in Europe enable users to create QESs using their mobile phones. Examples are the Austrian Mobile Phone Signature [1] or the Estonian Mobiil-ID [20], which have been in productive operation for years. Irrespective of their underlying implementation, signature solutions have evolved to relevant building blocks of transactional e-services.

For a long time, transactional e-services have been developed for classical end-user devices such as desktop computers and laptops. Accordingly, existing signature solutions are tailored to the characteristics of these devices as well. This applies to smart card based solutions as well as to solutions relying on the user's mobile phone. Existing signature solutions implicitly assume that the user accesses e-services with a desktop computer or laptop and in addition makes use of a smart card or a mobile phone to create required QESs. Hence, the use of two separate devices has been an implicit assumption that has influenced design and development of current signature solutions. Unfortunately, this assumption is not valid any longer. During the past few years, desktop computers and laptops have been gradually replaced by smartphones and related mobile end-user devices. This raises various challenges for e-service, which need to be prepared for access by mobile devices. In particular, this also applies to existing signature solutions for the creation of QESs. Being tailored to the characteristics of classical end-user devices, these solutions usually cannot be applied on smartphones and other mobile end-user devices. For instance, mobile devices usually lack support for card-reading devices, which are a prerequisite for smart card based solutions. Similarly, signature solutions relying on mobile phones usually cannot be applied on smartphones or tablet computers either, as their underlying security concepts have been designed for scenarios, in which the mobile phone is used as an additional device and is solely used during the creation of QESs. The inappropriateness of existing signature solutions raises the need for new solutions that enable users to create QESs on modern mobile end-user devices.

Only if such solutions are provided, transactional e-services can be adapted such that they can be accessed from and used with mobile end-user devices.

To facilitate development and provision of such a solution, this paper identifies possible architectures and assesses them with regard to relevant success factors. This way, the best architecture for signature solutions tailored to mobile end-user devices is determined and a sustainable basis for concrete implementations is provided. To achieve this goal, a thorough methodology is followed, which is also reflected by the structure of this paper. In Sect. 2, requirements, success factors, and relevant target platforms are identified. From the identified requirements, an abstract model is derived in Sect. 3. This model is used to systematically identify possible architectures. In Sects. 4 and 5, all possible architectures are assessed in terms of the success factors identified in Sect. 2. This way, the most suitable architecture is determined. Conclusions are finally drawn in Sect. 6.

2 Preliminaries

Signature solutions that enable users to create QESs on their mobile devices need to satisfy several legal and technical requirements. At the same time, these solutions also need to consider relevant success factors, in order to achieve an adequate level of user acceptance. Finally, relevant target platforms must be identified, on which these solutions shall be applicable. As preparation for the identification of possible architectures, these aspects are discussed in this section in more detail.

2.1 Requirements

Putting the focus on the EU, relevant requirements for legally binding electronic signatures, i.e. QESs, are mainly defined by respective EU laws. Concretely, the EU eIDAS Regulation [29], which is about to replace the EU Signature Directive [28], defines the concept of QESs, the relation of QESs to other types of electronic signatures, and requirements that must be met by QESs. From the eIDAS Regulation, which will represent the relevant legal basis in the near future, the following set of basic requirements for QESs can be extracted:

- **R1: Reliance on QSCD:** QESs must be created with a Qualified Signature Creation Device (QSCD). QSCDs are certified hardware devices that are able to reliably and securely store cryptographic key material and that are capable to carry out cryptographic operations using this key material. Typical realizations of QSCDs are smart cards or hardware security modules (HSMs). Requirements of QSCDs are defined in Annex II of the EU eIDAS Regulation [29]. Requirements for the related concept of Secure Signature Creation Devices (SSCDs), which represent the pendant to QSCDs in the EU Signature Directive [28], are also defined in relevant standards such as the CEN Workshop Agreement 14169 [8].

- **R2: Reliance on Qualified Certificates:** According to the EU eIDAS Regulation [29], QESs must be based on qualified certificates. Qualified certificates are a subset of electronic certificates, which are the preferred means to link a user's identity to his or her cryptographic keys. Qualified certificates are a special kind of electronic certificates. They need to satisfy several requirements, which are all defined in Annex I of the EU eIDAS Regulation [29]. Most of these requirements define mandatory contents of qualified certificates.
- **R3: Appropriate User Authentication:** QESs are based on so-called advanced electronic signatures (AdESs), which are also defined by the EU eIDAS Regulation [29]. According to Art. 26 of this regulation, AdESs and hence also QESs must be *'created using electronic signature creation data that the signatory can, with a high level of confidence, use under his sole control'* [29]. This implies that solutions for the creation of QESs must implement a reliable user authentication and authorization mechanism. This mechanism reliably protects the user's cryptographic key material stored in the QSCD and assures that this material remains under the legitimate user's sole control. This usually implies application of a multi-factor authentication scheme including for example the authentication factors Knowledge and Possession.

By taking into account these three basic requirements, possible architectures of signature solutions for mobile end-user devices can be derived systematically. This will be elaborated in more detail in Sect. 3.

2.2 Success Factors

According to the followed methodology, identified possible architectures will be assessed with regard to relevant success factors. So far, there is hardly any specific related work that focuses on success factors of signature solutions for mobile devices. Hence, we derive relevant success factors from related work on mobile applications, mobile government, and signature solutions for classical end-user devices. Concretely, we consider works by El-Kiki et al. [13,14], Al-Khamayseh et al. [4], Karan et al. [21], and Al-Hadidi et al. [3].

Comparison of these works and merging their results and conclusions is rather difficult, as they all identify relevant success factors on different abstraction levels. However, several common findings can be extracted by choosing an adequate level of abstraction. Concretely, the factors Security, Usability, and Feasibility are identified as key for success by most authors. We hence focus on these factors when assessing possible architectures of signature solutions for mobile end-user devices. The factors Security and Usability will be explicitly considered by conducting respective assessments in Sect. 4 and in Sect. 5, respectively. In addition, the success factor Feasibility is implicitly considered by limiting the identification of possible architectures to those being feasible on current mobile end-user devices.

2.3 Target Platforms

To assess possible architectures, characteristics of mobile platforms, i.e. mobile end-user devices and mobile operating systems, must be taken into account. Currently available mobile platforms differ in terms of inherent characteristics, provided features, capabilities, and limitations. Hence, a growing number of considered platforms increases the complexity of conducted assessments. Therefore, we limit our assessments to the two currently dominating platforms Google Android [18] and Apple iOS [6]. This is reasonable, as these two platforms together share more than 95 % of the entire smartphone market [23].

3 Possible Architectures

The development of signature solutions that support the creation of QESs is a challenging task, as legal requirements, relevant success factors, and typical characteristics of employed end-user devices need to be considered. In this section, possible architectures for such solutions are identified. For this purpose, an abstract model is introduced first. Subsequently, possible architectures are derived systematically from this abstract model.

3.1 Abstract Model

Definition and use of abstract models for signature solutions is an approach commonly followed in standardization works and in scientific literature. For instance, models of signature-creation processes and solutions are given in documents published by the European Committee for Standardization (CEN). For example, the CEN Workshop Agreement (CWA) 14169 [8] identifies security requirements for SSCDs. It does so by identifying core components of signature-creation solutions that include an SSCD and by assembling these components to a simple model. Since CWA 14169 mainly focuses on aspects related to the SSCD, additional parties and components that play a role in signature-creation solutions are only briefly sketched. These components are further detailed and considered by a model defined by CWA 14170 on security requirements for signature creation applications [9]. CWA 14170 defines a functional model for signature-creation solutions that rely on an SSCD and on a so-called Signature Creation Application (SCA), which provides access to the SSCD. An adapted version of this model is also used in the draft version of ETSI's conformity assessment for signature creation and validation [16]. In this document, the rather complex model from CWA 14170 is reduced to a few components. In addition, exchanged information between components of this reduced model is also considered.

Identification, definition, and development of models of signature-creation solutions have also been topics of interest in the scientific domain. Recent scientific publications have used different models to systematically discuss concepts related to electronic signatures. For example, Leitold et al. [22] have discussed security concepts of the Austrian Citizen Card by means of a simplified model of a signature-creation system. This model identifies and combines components of smart card based signature solutions. For instance, the model considers PIN pads, SSCDs, and card-acceptor devices. To serve the purpose of the presented paper, Leitold et al. use a specific model that is especially tailored to the special use case of smart card based signature solutions. In contrast, Arnellos et al. [7] apply a more abstract model, which is used to discuss the structural reliability of signed documents on a semantic level. These two rather contrary examples show that for scientific work on electronic signatures comparable considerations apply as for relevant standards on this topic: employed models heavily depend on the context and the use case, for which they have been defined and developed. Still, most models share several similarities and are often based on common definitions, terms, and notations, which are usually provided by legal foundations such as the EU Signature Directive [28] or the eIDAS Regulation [29].

The set of models used in public standards and published scientific work show that a specific model for signature-creation solutions needs to be defined for each context. Accordingly, it makes sense to define a specific model for the provision of signature solutions for mobile end-user devices. The model proposed for this purpose is shown in Fig. 1. It takes into account the three basic requirements defined by the legal basis of QES. Furthermore, the proposed model is intentionally kept on a technology-agnostic and implementation-independent level. This way, it can serve as basis for the systematic identification of possible architectures. According to the chosen level of abstraction, the proposed model mainly identifies relevant components of signature solutions supporting QESs and illustrates basic interactions between these components.

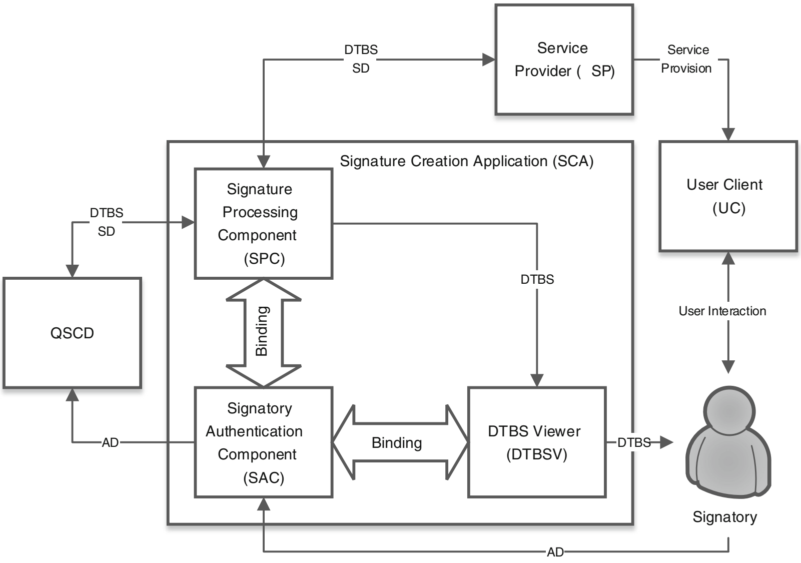


Fig. 1. Abstract model.

The abstract model shown in Fig. 1 comprises the Signatory and four top-level components, i.e. the QSCD, the SCA, the Service Provider (SP), and the User Client (UC). For the SCA, internal components are modeled as well. Concretely, the functionality of the SCA is covered by three subcomponents, i.e. the Signature Processing Component (SPC), the Signatory Authentication Component (SAC), and the DTBS Viewer (DTBSV). Each of these components covers specific functionality of the SAC. According to the model shown in Fig. 1, a typical signature-creation process comprises the following steps:

1. The Signatory uses the UC to access a service provided by the SP. For instance, the Signatory could use a web browser to access a service provided by a web application. To enable user interaction, the UC needs to implement a user interface to the Signatory.
2. The SP requires the Signatory to create a QES. To start the signature-creation process, the SP sends the Data-To-Be-Signed (DTBS) to the SPC.
3. The SPC forwards the DTBS to the QSCD, which is required to create the QES according to Requirement R1. Note that the SPC is the only component that is able to send DTBS to the QSCD.
4. To cover Requirement R3, the QSCD requires the Signatory to provide valid Authentication Data (AD), in order to authorize the signature-creation process in the QSCD. Depending on the concrete implementation, AD can for instance be realized by means of a PIN or password.
5. Required AD are collected by the SAC. For this purpose, the SAC needs to implement a user interface to the Signatory.
6. At the same time, the SPC forwards the DTBS to the DTBSV. The DTBSV displays the DTBS. This enables the Signatory to check whether or not he or she wants to provide the required AD. Displaying the DTBS again requires a user interface to the Signatory.
7. After provision of the AD, the SAC forwards the AD to the QSCD. Note that the SAC is the only component that is able to send AD to the QSCD.
8. The QSCD creates the QES on the DTBS using the Signatory's private signing key.
9. The resulting Signed Data (SD) are returned to the SPC, which forwards them to the SP.
10. The Signatory is notified of the successful signature-creation process via the UC.

In addition to this basic process flow, two aspects need to be noted. First, there must be bindings between subcomponents of the SCA as illustrated in Fig. 1. These bindings assure that the DTBS forwarded to the QSCD correspond to the DTBS displayed to the Signatory and that provided AD are used to authorize the signing of displayed DTBS only. Concrete implementations of the abstract model must assure that these bindings can be appropriately verified. Second, Requirement R2 is not directly covered by the proposed abstract model. As Requirement R2 mainly concerns the structure and contents of issued signing certificates, this requirement needs to be considered during the certificateissuing process. This mainly concerns the responsible Certification Authority (CA), which is not directly involved in signature-creation processes.

3.2 Architecture Candidates

The abstract model shown in Fig. 1 identifies relevant components and defines the basic process flow of a signature-creation process. Due to its abstract nature, this model is perfectly suitable to systematically derive possible architectures of signature solutions for mobile end-user devices.

To further develop the abstract model towards a concrete solution, the concrete implementation of all identified components needs to be fixed. Aiming for a signature solution that can be applied on mobile end-user devices, identified components can—from a pure conceptual perspective—be implemented in two different ways. They can either be implemented locally on the mobile device or remotely in a server environment. Outsourcing functionality to remote entities is common practice for mobile applications, as they usually have to cope with limited processing power and storage capacities on mobile end-user devices. In this context, especially cloud-based approaches can be efficient means to overcome issues caused by limited hand-held devices.

By varying locally and remotely implemented components and functionalities, different architectures can be derived. As the abstract model defined in Fig. 1 comprises six (sub)components, there are 64 different variations in theory. However, only three (sub)components can be implemented remotely in practice,

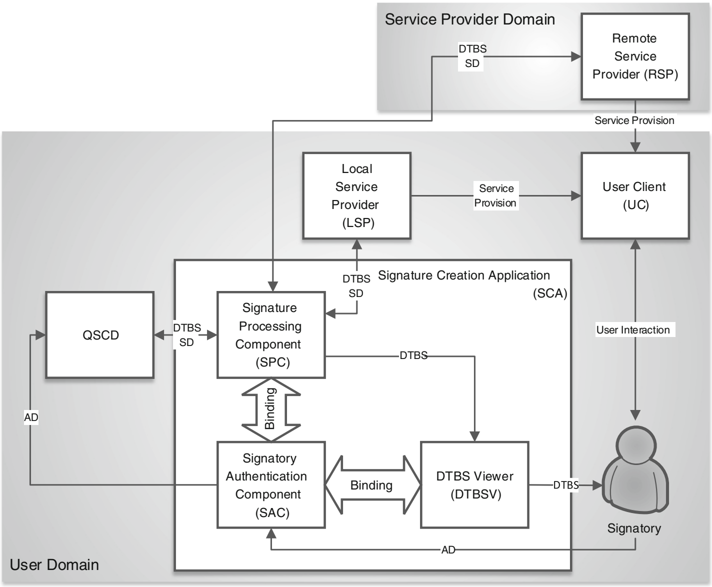


Fig. 2. Architecture Candidate A (AC A).

as all components with direct user interface to the Signatory must be realized locally in any case. Hence, only the QSCD, the SPC, and the SP can be implemented either locally or remotely in practice. This reduces the number of possible variations to eight. The eight remaining variations can be subsumed to four architectures by varying the implementation of the QSCD and the SPC only. Accordingly, each of these four architectures must consider both a Local Service Provider (LSP) and a Remote Service Provider (RSP). The four resulting architectures are shown in Figs. 2, 3, 4 and 5.

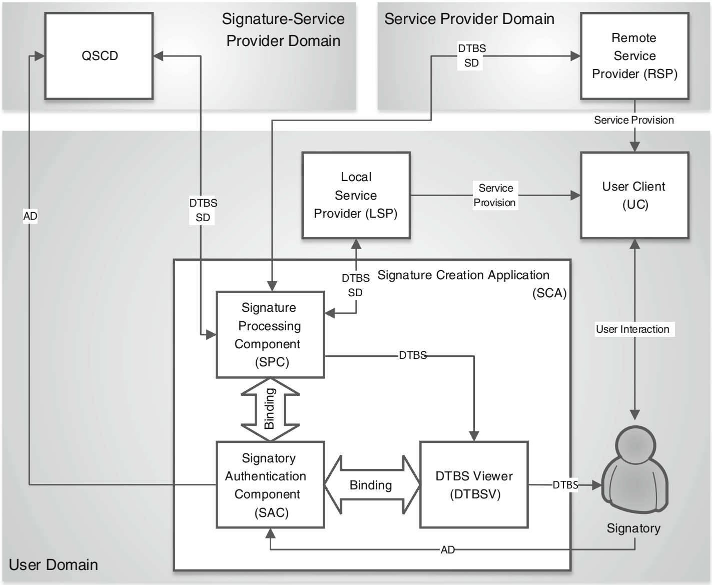

QSCD
Signature-Service Provider Domain
Service Provider Domain
DTBS SD
Remote Service Provider (RSP)
Service Provision
AD
DTBS SD
Local Service Provider (LSP)
Service Provision
User Client (UC)
DTBS SD
Signature Creation Application (SCA)
Signature Processing Component (SPC)
DTBS
User Interaction
Binding
Signatory Authentication Component (SAC)
Binding
DTBS Viewer (DTBSV)
DTBS
Signatory
AD
User Domain

Fig. 3. Architecture Candidate B (AC B).

The four possible architectures, i.e. architecture candidates (ACs), all comprise the same components as the abstract model, from which they have been derived. However, depending on the respective AC, the components are spread over different domains. The components LSP, UC, DTBSV, and SACs are implemented in the local User Domain in any case. Similarly, the RSP is always implemented in the remote Service Provider Domain. Hence, the four ACs mainly differ regarding their implementation of the components QSCD and SPC.

As all ACs have been derived from the abstract model shown in Fig. 1, they implicitly comply with the requirements identified in Sect. 2. Hence, they can all be used as basis for the development of signature solutions for mobile end-user devices. To determine the most appropriate approach, the four ACs are assessed with regard to the identified relevant success factors Security and Usability in the following sections.

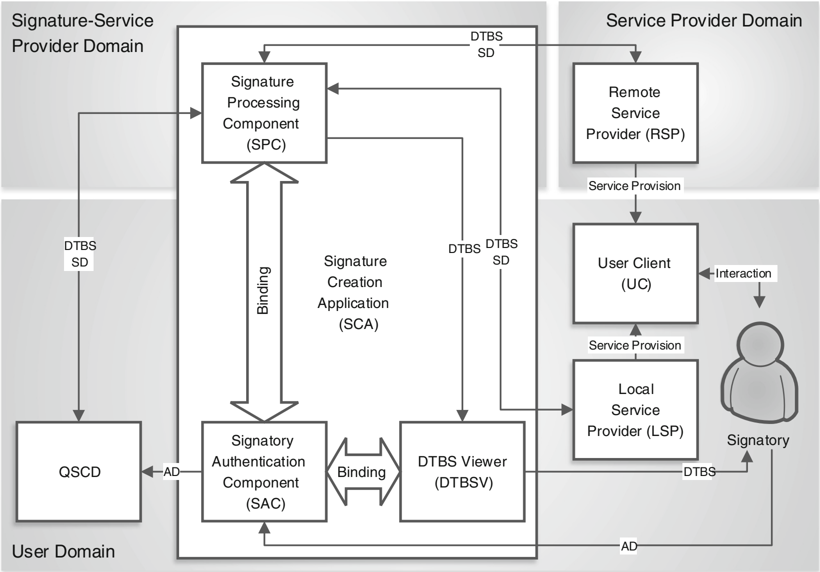

Signature-Service Provider Domain
Service Provider Domain
DTBS
SD
Signature Processing Component (SPC)
Remote Service Provider (RSP)
Service Provision
DTBS
SD
Binding
Signature Creation Application (SCA)
DTBS
DTBS
SD
User Client (UC)
Interaction
Service Provision
Local Service Provider (LSP)
Signatory
QSCD
AD
Signatory Authentication Component (SAC)
Binding
DTBS Viewer (DTBSV)
DTBS
AD
User Domain

Fig. 4. Architecture Candidate C (AC C).

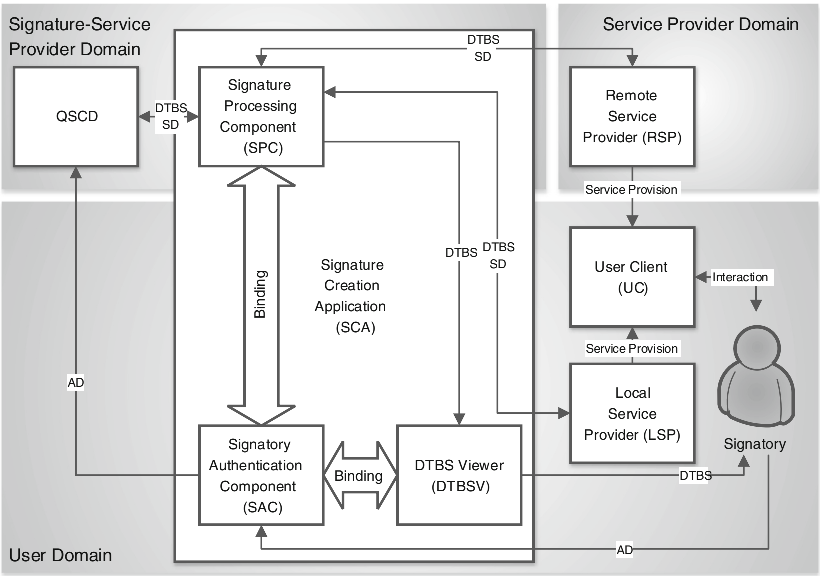

Signature-Service Provider Domain
Service Provider Domain
DTBS
SD
QSCD
DTBS
SD
Signature Processing Component (SPC)
Remote Service Provider (RSP)
Service Provision
Binding
Signature Creation Application (SCA)
DTBS
DTBS
SD
User Client (UC)
Interaction
Service Provision
AD
Local Service Provider (LSP)
Signatory
Signatory Authentication Component (SAC)
Binding
DTBS Viewer (DTBSV)
DTBS
AD
User Domain

Fig. 5. Architecture Candidate D (AC D).

4 Security Assessment

Security is crucial for signature solutions that enable users to create legally binding QESs. These solutions must assure that created signatures cannot be forged and that access to secret cryptographic signing keys is restricted to the legitimate Signatory. The level of provided security depends to a large extent on the concrete implementation. However, security is also influenced by the respective implementation's underlying architecture. Hence, in this section the four identified ACs, which can serve as basis for concrete implementations, are assessed in order to reveal their advantages and disadvantages in terms of security.

4.1 Methodology

A thorough security assessment requires an elaborate methodology to assure meaningful results. Existing approaches such as Common Criteria (CC) based concepts [12] are useful to assess a concrete solution but are less effective for the comparisons on a pure conceptual level. Hence, we only roughly base the methodology followed on the concept of CC, but adapt it where necessary. Concretely, the following steps are conducted to assess and compare the security of the four ACs. First, a few assumptions are made to define the scope of the assessment. Then, relevant assets are identified and mapped to the four ACs, in order to identify protection-deserving components and communication paths. This way, the security of the four ACs can be compared on conceptual level. Finally, capabilities of the two chosen target platforms to ensure the security of protection-deserving components and communication paths are analyzed. This way, the implementation perspective is taken into account as well.

4.2 Assessment

According to the methodology defined above, the security of the four identified ACs is assessed. This is detailed in the following subsections.

Assumptions: Following the defined methodology, the scope of the conducted security assessment is defined by making two basic assumptions. First, we assume that server-based components are secure. This is reasonable, as these components can be operated in an especially protected environment such as certified data-processing centers. Second, we assume that QSCDs are secure. This is also a valid assumption, as QSCDs need to undergo strict certification procedures, which assure that QSCDs meet defined requirements.

Assets: After defining the security assessment's scope, relevant assets need to be identified. For the present use case, relevant assets can be extracted directly from the abstract model shown in Fig. 1. Concretely, the following assets can be identified:

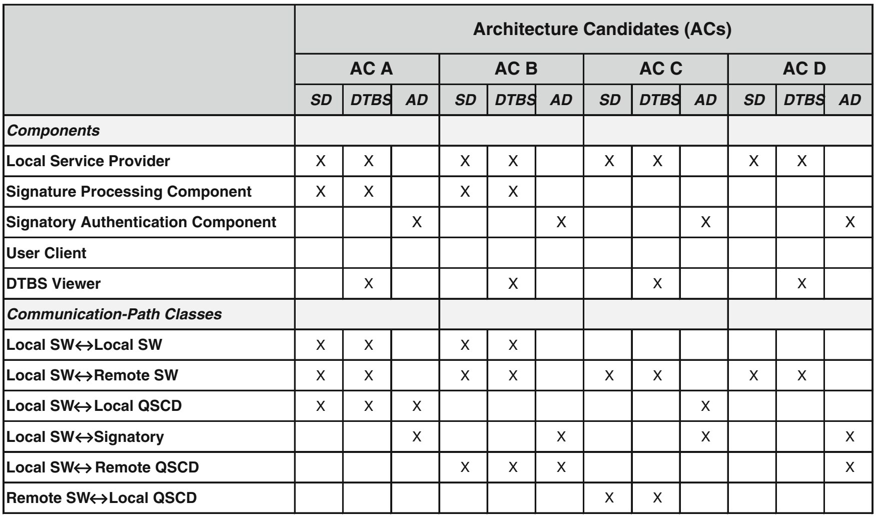
- **DTBS:** The DTBS must be protected from eavesdropping and unauthorized modifications by adversaries. This is crucial to assure that DTBS defined by the SP are not altered before being signed. Furthermore, the DTBS must also be protected when being displayed to the Signatory via the DTBSV. This is necessary in order to enable the Signatory to check what data is about to be signed.
- **AD:** The confidentiality of the AD must be assured. This prevents adversaries, who are able to intercept AD, from reusing them and from creating QESs on behalf of the legitimate Signatory.
- **SD:** The integrity of the SD must be guaranteed, in order to prevent adversaries from applying modifications to invalidate the created QES. Depending on the use case, the confidentiality of the SD might also be necessary.

Note that there are several additional assets that deserve protection. For instance, the Signatory's secret signing key must be kept confidential in any case. However, this key and other related assets are stored inside the QSCD. As the QSCD is assumed to be secure, assets protected by this component are not considered in detail for the conducted security assessment.

Conceptual Assessment: With the help of the three identified assets, the security of the four ACs can be compared on conceptual level. For this purpose, Fig. 6 lists all components, on which identified assets are potentially prone to attacks. Essentially, these are all components used by the four ACs that are not assumed to be secure. In addition, Fig. 6 also lists all relevant communication paths between relevant components, on which transferred assets are potentially prone to attacks. For the sake of simplicity, several communication paths have been combined to derive six more general classes of communication paths. For instance, the communication paths between the Signatory and the DTBSV, and between the Signatory and the SAC have been combined to one communication-path class (Local Software (SW) ↔ Signatory). For all listed components and communication-path classes, we have assessed their relevance for the four ACs. Concretely, we have assessed for each AC, which assets are present at which components and communication-path classes. The results of this mapping are shown in Fig. 6.

Obtained results indicate that AC D is advantageous from a conceptual perspective. AC D has the lowest number of components and communication-path classes, on which assets are prone to attacks. Hence, from a pure conceptual perspective, this architecture candidate appears to be advantageous. This result is even more unambiguous, if the possibility of an LSP is ruled out. This precondition reduces the number of relevant components and communication-path classes and further emphasizes the advantage of AC D. This is illustrated in Fig. 7.

Mapping to Target Platforms: Comparison on a pure conceptual level clearly indicates AC D to be advantageous. However, the plain number of potentially vulnerable components and communication-path classes can of course be



	Architecture Candidates (ACs)											
	AC A			AC B			AC C			AC D		
	SD	*DTBS*	*AD*	*SD*	*DTBS*	*AD*	*SD*	*DTBS*	*AD*	*SD*	*DTBS*	*AD*
Components												
Local Service Provider	X	X		X	X		X	X		X	X	
Signature Processing Component	X	X		X	X							
Signatory Authentication Component			X			X			X			X
User Client												
DTBS Viewer		X			X			X			X	
Communication-Path Classes												
Local SW↔Local SW	X	X		X	X							
Local SW↔Remote SW	X	X		X	X		X	X		X	X	
Local SW↔Local QSCD	X	X	X						X			
Local SW↔Signatory			X			X			X			X
Local SW↔ Remote QSCD				X	X	X						X
Remote SW↔Local QSCD							X	X				

Fig. 6. Conceptual architecture comparison.

a first indicator only. In addition, capabilities to protect these components and communication-path classes using currently available technologies must also be taken into account. This can be achieved by analyzing related work on security features and vulnerabilities of current mobile platforms. Such analyses have for instance been provided by Enck et al. [15] or Rogers et al. [27]. From the results of these works, several interesting findings can be derived.

The security of all local components processing one or more assets depends to a large extent on security features provided by the mobile platform, on which these components are implemented. Concretely, this applies to the components LSP, SPC, SAC, and DTBSV. In practice, the security of these components depends on the underlying platform's capabilities to protect local software. On both target platforms considered in this paper, i.e. Android and iOS, third-party software must be implemented by means of mobile apps. Both platforms feature various security mechanisms that improve the security of installed apps. Examples are sandboxing mechanisms, which isolate installed apps from each other. A detailed overview of security mechanisms integrated into Android and iOS has been provided by Rogers et al. [27] and Zefferer et al. [30]. In principle, integrated security mechanisms work reliably in practice and provide a sufficient level of security. However, they become useless, if attackers gain root access to the mobile operating system, e.g. by exploiting known vulnerabilities. During the past years, especially Android has turned out to be prone to such attacks. This is mainly due to Android's considerable fragmentation, i.e. the fact that there are numerous customized and modified Android versions in the field, which are often not updated any longer [25]. In case a mobile end-user device or operating system is prone to root attacks, the security of installed apps cannot be taken

	Architecture Candidates (ACs)											
	AC A			AC B			AC C			AC D		
	SD	*DTBS*	*AD*	*SD*	*DTBS*	*AD*	*SD*	*DTBS*	*AD*	*SD*	*DTBS*	*AD*
Components												
Signature Processing Component	X	X		X	X							
Signatory Authentication Component			X			X			X			X
User Client												
DTBS Viewer		X			X			X			X	
Communication-Path Classes												
Local SW↔Local SW	X	X		X	X							
Local SW↔ Remote SW	X	X		X	X		X	X		X	X	
Local SW↔ Local QSCD	X	X	X						X			
Local SW↔ Signatory			X			X			X			X
Local SW↔Remote QSCD				X	X	X						X
Remote SW↔Local QSCD							X	X				

Fig. 7. Conceptual architecture comparison considering remote service providers only.

for granted any longer. It can hence be concluded that even though featured security mechanisms provide a certain level of security, absolute security of local components must not be assumed. Even though Android must be considered as more prone to rooting, this threat basically applies to both target platforms and implies that locally processed and stored assets are potentially vulnerable.

While root access is a problem for all local components, several specific aspects need to be considered for the component SAC. The functionality of this component is rather simple and basically limited to obtaining AD from the Signatory and forwarding obtained AD to the QSCD. Because of its limited functionality, the SAC can also be implemented by other means than mobile apps. For instance, if a local Subscriber Identity Module (SIM) is used as QSCD, a simple SIM application can be used to implement the functionality of the SAC. In this case, the provided security level of SIM applications is relevant. Unfortunately, an attacker with unlimited root access to the operating system must be assumed to have the opportunity to compromise SIM applications as well. Hence, also alternative implementations of the SAC do not provide absolute security. Another relevant aspect is the fact that the SAC potentially needs to implement enhanced authentication functionality in case a remote QSCD is used. In this case, the QSCD does not implicitly cover the authentication factor Possession, as it is not in physical possession of the Signatory. Hence, more sophisticated authentication mechanisms potentially need to be implemented by the SAC, in order to implement multi-factor authentication schemes. Depending on the concrete implementation of these schemes, this can enable additional attack vectors. For instance, several e-banking solutions rely on one-time passwords delivered via Short Message Service (SMS) to cover the authentication factor Possession. By proving reception of a one-time password delivered via SMS, the user proves possession of his or her SIM. The so-called Eurograbber attack campaign has demonstrated weaknesses of this authentication scheme when being applied on modern smartphones that enable malware access to incoming SMS messages [10].

Except for the SAC, the achievable security is comparable for all local components that store or process assets. In contrast, the situation is more complex for the different communication-path classes, over which assets are transmitted. The security of assets transmitted between different local software components (Local SW ↔ Local SW) mainly depends on inter-process communication (IPC) capabilities of the underlying mobile platform. The two target platforms Android and iOS differ significantly in this aspect. Android provides broad support for IPC and enables an easy exchange of data between local components. This improves the feasibility of mobile applications, but reduces security, as IPC features can also be employed by e.g. malware to compromise assets. This is especially the case if provided features are used in a wrong way [11]. On Android, the security of data exchanged between local software components hence depends heavily on a correct use of provided IPC features. On iOS, the situation is less critical, as only limited IPC features are provided. This decreases feasibility, but at the same time reduces the probability of security-critical implementation errors.

While the communication between local software components is potentially problematic, data exchange between local and remote software (Local SW ↔ Remote SW) can be protected by means of established protocols. For instance, Transport Layer Security (TLS) [24] can be used to reliably assure the confidentiality and integrity of data exchanged between local and remote software. In contrast, the communication path between the Signatory and local software is more problematic. Neither Android nor iOS provide secure input/output capabilities. ARM TrustZone is a potential future solution to this problem, but not yet available on most mobile devices.

Regarding the communication between local software and a local QSCD (Local SW ↔ Local QSCD), a trade-off between feasibility and security can be identified. Android provides mobile apps the opportunity to access local hardware elements that implement QSCD functionality. However, this implies that also malware residing on the mobile device can do so. In contrast, iOS is more restrictive. This reduces the feasibility of ACs relying on local QSCDs on this platform, but also prevents access to hardware elements by malware. Where available, secure communication between local software and local QSCDs must not be taken for granted.

In case of local software and a remote QSCD (Local SW ↔ Remote QSCD), the situation is different. As this setup requires cross-domain communication, additional remote software is needed to extent the QSCD's functionality by means of cross-domain communication capabilities. Again, communication between this module and local software can be easily secured with the help of established protocols such as TLS. The situation is more difficult in case of remote software and a local QSCD (Remote SW ↔ Local QSCD). Again, cross-domain communication is required, which implies the need for an additional module to enhance the QSCD with cross-domain communication capabilities. If this module is implemented as mobile app, communication between this app and the local QSCD is required. This corresponds to the communication-path class Local SW ↔ Local QSCD. If the local SIM is used as QSCD, the required additional module can also be implemented as SIM application. However, this requires a cooperation of the mobile network operator, as communication between remote software and the local SIM needs to rely on the mobile network.

Focusing on the implementation perspective with special regard to the two target platforms Android and iOS shows that security cannot be assured for various components and communication-path classes. By mapping obtained findings to the four ACs, the best AC, i.e. the one that includes the fewest problematic components, can be identified. For a comparative analysis, it is sufficient to focus on those components and communication-path classes, for which there are differences between the four ACs. Figure 6 shows that this applies to one component and four communication-path classes.

Concretely, the SPC is the only component that shows AC-specific differences regarding the three relevant assets. Concretely, the assets SD and DTBS are prone to attacks on the component SPC for AC A and AC B only. Hence, AC C and AC D are advantageous in this regard. Similar conclusions can be drawn for the communication between local software components. This is required for the exchange of assets by AC A and AC B only. As IPC is potentially insecure on the assessed target platforms, AC A and AC B must be regarded as disadvantageous.

The remaining three communication-path classes, for which there are differences between the four ACs, are all related to data exchange between software components and the QSCD. Simply counting the occurrences of assets on these communication-path classes indicates that AC D is advantageous from a conceptual perspective. Following this approach, only the asset AD is transmitted once. For all other ACs, all three assets are present at least once on one of the three communication-path classes. The advantage of AC D is also revealed when taking into account a concrete implementation on the two target platforms. As discussed above, secure communication is easier to achieve for remote QSCDs than for local QSCDs. Thus, AC B and AC D are advantageous in this regard.

4.3 Findings

From the conducted security assessment of the four ACs, several findings can be derived. A comparison of the four ACs on a pure conceptual level yields AC D to be advantageous, as this AC shows the lowest number of components and communication paths, on which assets are potentially prone to attacks. Similar findings are also obtained, when the implementation perspective and the current state of the art are taken into account. Concretely, the conducted assessment has revealed that a local implementation of the SPC is disadvantageous. Furthermore, a local implementation of the QSCD has turned out to be disadvantageous as well. Thus, AC D is also advantageous from an implementation perspective. The other three ACs suffer from the inability to provide a sufficient level of security on current mobile platforms. Although the concrete threat potential depends on the functionality being implemented locally, AC D is advantageous in any case, as it implements as many components remotely as possible.

5 Usability Assessment

In addition to security, usability has also been identified as crucial success factor. In this section, we hence assess the usability of the four identified ACs. More precisely, we assess the ACs' capabilities to serve as basis for concrete usable implementations. For this purpose, the methodology followed is introduced first. Based on this methodology, the usability assessment is then conducted. Finally, relevant findings obtained from the conducted assessment are summarized.

5.1 Methodology

As both chosen target platforms provide similar input and output capabilities, the basic level of usability provided by Android and iOS is the same. The usability of a mobile signature solution hence mainly depends on its underlying architecture and implementation. In contrast to the conducted security assessment, platform specifics therefore do not need to be taken into account. Instead, the usability of the four ACs can be compared on a completely platform-independent level. Accordingly, the following methodology has been followed for the conducted usability assessment. First, usability-influencing aspects are derived from related scientific work. The four ACs are then assessed and ranked according to the derived aspects. This finally yields the AC that is best suited to serve as basis for usable signature solutions.

5.2 Assessment

Following the defined methodology, the four ACs are assessed in this section. For this purpose, relevant usability aspects are identified first. The assessment is then based on these aspects.

Usability Aspects: Following the methodology defined, relevant usability aspects are derived from related scientific work. Unfortunately, there is hardly any specific work on relevant usability factors for mobile signature solutions. The probably best suited approach for the present use case has been introduced by Harrison et al. [19], who have proposed the PCMAD usability model for mobile applications. This model defines the aspects Effectiveness, Efficiency, Satisfaction, Learnability, Memorability, Errors, and Cognitive Load to be crucial for mobile applications. Recent usability analyses of signature solutions for classical end-user devices have shown that the need for certain software and hardware can also be a usability-reducing factor [31]. By combining this finding with the PCMAD model proposed by Harrison et al. [19], the following set of relevant usability aspects can be derived: Effectiveness, Efficiency, Satisfaction, Hardware Independence, Software Independence, Learnability, Memorability, Cognitive Load, and Error Robustness. Based on these aspects, the usability of the four ACs is analyzed in the following subsection.

Usability Analysis: The usability assessment of the four ACs is based on nine usability aspects. In the following, each of these aspects is discussed separately for the four ACs. In addition, the four ACs are ranked according to their capability to comply with the respective aspect.

Harrison et al. describe the aspect Effectiveness as the *'ability of a user to complete a task in a specified context'* [19]. In general, all ACs provide service availability, i.e. allow the successful completion of signature-creation processes. However, service availability is reduced, if required remote components cannot be accessed, e.g. because of a lacking Internet connection. In this regard, AC A is advantageous, as it implements locally as many components as possible. Service availability can also be reduced by the need for additional entities. Solutions based on AC C typically require the Signatory's mobile network operator to access the local QSCD. This can be problematic in roaming scenarios. In summary, AC A is ranked best, as it totally avoids remote components and is independent from additional entities. AC C is ranked last, as it relies on remote components and typically requires on an additional external entity.

For the usability aspect Efficiency, the situation is slightly different. Efficiency defines the speed and the accuracy, with which an intended task can be completed by the user [19]. With regard to mobile signature solutions, efficiency is mainly affected by the complexity of the required user-authentication process, as this is the only mandatory user interaction. In general, user authentication is typically more complex in case of remotely implemented QSCDs. In this case, the QSCD is not under physical control of the Signatory and hence cannot implicitly cover the authentication factor possession. Therefore, more complex authentication schemes must be applied to assure an adequate level of security, which in turn increases complexity. Accordingly, AC A and AC C are by trend more efficient than AC B and AC D.

The factor Satisfaction describes *'the perceived level of comfort and pleasantness afforded to the user through the use of software'* [19]. The aspect Satisfaction is hence mainly influenced by the provided user interface (UI). As the UI rather depends on the concrete implementation than on the underlying architecture, a conceptual comparison of the four ACs is difficult. Nevertheless, AC C can be identified as slightly disadvantageous. Its reliance on a local QSCD and a remote SPC potentially requires an integration of alternative technologies such as SIM applications. These technologies provide fewer opportunities to implement satisfactory UIs. Accordingly, AC C is ranked worse compared to the other three ACs with regard to the aspect Satisfaction.

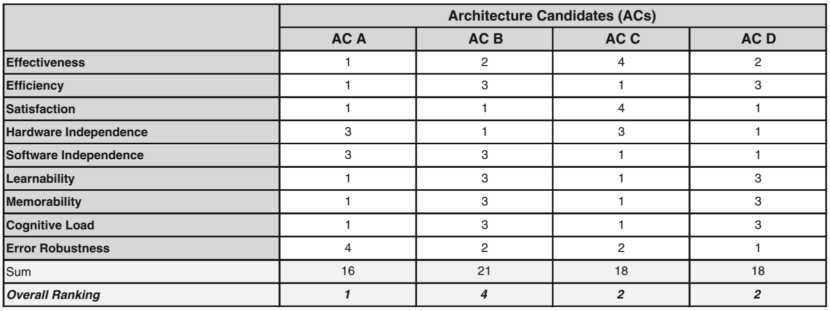
The usability aspect Hardware Independence must be considered, as the need to acquire and maintain additional hardware potentially reduces usability. This has been shown by means of several usability tests. With regard to the four ACs, AC A and AC C must be regarded as disadvantageous. Accordingly, they are ranked worse than AC B and AC D. Similar to the aspect Hardware Independence, also the factor Software Independence is crucial for usability. Again, the need to acquire and maintain additional software potentially reduces usability. AC A and AC B implement the SPC locally and hence realize more software components on the mobile end-user device than AC C and AC D. Hence, with regard to the aspect Software Independence, AC C and AC D are advantageous and are ranked better than AC A and AC B.

The ability of a user to learn how to use an application is covered by the aspect Learnability and also contributes to an application's overall usability. The learnability of an application is closely related to its complexity, which is measured by the aspect Efficiency. Hence, similar conclusions can be drawn for the aspect Learnability as for the aspect Efficiency. ACs requiring a more complex user-authentication scheme are more complex and hence more difficult to learn. Accordingly, AC B and AC D are ranked worse than AC A and AC C. Similar results can also be obtained for the aspects Memorability and Cognitive Load. Memorability describes the ability of a user to retain how to use an application. Cognitive Load describes the impact that using the respective application has on the performance of other tasks that are carried out in parallel. Similar to Learnability, Memorability and Cognitive Load are both related to the application's complexity. Hence, also for these aspects, AC A and AC C are advantageous.

Finally, Error Robustness has been identified as crucial usability aspect as well. For a conceptual comparison of different ACs, it can be assumed that remotely implemented components are less prone to errors. This is reasonable, as remote components can be implemented in data-processing centers that feature redundancy mechanisms to assure required service levels. Under this assumption, the error robustness of an AC increases with the number of remote components. This yields AC D to be the most advantageous solution followed by AC B and AC C. AC A shows the poorest error robustness.

5.3 Findings

Figure 8 summarizes the results of the conducted usability assessment. For each identified aspect, Fig. 8 shows the determined ranking of the four ACs. In addition, an overall ranking is derived from the aspect-specific results. This yields AC A as the most usable AC. AC C and AC D share the second rank.



	Architecture Candidates (ACs)			
	AC A	AC B	AC C	AC D
Effectiveness	1	2	4	2
Efficiency	1	3	1	3
Satisfaction	1	1	4	1
Hardware Independence	3	1	3	1
Software Independence	3	3	1	1
Learnability	1	3	1	3
Memorability	1	3	1	3
Cognitive Load	1	3	1	3
Error Robustness	4	2	2	1
Sum	16	21	18	18
Overall Ranking	***1***	***4***	***2***	***2***

Fig. 8. Usability assessment.

While results obtained from the conducted usability assessment seem quite clear, they suffer from several limitations. First, usability has been assessed by means of abstract architectures only. In practice, usability is also heavily influenced by the concrete implementation of a specific architecture. Second, derivation of the overall ranking has implicitly assumed that all usability aspects are equally important. This is usually not the case in practice, where the relevance of different aspects depends on the concrete context.

Nevertheless, the conducted usability assessment yields several useful findings, as it clearly indicates, which architectures are beneficial regarding which usability aspects. Obtained results also show that AC D is the most bipolar AC. For most assessed aspects, AC D is either among the best or among the worst candidates. Most usability aspects, for which AC D is disadvantageous, are related to the aspect efficiency. Hence, the main usability drawback of AC D is apparently its reduced efficiency, which is mainly caused by the need for more complex user-authentication schemes. If this drawback can be removed, AC D will represent the most usable AC.

6 Conclusions

The continuing success and popularity of mobile computing raises the requirement for signature solutions that can be applied and used on mobile end-user devices. In this paper, this problem has been addressed by identifying possible architectures for such solutions. In total, four ACs denoted as AC A, AC B, AC C and AC D have been systematically derived, which basically cover all possible implementation variants. Pros and cons of the four ACs have been revealed by means of a security and usability assessment. With regard to security, AC D has turned out to be advantageous, as it avoids to a large extent the implementation of components on potentially insecure mobile devices. With regard to usability, AC A has been determined as best alternative. AC D has reached the second rank and has also been identified as most bipolar alternative, whose usability could be significantly improved by means of efficient user-authentication schemes. Overall, AC D can hence be identified as the overall winner, if concrete solutions basing on this architecture manage to implement appropriately efficient user-authentication schemes.

All models and architectures proposed and developed in this paper have been intentionally kept on an abstract level. The same holds true for conducted assessments, which have also mainly been applied on a conceptual level. Staying on an abstract level might appear disadvantageous at a first glance, as this approach does not immediately yield a concrete implementation or product. However, the mobile market is currently undergoing fast and frequent technological changes. Providing a concrete solution that bases on the current state of the art is hence not sustainable. Therefore, the solution developed in this paper remains independent from the current state of the art. This way, this paper provides a sustainable architectural basis for future mobile signature solutions and paves the way for transactional e-services on mobile end-user devices in the long term.

References

1. A-Trust: Handy-Signatur - Your digital identity (2015). https://www.handy-signatur.at
2. Agência para a Modernização Administrativa: Cartão de Cidadão (2015). http://www.cartaodecidadao.pt
3. Al-Hadidi, A., Rezgui, Y.: Critical success factors for the adoption and diffusion of m-Government services: a literature review. In: Proceedings of the European Conference on e-Government, ECEG, pp. 21–28 (2009)
4. Al-khamayseh, S., Lawrence, E., Zmijewska, A.: Towards understanding success factors in interactive mobile government (2007). http://www.mgovernment.org/
5. ANSI: Public Key Cryptography for the Financial Services Industry, The Elliptic Curve Digital Signature Algorithm (ECDSA) (2005) http://webstore.ansi.org/RecordDetail.aspx?sku=ANSI+X9.62%3A2005
6. Apple: iOS 8 (2015). https://www.apple.com/at/ios/
7. Arnellos, A., Lekkas, D., Zissis, D., Spyrou, T., Darzentas, J.: Fair digital signing: the structural reliability of signed documents. Comput. Secur. **30**(8), 580–596 (2011). http://www.sciencedirect.com/science/article/pii/S016740481100112X
8. CEN: CWA 14169 - Secure Signature-Creation Devices "EAL 4+". Technical report, European Committee for Standardization (2004)
9. CEN: CWA 14170 - Security Requirements for Signature Creation Applications (2004). http://standards.cen.eu/dyn/www/f?p=204:110:0::::FSP_PROJECT,FSP_ORG_ID:23764,400296&cs=1C1B2F4DF3464C9FD768CB422F16D3387
10. Check Point Software Technologies Ltd: Media Alert: Check Point and Versafe Uncover New Eurograbber Attack (2012). http://www.checkpoint.com/press/2012/120512-media-alert-cp-versafe-eurograbber-attack.html
11. Chin, E., Felt, A.P., Greenwood, K., Wagner, D.: Analyzing Inter-application communication in Android. In: Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services, MobiSys 2011, pp. 239–252. ACM Press (2011). http://www.eecs.berkeley.edu/~emc/papers/mobi168-chin.pdf
12. Common Criteria (2013). http://www.commoncriteriaportal.org/
13. El-Kiki, T.: mGovernment: a reality check. In: Conference Proceedings of the 6th International Conference on the Management of Mobile Business, ICMB 2007, p. 37. IEEE (2007)
14. El-Kiki, T., Lawrence, E.: Mobile user satisfaction and usage analysis model of mGovernment services. In: Proceedings of the Second European Mobile Government Conference, pp. 91–102 (2006)
15. Enck, W., Ongtang, M., McDaniel, P.: Understanding android security. IEEE Secur. Priv. **7**, 50–57 (2009)
16. ETSI: Conformity Assessment for Signature Creation and Validation Applications (2014). http://docbox.etsi.org/esi/Open/Latest_Drafts/prEN_419103_v002_conformity-assessment-sign-creation-validation_COMPLETE-draft.pdf
17. Fairchild, A., de Vuyst, B.: The evolution of the e-ID card in Belgium: data privacy and multi-application usage. In: Sixth International Conference on Digital Society, pp. 13–16, Valencia (2012)
18. Google: Android (2015). https://www.android.com/
19. Harrison, R., Flood, D., Duce, D.: Usability of mobile applications: literature review and rationale for a new usability model. J. Interact. Sci. **1**(1), 1 (2013)
20. ID.ee: Mobiil-ID (2015). http://id.ee/index.php?id=36881

21. Karan, K., Khoo, M.: Mobile diffusion and development: issues and challenges of m-Government with India in perspective. In: Proceedings of the 1st International Conference on M4D Mobile Communication Technology for Development, pp. 138–149 (2008)
22. Leitold, H., Hollosi, A., Posch, R.: Security architecture of the Austrian citizen card concept. In: 2002 Proceedings of the 18th Annual Computer Security Applications Conference, pp. 391–400 (2002)
23. mobiForge: Mobile software statistics 2014 (2015). http://mobiforge.com/research-analysis/mobile-software-statistics-2014
24. Network Working Group: The Transport Layer Security (TLS) Protocol Version 1.2 (2008). http://tools.ietf.org/rfcmarkup/5246
25. OpenSignal: Android fragmentation visualized. Technical report (2014). http://opensignal.com/reports/2014/android-fragmentation/
26. Rivest, R.L., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM **21**(2), 120–126 (1978)
27. Rogers, M., Goadrich, M.: A hands-on comparison of iOS vs. Android. In: Proceedings of the 43rd ACM Technical Symposium on Computer Science Education, SIGCSE 2012, p. 663. ACM, New York (2012)
28. The European Parliament, the Council of the European Union: Directive 1999/93/EC of the European Parliament and of the Council of 13 on a Community Framework for Electronic Signatures, December 1999
29. The European Parliament, the Council of the European Union: Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 on Electronic Identification and Trust Services for Electronic Transactions in the Internal Market and Repealing Directive 1999/93/EC, July 2014
30. Zefferer, T., Kreuzhuber, S., Teufl, P.: Assessing the suitability of current smartphone platforms for mobile government. In: Kő, A., Leitner, C., Leitold, H., Prosser, A. (eds.) EDEM 2013 and EGOVIS 2013. LNCS, vol. 8061, pp. 125–139. Springer, Heidelberg (2013)
31. Zefferer, T., Krnjic, V.: Usability evaluation of electronic signature based e-Government solutions. In: Proceedings of the IADIS International Conference WWW/INTERNET 2012, pp. 227–234 (2012)

Author Index