

Adam Jatowt · Akira Maeda ·
Sue Yeon Syn (Eds.)

LNCS 11853

Digital Libraries at the Crossroads of Digital Information for the Future

21st International Conference
on Asia-Pacific Digital Libraries, ICADL 2019
Kuala Lumpur, Malaysia, November 4–7, 2019, Proceedings



Springer

Founding Editors

Gerhard Goos

Karlsruhe Institute of Technology, Karlsruhe, Germany

Juris Hartmanis

Cornell University, Ithaca, NY, USA

Editorial Board Members

Elisa Bertino

Purdue University, West Lafayette, IN, USA

Wen Gao

Peking University, Beijing, China

Bernhard Steffen

TU Dortmund University, Dortmund, Germany

Gerhard Woeginger 

RWTH Aachen, Aachen, Germany

Moti Yung

Columbia University, New York, NY, USA


More information about this series at <http://www.springer.com/series/7409>

Adam Jatowt · Akira Maeda ·
Sue Yeon Syn (Eds.)

Digital Libraries at the Crossroads of Digital Information for the Future

21st International Conference
on Asia-Pacific Digital Libraries, ICADL 2019
Kuala Lumpur, Malaysia, November 4–7, 2019
Proceedings

Editors

Adam Jatowt 
Kyoto University
Kyoto, Japan

Akira Maeda 
Ritsumeikan University
Kusatsu, Japan

Sue Yeon Syn 
The Catholic University of America
Washington, DC, USA

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Computer Science
ISBN 978-3-030-34057-5 ISBN 978-3-030-34058-2 (eBook)
<https://doi.org/10.1007/978-3-030-34058-2>

LNCS Sublibrary: SL3 – Information Systems and Applications, incl. Internet/Web, and HCI

© Springer Nature Switzerland AG 2019

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, expressed or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

We are happy and proud to deliver this volume containing papers accepted at the 21st International Conference on Asia-Pacific Digital Libraries (ICADL 2019; <http://icadl2019.org/>), which was held in Kuala Lumpur, Malaysia, during November 4–7, 2019. The ICADL conference invites research papers devoted to broad aspects of digital libraries and aims to bring together researchers, developers, and other experts in digital libraries and in related fields. This year the conference ran under the theme: “Digital Libraries at Cross-Roads of Digital Information for the Future – Connecting Data, Technologies and People in Different Domains.”

ICADL 2019 was co-located with the 9th Asia-Pacific Conference on Library Information Education and Practice (A-LIEP 2019; <http://aliep2019.org/>). ICADL and A-LIEP share some common research interests, for example, the focus on research data in various domains such as STEM, health information, humanities and social sciences, and their different perspectives such as data analytics and research data management. The annual meeting of Asia-Pacific chapter of iSchools (AP-iSchools; <https://ischools.org/>) was also co-located with ICADL 2019. Altogether, this year, ICADL was a part of a broader international forum organized to enable the meeting of professionals from various backgrounds, yet who share common research interests in digital information and digital libraries for the future.

In response to the conference call, 54 papers were submitted to the conference and each paper was reviewed by at least 3 Program Committee members. Based on the reviews and recommendations from the Program Committee, 13 long papers and 13 short papers were selected and included in the proceedings. Some of the papers were invited to be presented during the poster and demonstration session in order to encourage more cross-community discussions and demonstrate practical works.

We were honored to have three keynote speakers who delivered joint ICADL 2019 and A-LIEP 2019 plenary talks: Gobinda Chowdhury (Northumbria University, UK), Joyce Chao-Chen Chen (National Taiwan Normal University, Taiwan), and Abu Bakar Abdul Majeed (Universiti Teknologi MARA, Malaysia). The conference program also included a doctoral consortium and two workshops: Information Commons for Manga, Anime and Video Games (MAGIC), and the Workshop on Longevity and Preservation of Cultural Digital Assets.

The success of ICADL 2019 was the result of great effort from the team of many individuals. We would like to thank the Program Committee members for their effort in reviewing submitted contributions, as well as Sohami Zakaria, Shigeo Sugimoto, and Christopher Khoo (general chairs), Unmil Karadkar (poster and demonstration chair), Hao-ren Ke and Natalie Pang (workshop and tutorial chairs), Diljit Singh (publication chair), Wirapong Chansanam, Sonphan Choemprayong, Annika Hinze, Emi Ishita, Hao-ren Ke, Joon Lee, Miguel Nunes, Natalie Pang, Shigeo Sugimoto, Xiaoguang Wang, and Fred Y. Ye (doctoral consortium chairs), and the local organization team.

Finally, we express our gratitude to all the authors, presenters, and participants of the conference. We hope that you enjoy the conference proceedings.

November 2019

Adam Jatowt
Akira Maeda
Sue Yeon Syn

Organization

Organizing Committee

Organizing Committee (International Forum 2019)

Sohami Zakaria (General Chair)	Universiti Teknologi MARA, Malaysia
Shigeo Sugimoto	University of Tsukuba, Japan
Christopher Khoo	NTU, Singapore

ICADL Program Committee Chairs

Adam Jatowt	Kyoto University, Japan
Akira Maeda	Ritsumeikan University, Japan
Sue Yeon Syn	Catholic University of America, USA

Posters and Work-in-Progress Papers Chair

Unmil Karadkar	University of Graz, Austria
----------------	-----------------------------

Workshop Chairs

Natalie Pang	National University of Singapore, Singapore
Hao-ren Ke	National Taiwan Normal University, Taiwan

Publication Chair

Diljit Singh	Formerly University of Malaya, Malaysia
--------------	---

Doctoral Consortium Committee

Wirapong Chansanam	Khon Kaen University, Thailand
Sonphan Choemprayong	Chulalongkorn University, Thailand
Annika Hinze	University of Waikato, New Zealand
Emi Ishita	Kyushu University, Japan
Hao-ren Ke	National Taiwan Normal University, Taiwan
Joon Lee	Seoul National University, South Korea
Miguel Nunes	Sun Yat-sen University, China
Natalie Pang	National University of Singapore, Singapore
Shigeo Sugimoto	University of Tsukuba, Japan
Xiaoguang Wang	Wuhan University, China
Fred Y. Ye	Nanjing University, China

Program Committee

Trond Aalberg	Oslo Metropolitan University, Norway
David Bainbridge	University of Waikato, New Zealand
Biligsaikhan Batjargal	Research Organization of Science and Technology, Ritsumeikan University, Japan
Songphan Choemprayong	Chulalongkorn University, Thailand
Gobinda Chowdhury	Northumbria University, UK
Fabio Crestani	University of Lugano (USI), Switzerland
Sally Jo Cunningham	Waikato University, New Zealand
Milena Dobрева	UCL Qatar, Qatar
Schubert Foo	Nanyang Technological University, Singapore
Edward Fox	Virginia Tech, USA
Dion Goh	Nanyang Technological University, Singapore
Mohammed Hasanuzzaman	ADAPT Centre-Dublin, Ireland
Emi Ishita	Kyushu University, Japan
Adam Jatowt	Kyoto University, Japan
Unmil Karadkar	The University of Texas at Austin, USA
Makoto P. Kato	University of Tsukuba, Japan
Yukiko Kawai	Kyoto Sangyo University, Japan
Hao-Ren Ke	Graduate Institute of Library and Information Studies, National Taiwan Normal University, Taiwan
Yunhyong Kim	University of Glasgow, UK
Chern Li Liew	Victoria University of Wellington, New Zealand
Fernando Loizides	Cardiff University, UK
Akira Maeda	Ritsumeikan University, Japan
Tanja Merčun	University of Ljubljana, Slovenia
Atsuyuki Morishima	University of Tsukuba, Japan
David Nichols	University of Waikato, New Zealand
Chifumi Nishioka	Kyoto University, Japan
Hiroaki Ohshima	Graduate School of Applied Informatics, University of Hyogo, Japan
Gillian Oliver	Monash University, Australia
Wee-Hong Ong	Universiti Brunei Darussalam, Brunei
Natalie Pang	National University of Singapore, Singapore
Georgios Papaioannou	UCL Qatar, Qatar
Christos Papatheodorou	Ionian University, Greece
Edie Rasmussen	The University of British Columbia, Canada
Andreas Rauber	Vienna University of Technology, Austria
Ali Shiri	University of Alberta, Canada
Panote Siriiraya	Kyoto Institute of Technology, Japan
Armin Straube	UCL Qatar, Qatar
Shigeo Sugimoto	University of Tsukuba, Japan
Sue Yeon Syn	The Catholic University of America, USA
Giannis Tsakonas	University of Patras, Greece
Nicholas Vanderschantz	University of Waikato, New Zealand

Diane Velasquez	University of South Australia, Australia
Dan Wu	School of Information Management, Wuhan University, China
Takehiro Yamamoto	University of Hyogo, Japan
Yusuke Yamamoto	Shizuoka University, Japan
Sohaimi Zakaria	Universiti Teknologi MARA, Malaysia
Maja Žumer	University of Ljubljana, Slovenia

Additional Reviewers

Banerjee, Bipasha
Jude, Palakh
Karajeh, Ola
Landoni, Monica
Li, Kangying
Papachristopoulos, Leonidas
Song, Yuting
Takaku, Masao
Torfi, Amirsina

Contents

Text Classification

- PCGAN-CHAR: Progressively Trained Classifier Generative Adversarial Networks for Classification of Noisy Handwritten Bangla Characters. 3
Qun Liu, Edward Collier, and Supratik Mukhopadhyay
- Weakly-Supervised Neural Categorization of Wikipedia Articles. 16
Xingyu Chen and Mizuho Iwaihara
- Supervised Classifiers to Identify Hate Speech on English and Spanish Tweets. 23
Sattam Almatarneh, Pablo Gamallo, Francisco J. Ribadas Pena, and Alexey Alexeev

Altmetrics

- An Overview of Altmetrics Research: A Typology Approach. 33
Han Zheng, Xiaoyu Chen, and Xu Duan
- User Motivation Classification and Comparison of Tweets Mentioning Research Articles in the Fields of Medicine, Chemistry and Environmental Science. 40
Mahalakshmi Suresh Kumar, Shreya Gupta, Subashini Baskaran, and Jin-Cheon Na
- Understanding the Twitter Usage of Science Citation Index (SCI) Journals. . . 54
Aravind Sesagiri Raamkumar, Mojisola Erdt, Harsha Vijayakumar, Aarthy Nagarajan, and Yin-Leng Theng

Scholarly Data Analysis and Recommendation

- Research Paper Recommender System with Serendipity Using Tweets vs. Diversification 63
Chifumi Nishioka, Jörn Hauke, and Ansgar Scherp
- The Rise and Rise of Interdisciplinary Research: Understanding the Interaction Dynamics of Three Major Fields – Physics, Mathematics and Computer Science 71
Rima Hazra, Mayank Singh, Pawan Goyal, Bibhas Adhikari, and Animesh Mukherjee

The Evolving Ecosystem of Predatory Journals: A Case Study
in Indian Perspective 78
Naman Jain and Mayank Singh

Metadata and Entities

Identifying and Linking Entities of Multimedia Franchise on Manga,
Anime and Video Game from Wikipedia 95
*Kosuke Oishi, Tetsuya Mihara, Mitsuharu Nagamori,
and Shigeo Sugimoto*

Impact of OCR Quality on Named Entity Linking. 102
*Elvys Linhares Pontes, Ahmed Hamdi, Nicolas Sidere,
and Antoine Doucet*

Authoring Formats and Their Extensibility for Application Profiles 116
*Nishad Thalpath, Mitsuharu Nagamori, Tetsuo Sakaguchi,
and Shigeo Sugimoto*

Digital Libraries and Digital Archives Management

Connecting Data for Digital Libraries: The Library, the Dictionary
and the Corpus 125
Maciej Ogrodniczuk and Włodzimierz Gruszczyński

Modernisation of a 12 Years Old Digital Archive:
Experiences and Lessons Learned 139
Ilvio Bruder, Martin Duffer, and Andreas Heuer

Investigating the Feasibility of Digital Repositories in Private Clouds 151
Mushashu Lumpa and Hussein Suleman

Multimedia Processing

Towards Automatic Cataloging of Image and Textual Collections
with Wikipedia 167
Tokinori Suzuki, Daisuke Ikeda, Petra Galuščáková, and Douglas Oard

Automatic Semantic Segmentation and Annotation of MOOC
Lecture Videos 181
Ananda Das and Partha Pratim Das

Play, Pause, and Skip: Touch Screen Media Player Interaction
on the Move. 189
Nicholas Vanderschantz and Zhiqin Yang

Search Engines

- Towards a Story-Based Search Engine for Arabic-Speaking Children. 205
*Rawan Al-Matham, Wed Ateeq, Reem Alotaibi, Sarah Alabdulkarim,
 and Hend Al-Khalifa*
- Metadata-Based Automatic Query Suggestion in Digital Library
 Using Pattern Mining 213
Susmita Sadhu and Plaban Kumar Bhowmick
- Investigating the Effectiveness of Client-Side Search/Browse
 Without a Network Connection. 227
Hussein Suleman
- SchenQL: A Concept of a Domain-Specific Query Language
 on Bibliographic Metadata 239
Christin Katharina Kreutz, Michael Wolz, and Ralf Schenkel

Information Extraction

- Formalising Document Structure and Automatically Recognising
 Document Elements: A Case Study on Automobile Repair Manuals 249
Hodai Sugino, Rei Miyata, and Satoshi Sato
- Weakly-Supervised Relation Extraction in Legal Knowledge Bases. 263
Haojie Huang, Raymond K. Wong, Baoxiang Du, and Hae Jin Han
- Social Media Aware Virtual Editions for the Book of Disquiet. 271
Duarte Oliveira, António Rito Silva, and Manuel Portela
- Language Identification for South African Bantu Languages Using
 Rank Order Statistics. 283
Meluleki Dube and Hussein Suleman

Posters

- Creating a Security Methods Taxonomy for Information
 Resource Management. 293
Yejun Wu and Fansong Meng
- A Dynamic Microtask Approach to Collecting and Organizing
 Citizens' Opinions. 298
*Masaki Matsubara, Yuhei Matsuda, Ryohei Kuzumi, Masanori Koizumi,
 and Atsuyuki Morishima*

**Sentiment Analysis of Tweets Mentioning Research Articles
in Medicine and Psychiatry Disciplines 303**
*Sampathkumar Kuppan Bharathwaj, Jin-Cheon Na, Babu Sangeetha,
and Eswaran Sarathkumar*

Measurement and Visualization of IIF Image Usages 308
Chifumi Nishioka and Kiyonori Nagasaki

Improving OCR for Historical Documents by Modeling Image Distortion . . . 312
*Keiya Maekawa, Yoichi Tomiura, Satoshi Fukuda, Emi Ishita,
and Hideaki Uchiyama*

Author Index 317

Text Classification



PCGAN-CHAR: Progressively Trained Classifier Generative Adversarial Networks for Classification of Noisy Handwritten Bangla Characters

Qun Liu^(✉), Edward Collier, and Supratik Mukhopadhyay

Louisiana State University, Baton Rouge, LA 70803, USA
{qliu14,ecol128}@lsu.edu
supratik@csc.lsu.edu

Abstract. Due to the sparsity of features, noise has proven to be a great inhibitor in the classification of handwritten characters. To combat this, most techniques perform denoising of the data before classification. In this paper, we consolidate the approach by training an all-in-one model that is able to classify even noisy characters. For classification, we progressively train a classifier generative adversarial network on the characters from low to high resolution. We show that by learning the features at each resolution independently a trained model is able to accurately classify characters even in the presence of noise. We experimentally demonstrate the effectiveness of our approach by classifying noisy versions of MNIST [13], handwritten Bangla Numeral, and Basic Character datasets [5, 6].

Keywords: Progressively training · General adversarial networks · Classification · Noisy characters · Handwritten bangla

1 Introduction

Early work in neural networks focused on classification of handwritten characters [16, 17]. Since then, there has been a lot of research on character recognition. While in many cases, text processing deals directly with the character strings themselves, there are a growing number of use cases for recognizing characters and text in physical real world prints and documents. This includes processing receipts and bank statements, transcribing books and medical prescriptions, or translating text. Aside from the large amounts of computer generated text, there are vast quantities of scanned handwritten text that can be processed. Such text is generally collected as images, which invariably introduces some noise (e.g., damaged documents, noise added due to camera motion, etc.). While computer generated text classification might be more stable to noise, recognition of handwritten text breaks down with the introduction of noise.

In this work, we build on recent work in adversarial training [14] to improve on the state-of-the-art in representing sparse features [3, 14, 23]. We define sparse representations as noisy, generally compact, representations of signals [7, 12]. This is the case for many real world images which contain various sources of noise that can distort their true representation. Such noise can easily reduce the quality of classifications and challenge the power of classifiers [19, 22]. Most algorithms include denoising step for the images before classification [13], while our approach can directly classify without denoising step due to progressively learn features at increasing resolutions to accurately classify the noisy digits/characters.

Figure 1 shows the architecture of our approach. We utilized the progressive technique which is a newly proposed method [14], for training *Auxiliary Classifier Generative Adversarial Networks (ACGAN)* that has been attractive due to its ability to improve and stabilize the network. It facilitates networks to learn features in a generic to specific manner as the input progresses down the model [23]. Low resolution features are more resistant to noise due to their generic nature. By individually learning representations at each resolution, our method is able to leverage the noise-resistant generic features to make more accurate and better predictions for noisy handwritten characters to achieve state-of-the-art performance.

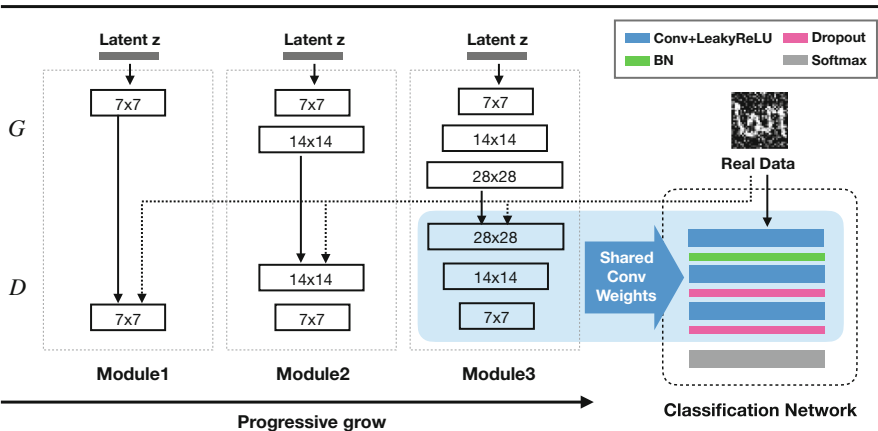


Fig. 1. Overview of our proposed progressively trained classifier generative adversarial networks (PCGAN-CHAR) architecture.

In general, our framework uses a generative adversarial network (GAN) [9] as a basic component for its noise-resilient ability and the discriminative power of its discriminator for classification. GANs contain two competing networks, a generator and a discriminator [9], playing a minmax game. The discriminator [9] tries to discern real samples from fake ones generated by the generator in an attempt to fool the discriminator. Because of this behavior, as one network tries to minimize its own loss, this in turn maximizes the other network’s loss.

Generators from GANs have been shown to generate outputs that are almost indiscernible from real samples, while the discriminator trained by a GAN has more discriminative power with respect to classification being exposed to both real and fake data, where much of the fake data contains some noise.

We used the discriminator in the Classifier GAN for our Classification Network for the noisy handwritten characters, but to make it more robust to noise and resolution we novelly adopted the innovative GAN training technique, Progressive growing [14], to our Classifier GAN. In progressive growing each layer of the GAN is trained individually on increasing resolutions. This allows each feature to specialize and simplifies the problem at the individual layers.

To the best of our knowledge, this is the first work that designs a Progressive trained Classifier GAN (PCGAN) for retaining the noise-resistant discriminator in a classifier GAN for robust classification in noisy settings. This paper makes the following contributions.

- It presents a novel robust noise-resilient classification framework using progressively trained classifier general adversarial networks.
- The proposed classification framework can directly classify raw noisy data without any preprocessing steps that include complex techniques such as denoising or reconstruction.
- It experimentally demonstrates the effectiveness of the framework on the Noisy Bangla Numeral, the Noisy Bangla Characters, and the Noisy MNIST benchmark datasets.

2 Related Work

Handwritten character based datasets have become benchmarks in computer vision research. Handwritten characters contain sparse representations, or features, while also containing significant amounts of noise [3]. Early work on classification of handwritten characters focused on dimensionality reduction and denoising. This includes the use of quadtrees [1, 3, 18] and intermediate layers of Convolutional Neural Networks for representations and Deep Belief Networks (DBN) for denoising [3].

Researchers have tried a variety of methods to solve the noisy character classification problem. For example, multi-stage approaches have used chain code histogram features to discriminate classes in [5]. Similar to this work, increasing resolutions are used to assist in classification. Other multistage approaches have used modified quadratic discriminant function (MQDF) and gradients from neural networks to classify characters from many classes [6].

Convolutional neural networks (CNNs) [2, 4, 8, 10, 15] have been widely used in image processing, and are increasingly being used in character and text classification [13]. The performance of CNNs can be greatly affected by blurry images, where noise increases the separation between the output and the ground truth in feature space [21]. While Euclidean distance has shown to be a decent method for measuring the closeness between two images in feature space, it is difficult to measure the sharpness and quality of images. Through adversarial training,

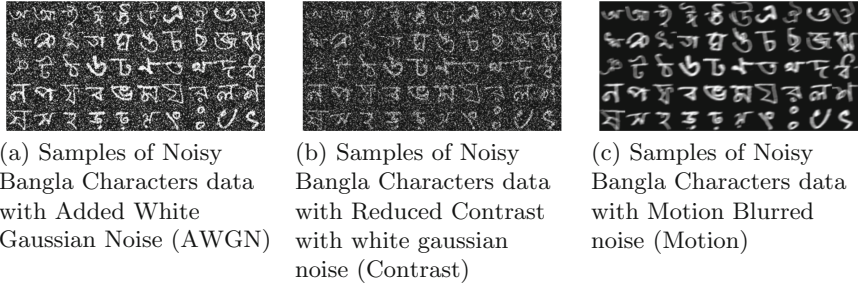


Fig. 2. Sample data for different types of added noise for noisy Bangla Characters. Three types of noisy data, added white Gaussian Noise, Reduced Contrast with white gaussian noise, motion Blurred noise, shown above from top to bottom.

Generative adversarial networks (GANs) can both imagine structure where there is none, to produce sharp images, and discern real from fake [9]. Progressively growing GANs are an improvement on the GAN architecture that can produce sharp realistic images [14]. This innovation is important for GANs operate on high resolution input. When handling high resolution data it can be too easy for the GAN to discriminate between the fake, generally low resolution imagery, and the real, which are high resolution. Progressively grown GANs learn the resolutions at each layer in isolation by training each layer almost independently, before adding the next layer and training again. This process utilizes transfer learning and the generic to specific learning behaviour exhibited in neural networks [23]. Auxiliary Classifier GANs (ACGAN) [20] has introduced class labels in GANs by adding an auxiliary classifier which leveraged model in its prior, their research focused on image synthesis tasks and has shown better performance. The focus of our paper is classification of noisy handwritten (Bangla) characters.

3 Methodology

In this section, we present an overview of our proposed method. We separate our methodology into three subsections; Generative Adversarial Networks, Progressively Trained Classifier GAN, and Classification Network. In the Generative Adversarial Network section, we outline the formulation of our GAN objective function. We describe the progressive growing structure of the GAN in Sect. 3.2, along with a general training algorithm. Details about the Classification Network are provided in Sect. 3.3.

3.1 Generative Adversarial Network

A Generative Adversarial Network (GAN) [9] comprises of two networks: a generator and a discriminator. The generator G takes as input a random noise vector z and outputs a fake image $G(z)$. It learns a mapping function, $z \rightarrow y$, between

the latent space z and the feature space defined by the task y . The discriminator D takes as input x either a real image or a fake image produced by G to calculate class probabilities. It attempts to learn a mapping between feature space y and a discriminatory space q , $y \rightarrow q$. The goal is to map all true y to the positive class while all fake y_f to the negative class.

To learn these mappings the generator G plays with the discriminator D a two-player min/max game, $\min_G \max_D L(G, D)$, with a loss $L(G, D)$. The objective function [9] is defined as follows.

$$\begin{aligned} \min_G \max_D L(G, D) = & \mathbb{E}_y[\log D(y)] \\ & + \mathbb{E}_{x,z}[\log(1 - D(x|G(x|z)))] \end{aligned} \quad (1)$$

where the discriminator D will try to maximize the log-likelihood which is the first term, while the generator G try to minimize the second term.

The classifier GAN [20] takes a class label l and a random noise vector z to generate fake images X_f through its generator G , denoted as $G(l, z)$. The real images X_r are the training images in the noisy dataset under consideration. The objective function of the classification GAN comprises of two components each involving two log-likelihood L_1 and L_2 . The log-likelihood L_1 involves the conditional probability distribution $P(guess | x)$ where the input x can be a fake image X_f (generated by the generator) or a real image X_r (a training image from the noisy dataset under consideration) and $guess$ can take two values *fake* or *real*. Formally [20],

$$L_1 = \mathbb{E}[\log P(real | X_r)] + \mathbb{E}[\log P(fake | X_f)]. \quad (2)$$

The log-likelihood L_2 involves the conditional probability distribution $P(l | x)$ where l is the class label of x and x is a real image X_r or a fake image X_f . Formally,

$$L_2 = \mathbb{E}[\log P(l | X_r)] + \mathbb{E}[\log P(l | X_f)]. \quad (3)$$

The generator G will try to maximize $L_2 - L_1$, but the discriminator D will try to maximize $L_1 + L_2$.

3.2 Progressively Trained Classifier GAN

Progressive growing is a recent development from [14] that uses transfer learning to improve the quality of the learned models. Training is performed individually for the layers of the generator G and the discriminator D . New mirroring layers are added to G and D before a new training iteration is run. This increases the spatial resolution of the output image for each layer progressively added. Layers in G and D become more specialized to spatial resolution, resulting in them learning finer features. In addition to making the layers more specialized, progressively growing also simplifies the problem at each layer. This creates a more stable generative model with finer outputs.

Deep neural networks learn features in a low resolution to high resolution manner, or generic to specific. Progressive growing takes advantage of this behavior by transferring the weights learned from all previous training iterations to

identical layers for the next step. Each training iteration then only contains one untrained layer, the newest layer, allowing for features at each resolution to be learned independently and in relative isolation from the other layers. The discriminator grows in parallel with the generator with each layer learning to discriminate specific resolutions.

Most progressive GANs are designed with generative tasks in mind, putting a focus on the generator. In our work, however, we use the well trained discriminator that is the end result of progressive training. Each layer of the discriminator specializes at specific resolutions allowing it to learn more fine-grain features. Additionally, the discriminator has seen a wide range of samples produced by the generator as it learns; from noisy to sharp. We call this discriminator “well trained” because it is more robust to noise and sparse features, characteristics inherent in handwritten characters.

We show the general training algorithm for our proposed framework for progressively training the classifier GAN in Algorithm 1 for a given number of progressive stages. As seen in the Algorithm, we first initialize the progressive stages (indicated as modules in Fig. 1), for learning essential features at different resolutions. The input resolution for the discriminator increases from 7×7 for the first module to 28×28 for the third module as seen in Fig. 1. After initializing a module, our algorithm begins training the GAN using the normal training procedure. Since our GAN is used for classification purposes, unlike other variants of GANs used to synthesize high quality images, which mainly focus on the generator, we focus on the discriminator and aim to improve its classification ability. Similar to ACGAN [20], we add an auxiliary classifier to the discriminator to compute class labels, apart from using a binary classifier for discerning if an image is fake or real. The latter classifier corresponds to the loss $\mathcal{L}^{discern}$ as shown in Algorithm 1; this loss is used to enhance the robustness of the discriminator by learning better representations of variations within a class [11]. After a module is trained through epochs, the weights in its generator and discriminator are transferred to the next module (as seen in Fig. 1, the weights from $Module_i$ is transferred to $Module_{i+1}$, with $i \geq 1$, with the number of layers increasing as we progress from the i th module to the $i + 1$ th module for augmenting resolution). This method of progressive training continues until training for the last module has converged.

3.3 Classification Network

After being progressively trained, the discriminator of the classifier GAN has learned the input space in such a way that the lower layers specialize on low resolutions while the higher layers specialize on high resolutions. Progressive training results in a stabilized discriminator that has learned the essential features from noisy data at multiple resolutions, resulting in better classification performance. As we can see in Fig. 1, as we go from the i th module to the $i + 1$ th module, the number of layers increases. The input resolution for discriminator increases from 7×7 in the first module to 28×28 in the third module. The weights of trained discriminator in Module 3 are transferred to the classification network, which is a convolutional neural network as described below. To

Algorithm 1. General Training Algorithm

```

1 for number of modules do
2   Initialize(module)
3   for epoch=1,2,...,K do
4     for number of batches do
5        $B_r \leftarrow$  Sample a batch of  $n$  real images with labels
6        $B_z \leftarrow$  Sample a batch of  $n$  random vectors
7        $B_l \leftarrow$  Sample a batch of  $n$  random labels
8       Update the parameters in the discriminator regarding gradients,
          
$$\nabla_{\theta_d} \frac{1}{2n} \sum_{r \in B_r, z \in G(B_z, B_l)} \mathcal{L}_{discern}(r, z) + \frac{1}{n} \sum_{r \in B_r} \mathcal{L}_{class}(r)$$

           $B_z \leftarrow$  Sample a batch of  $2n$  random vectors
           $B_l \leftarrow$  Sample a batch of  $2n$  random labels Set discriminator trainable to false, update the parameters in the generator regarding gradients,
          
$$\nabla_{\theta_g} \frac{1}{2n} \sum_{z, z' \in G(B_z, B_l)} \mathcal{L}_{discern}(z, z') + \frac{1}{2n} \sum_{z, z' \in G(B_z, B_l)} \mathcal{L}_{class}(z, z')$$

9     end
10  end
11  Transfer weights to Next(module)
12 end

```

further improve the performance of the classification network, we finetuned the softmax layer. The details of the classification network are shown in Fig. 1. It includes three convolutional layers, each with LeakyReLU activation function. Two dropout layers and a batch normalization layer are added to the classification network as showed in Fig. 1. Finally, a softmax layer added as the last layer for classification.

4 Experimental Evaluation

The experiments mainly focus on Indian handwritten digits and characters datasets [13], namely, Noisy Bangla Numeral and Noisy Bangla Characters, are publicly available datasets we downloaded from online¹, it provided a training

¹ https://en.wikipedia.org/wiki/List_of_datasets_for_machinelearning_research#Handwriting_and_character_recognition.

dataset and a test dataset. For a comparative study, we also conducted experiments on a noisy version of a commonly used handwritten digits dataset, Noisy MNIST Dataset [13]. The original non-noisy versions of the Bangla Numeral and Character datasets are from [5,6]. We consider three different versions of each dataset, the first with Added White Gaussian Noise (AWGN), the second with Reduced Contrast with white Gaussian noise (Contrast), and the third with Motion Blurred noise (Motion). Sample data from each of the three different versions of the noisy Bangla character dataset are shown in Fig. 2.

4.1 Datasets

Noisy Bangla Numeral has three different versions, each with different type of noise added, AWGN, Contrast, and Motion. For each version, there are 10 classes of Bangla Numerals with a total of 23330 black and white images with image size 32×32 .

Noisy Bangla Characters contains 76000 black and white images for 50 classes of Bangla Characters with image size 32×32 , in each version. There are three different versions one for each type of added noise as stated above.

Noisy MNIST Dataset is the same as the original MNIST dataset except for added noise. Again, there are three different versions, one for each of the three types of noise considered. Each version contains 10 classes with a total of 70000 black and white images with image size 28×28 .

4.2 Implementation Details

We used an auxiliary classifier GAN (ACGAN) [20] as the classifier GAN in our framework. We considered 28×28 (the resolution of the noisy MNIST dataset) as the input resolution for our framework. The images in the noisy Bangla dataset were resized to 28×28 before being input to our framework.

The architecture of generators is the reverse of that of the discriminators (see Fig. 1). Specifically, for the generators, we used a dense layer to transform an input (latent z) to a format that corresponds to the input of the convolutional transpose layers for generating multiresolution images. For *Module*₁, after the first convolutional transpose layer, we used a batch normalization layer and another convolutional transpose layer to transform the feature maps to an output image with resolution 7×7 using a filter with kernel size 1×1 . A similar configuration is used generating output images with resolutions 14×14 and 28×28 respectively for *Module*₂ and *Module*₃ during the progressive training procedure. For the discriminator, during progressive training, we started from the input resolution of 7×7 (*Module*₁). The real images (i.e., the training images from the noisy dataset under consideration) are downscaled to 7×7 . The downscaled real images are combined with fake images produced by the generator and are fed to the discriminator for feature extraction and classification. The discriminator not only computes the class label but also discerns fake images from real ones.

Similar downscaling and combination operations are performed before feeding inputs to the discriminators for the second and the third modules.

During the progressive training, the GAN learns, in addition to other features, low resolution features that are more tolerant to noise, being generic in nature. While the GAN learns low resolution as well as noisy features, it never learns to denoise an image. At no point, does our framework produce a denoised image or learns the representation of a denoised image. Instead, low-resolution features that are not disrupted by noise are learned through progressive training. By individually learning representations at each resolution, our method is able to leverage the noise-resistant generic low resolution features to provide better classification performance even for noisy character data. This produces a robust discriminator that can classify noisy handwritten characters. Thus, in our framework, there is no preprocessing step that denoises the input explicitly or implicitly.

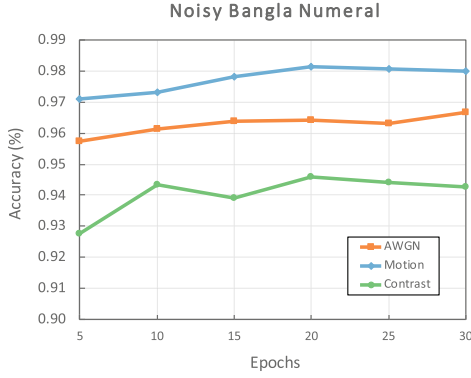
4.3 Evaluation

We have evaluated our framework on Noisy Bangla Numeral, Noisy Bangla Characters, and Noisy MNIST datasets with respect to classification accuracy as shown in Tables 1, 2, and 3, respectively. Figure 3 shows how the accuracy varies with the number of training epochs for Noisy Bangla Numeral (Fig. 3(a)), Noisy Bangla Characters (Fig. 3(b)), and Noisy MNIST (Fig. 3(c)) datasets.

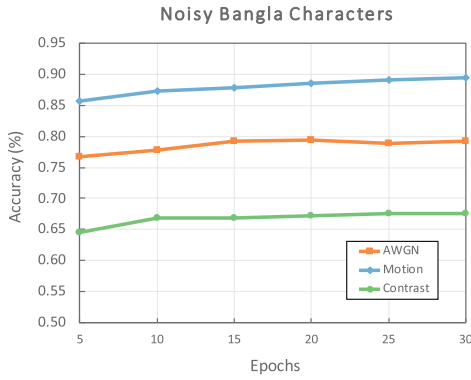
For each version (corresponding to each of the three types of added noise) of each dataset, we used the provided training and testing dataset splits [13]. As we can see in Table 1, for the Noisy Bangla Numeral dataset, our approach achieved the best performance (in terms of accuracy): **96.68%** on AWGN noise, **98.18%** on Motion noise, and **94.60%** on Contrast noise, which surpassed the second best ones by **1.22%**, **1.13%**, **1.75%**, respectively. In Fig. 3(a), one can see that the classification accuracy of our framework remains almost the same on AWGN noise with increasing number of epochs. For the same dataset, but with added Motion noise, the classification accuracy initially increases before stabilizing. In case of added Contrast noise, for the same dataset, the classification accuracy remains relatively unstable with increasing number of epochs.

In Table 2, for Noisy Bangla Characters, our framework obtained better performance than the state-of-the-art in the case of added AWGN noise (**79.85%**) and added Motion noise (**89.54%**) surpassing the second best by **3.11%** and **5.95%**, respectively. In the case of added Contrast noise, our framework achieved an accuracy of 68.41%, which is (**-1.25%**) slightly less than the state-of-the-art 69.66%.

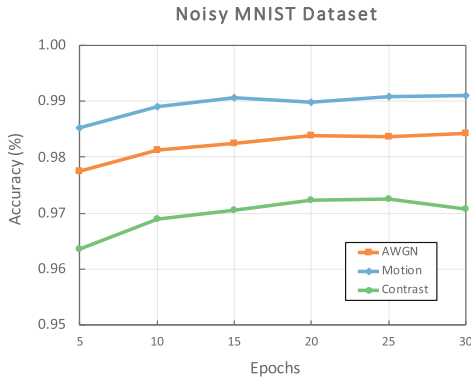
The performances of all methods on the Noisy Bangla Characters are much worse than those on the Noisy Bangla Numeral and the Noisy MNIST Datasets. Noisy Bangla Characters are relatively harder to classify than the other two datasets as this dataset has 50 classes versus 10 classes for each of the other two datasets. Additionally, among the three types of added noises, all methods have relatively poor performances on Contrast noise compared to their performances on AWGN noise and Motion noise. In Fig. 3(b), we can see that the classification



(a) Classification accuracy on three types of Noisy Bangla Numeral.



(b) Classification accuracy on three types of Noisy Bangla Characters.



(c) Classification accuracy on three types of Noisy MNIST Dataset.

Fig. 3. The classification accuracy of our approach. Our approach has been evaluated on datasets of Noisy Bangla Numeral, Noisy Bangla Characters, and Noisy MNIST with three types of added noise, AWGN, Motion, Contrast.

Table 1. Comparison of classification accuracy (%) on three types of Noisy Bangla Numeral

Methods	AWGN	Motion	Contrast
Basu et al. [3]	91.34	92.66	87.31
Dropconnect [13]	91.18	97.05	85.79
Karki et al. (w/o Saliency) [13]	95.08	94.88	92.60
Karki et al. (Saliency) [13]	95.46	95.04	92.85
PCGAN-CHAR (Ours)	96.68	98.18	94.60

Table 2. Comparison of classification accuracy (%) on three types of Noisy Bangla characters

Methods	AWGN	Motion	Contrast
Basu et al. [3]	57.31	58.80	46.63
Dropconnect [13]	61.14	83.59	48.07
Karki et al. (w/o Saliency) [13]	70.64	74.36	58.89
Karki et al. (Saliency) [13]	76.74	77.22	69.66
PCGAN-CHAR (Ours)	79.85	89.54	68.41

Table 3. Comparison of classification accuracy (%) on three types of Noisy MNIST dataset

Methods	AWGN	Motion	Contrast
Basu et al. [3]	90.07	97.40	92.16
Dropconnect [13]	96.02	98.58	93.24
Karki et al. (Saliency) [13]	97.62	97.20	95.04
PCGAN-CHAR (Ours)	98.43	99.20	97.25

accuracy of our framework remains relatively stable with increasing number of epochs on all the three types of added noise.

We also conducted experiments on the Noisy MNIST Dataset. The classification accuracies are shown in Table 3. In Table 3, it can be seen that the accuracy obtained by our framework exceeds the state-of-the-art by **0.81%** in the case of added AWGN noise, by **0.62%** in the case of added Motion noise, and by **2.21%** in the case of added Contrast noise, yielding best classification accuracies of **98.43%**, **99.20%**, and **97.25%**, respectively. In Fig. 3 for the Noisy MNIST Dataset, we can see the classification accuracies of our framework already surpassed the state-of-the-art after 5 epochs for all the three types of added noise.

To understand the statistical significance of the performance improvements obtained by our framework over [13], we used McNemar’s test (since our framework and [13] had same test datasets). Following are the results of the McNemar’s tests.

For noisy Bangla numeral with AWGN noise added: $\chi^2 = 47.02$, $df = 1$, $p < 7.025e - 12$; with reduced contrast and white Gaussian noise: $\chi^2 = 68.014$, $df = 1$, $p < 2.2e - 16$; here df represents degrees of freedom.

For noisy Bangla characters with added AWGN: $\chi^2 = 398$, $df = 1$, $p < 2.2e - 16$.

For noisy MNIST with added AWGN: $\chi^2 = 79.012$, $df = 1$, $p < 2.2e - 16$; with reduced contrast and white Gaussian noise: $\chi^2 = 219$, $df = 1$, $p < 2.2e - 16$.

Based on the results of the McNemar tests, the improvements obtained over [13], even in the case of AWGN and contrast variations are statistically significant.

The discriminator in our framework is trained to learn representations progressively from lower resolution to higher. Each layer of the discriminator specializes at specific resolutions allowing it to learn more fine-grain features. Lower resolution features are more resistant to noise due to their generic nature. Since the discriminator in our framework has been trained with a combination of fake images produced by the generator and real images belonging to the noisy dataset under consideration, it learned better representations of the variability within the classes. This explains the robustness of our framework to noise and sparse features.

5 Conclusion

In this paper, we presented a novel robust noise-resilient classification framework for noisy handwritten (Bangla) characters using progressively trained classification general adversarial networks. The proposed classification framework can directly classify raw noisy data without any preprocessing. We experimentally demonstrated the effectiveness of the framework on the Noisy Bangla Numeral, the Noisy Bangla Basic Characters, and the Noisy MNIST benchmark datasets.

References

1. Aref, W.G., Samet, H.: Decomposing a window into maximal quadtree blocks. *Acta Inf.* **30**(5), 425–439 (1993)
2. Basu, S., Ganguly, S., Mukhopadhyay, S., DiBiano, R., Karki, M., Nemani, R.R.: DeepSAT: A learning framework for satellite imagery. In: *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*, Bellevue, WA, USA, 3–6 November 2015
3. Basu, S., et al.: Learning sparse feature representations using probabilistic quadtrees and deep belief nets. *Neural Process. Lett.* **45**(3), 855–867 (2017)
4. Basu, S., et al.: A theoretical analysis of deep neural networks for texture classification. In: *2016 International Joint Conference on Neural Networks, IJCNN 2016*, Vancouver, BC, Canada, 24–29 July 2016, pp. 992–999 (2016)
5. Bhattacharya, U., Chaudhuri, B.B.: Handwritten numeral databases of Indian scripts and multistage recognition of mixed numerals. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(3), 444–457 (2009)

6. Bhattacharya, U., Shridhar, M., Parui, S.K., Sen, P., Chaudhuri, B.: Offline recognition of handwritten bangla characters: an efficient two-stage approach. *Pattern Anal. Appl.* **15**(4), 445–458 (2012)
7. Boureau, Y.L., Cun, Y.L., et al.: Sparse feature learning for deep belief networks. In: *Advances in Neural Information Processing Systems*, pp. 1185–1192 (2008)
8. Collier, E., DiBiano, R., Mukhopadhyay, S.: CactusNets: Layer applicability as a metric for transfer learning. In: *2018 International Joint Conference on Neural Networks, IJCNN 2018, Rio de Janeiro, Brazil, 8–13 July 2018*, pp. 1–8 (2018)
9. Goodfellow, I., et al.: Generative adversarial nets. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (eds.) *Advances in Neural Information Processing Systems 27*, pp. 2672–2680. Curran Associates, Inc. (2014). <http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf>
10. Hariharan, B., Arbeláez, P., Girshick, R., Malik, J.: Hypercolumns for object segmentation and fine-grained localization. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 447–456 (2015)
11. Hariharan, B., Girshick, R.: Low-shot visual recognition by shrinking and hallucinating features. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3018–3027 (2017)
12. Huang, K., Aviyente, S.: Sparse representation for signal classification. *NIPS* **19**, 609–616 (2006)
13. Karki, M., Liu, Q., DiBiano, R., Basu, S., Mukhopadhyay, S.: Pixel-level reconstruction and classification for noisy handwritten bangla characters. In: *16th International Conference on Frontiers in Handwriting Recognition, ICFHR 2018, Niagara Falls, NY, USA, 5–8 August 2018*, pp. 511–516 (2018). <https://doi.org/10.1109/ICFHR-2018.2018.00095>
14. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of GANs for improved quality, stability, and variation. *CoRR* abs/1710.10196 (2017). <http://arxiv.org/abs/1710.10196>
15. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*, pp. 1097–1105 (2012)
16. LeCun, Y., et al.: Backpropagation applied to handwritten zip code recognition. *Neural Comput.* **1**(4), 541–551 (1989)
17. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proc. IEEE* **86**(11), 2278–2324 (1998)
18. Markas, T., Reif, J.: Quad tree structures for image compression applications. *Inf. Process. Manage.* **28**(6), 707–721 (1992)
19. Nguyen, A., Yosinski, J., Clune, J.: Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 427–436 (2015)
20. Odena, A., Olah, C., Shlens, J.: Conditional image synthesis with auxiliary classifier GANs. In: *Proceedings of the 34th International Conference on Machine Learning*, vol. 70, pp. 2642–2651. JMLR. org (2017)
21. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: feature learning by inpainting. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2536–2544 (2016)
22. Su, J., Vargas, D.V., Sakurai, K.: One pixel attack for fooling deep neural networks. In: *Arxiv* (2018)
23. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? In: *Advances in Neural Information Processing Systems*, pp. 3320–3328 (2014)



Weakly-Supervised Neural Categorization of Wikipedia Articles

Xingyu Chen and Mizuho Iwaihara^(✉)

Graduate School of Information, Production and Systems, Waseda University,
Kitakyushu, Japan
chenxingyu@akane.waseda.jp, iwaihara@waseda.jp

Abstract. Deep neural models are gaining increasing popularity for many NLP tasks, due to their strong expressive power and less requirement for feature engineering. Neural models often need a large amount of labeled training documents. However, one category of Wikipedia does not contain enough articles for training. Weakly-supervised neural document classification can deal with situations even when only a small labeled document set is given. However, these RNN-based approaches often fail on long documents such as Wikipedia articles, due to hardness to retain memories on important parts of a long document. To overcome these challenges, we propose a text summarization method called WS-Rank, which extracts key sentences of documents with weighting based on class-related keywords and sentence positions in documents. After applying our WS-Rank to training and test documents to summarize then into key sentences, weakly-supervised neural classification shows remarkable improvement on classification results.

Keywords: Weakly-supervised learning · Text classification · Wikipedia category · Hierarchical category structure · Neural classification

1 Introduction

Although deep learning approach has been showing effectiveness on various NLP tasks such as text classification [7, 10, 14–16], necessity of a large volume of labeled training corpus is still a significant issue. Meng et al. [10] proposed a weakly supervised neural model, called WSTC, which utilizes CNN [5] (Convolutional Neural Networks) and HAN [15] (Hierarchical Attention Networks) on text classification, where effective models can be learned even when only a small set of labeled documents is available. However, these RNN-based approaches often face the problem of document length. The length of practical documents, such as Wikipedia articles, often exceed the learning ability of RNNs. In this paper, we discuss summarizing classifying documents by sentence extraction. After document summarization, reduced documents can be effectively classified by WSTC.

One simple approach for document summarization is to extract leading k sentences of one document. However, keyphrases that represent a document class may not appear in the leading sentences. Among graph-based extractive text summarizations [3, 11] and phrase extraction [6], TextRank [9] is still a representative text summarization

algorithm which can extract key sentences from a single document without any supervising information. To fit for our case, we propose WS-Rank (weakly-supervised TextRank) to promote keywords that well represent each class. We apply WS-Rank to both seed and test documents for WSTC.

Our contributions are summarized into two points: (1) Through experiments we clarify that existing weakly-supervised neural model WSTC [10] is not effective for long document classification. (2) We propose weakly-supervised TextRank for extracting key sentences from one document and combines it with the WSTC model. Through experiments we show our model outperforms the baseline WSTC models.

The rest of this paper is organized as follows. Section 2 covers related work. Section 3 introduces WS-Rank and its combination with WSTC. Section 4 presents evaluation results. Section 5 concludes this paper.

2 Related Work

There has been a large amount of research on neural text classification. Pengfei [12] applied RNN model for text classification. Lai et al. [8] proposed Recurrent Convolutional Neural Networks model which combine CNN and RNN model for text classification. Kim [5] and Zhang [16] applied CNN model for text classification and find that CNN model work well for text classification problem. Yang et al. [15] proposed Hierarchical Attention Networks (HAN) for document classification.

Meng et al. [10] proposed Weakly Supervised Text Classification (WSTC) model which achieves text classification, even when only limited seed information, such as several labeled seed documents, is given. According to the difference in encoder architectures, WSTC can be divided into WSTC-CNN and WSTC-HAN models. The WSTC model can be divided into seven steps:

1. **Extracting class related keywords.** We first extract k representative keywords from the labeled articles by using tf-idf [13] weighting, then we retrieve top- t nearest words in the word embedding space, and adopt these words as class-related words.
2. **Generating pseudo documents.** In order to pre-train the neural model, we first generate pseudo documents for each class, where words are sampled from the von Mises Fisher distribution on extracted class-related words. Each pseudo document is paired with a pseudo label to avoid overfitting problem.
3. **Neural Model.** WSTC utilizes two types of neural encoders: CNN [5] and HAN (Hierarchical Attention Neural Network) [15]. HAN is a variation of RNN which has a two-level encoder, where the first level is a word encoder and the second level is a sentence encoder.
4. **Neural model pre-training.** In order to overcome the overfit problem, this model creates pseudo labels for pseudo documents rather than using one-hot encoding. The neural model is pre-trained by minimizing the KL loss between output Y and pseudo labels L .
5. **Neural model self-training.** Unlabeled articles are used to self-train the Pre-WSTC model. First, the Pre-WSTC model is used to categorize the unlabeled articles, and

current output Y is obtained. Then pseudo labels are updated iteratively while minimizing the KL loss. When the number of changes in the category assignment is less than σ , this iterative process will stop.

3 Weakly Supervised Text Classification Neural Model with WS-Rank

3.1 WS-Rank

The WSTC-HAN model shows superior performance on benchmark corpora, which consists of news articles within 400 words, that is relatively shorter than practical informative documents such as Wikipedia articles and scientific papers. Neural encoders have difficulties in retaining memory on such long documents. So, we take the approach of summarizing long training and test articles into short versions. In this paper, we discuss three text extraction methods: leading k sentences, TextRank, and WS-Rank. WS-Rank is proposed in this paper, reflecting class-specific keywords.

Leading Sentences: We compress an article into short version, by selecting the first k sentences. But this method cannot capture important contents from the whole article.

TextRank: TextRank [9] is an unsupervised word/sentence scoring method.

WS-Rank: We extend TextRank to score keywords and sentences that are effective for training neural networks for classification. Also, since important and summarizing sentences are more likely to appear at the beginning or the end of the article, we incorporate sentence positions in scoring. Inspired by NE-Rank [2], we propose the WS-Rank algorithm as follows. Suppose we have m classes, and for each class i , we extract keyword set V_i by tf-idf:

$$V = \{V_1, V_2, V_i, \dots, V_m\} \quad (1)$$

WS-Rank extracts key sentences from one document as follows:

1. For each document d_k , split d_k into sentences. Let $[s_1, s_2, s_3, \dots, s_l]$ be the set of the sentences of all the documents.
2. We evaluate score St_j for each sentence s_j by TextRank.
3. For each sentence s_j in document d_k :
 - (1) For class i 's keyword set V_i , the score between sentence s_j and V_i is evaluated by Jaccard similarity:

$$Sk_{ij} = \text{Similarity}(V_i, s_j) = \frac{|\{w_k | w_k \in V_i \& w_k \in s_j\}|}{\log^{|V_i|} + \log^{|s_j|}} \quad (2)$$

- (2) We select the largest value as similarity weight $W(s_j)_{sim}$:

$$W(s_j)_{sim} = \max\{Sk_{1j}, Sk_{2j}, \dots, Sk_{mj}\} \quad (3)$$

The reason we select the largest value is to compare significance of each sentence by similarity to its most similar class.

- (3) The sentence position weight is evaluated by:

$$W(s_j)_{pos} = \left| pos_j - \frac{N}{2} \right| \times \left| \frac{len_j}{\max(len_m)} \right| \quad (4)$$

Here, pos_j is the position of s_j , N is the document length (the number of sentences), len_j is length (word count) of s_j , $\max(len_m)$ is the maximum length of the sentences in this document.

- (4) The final weight is defined as:

$$W(s_j)_{sim+pos} = W(s_j)_{sim} + W(s_j)_{pos} \quad (5)$$

- (5) Compute the WS-Rank score for each node s_j by:

$$\begin{aligned} Score(s_j) = & (1 - d) * W(s_j)_{sim+pos} \\ & + d * \sum_{v_j \in In(v_i)} \frac{W_{ij}}{\sum_{v_k \in out(v_j)} W_{jk}} Score(s_j) * W(s_j)_{sim+pos} \end{aligned} \quad (6)$$

4. Rank all sentences in one document by score $Score(s_j)$, and choose the top- k highly scored sentences as our short version of the document.

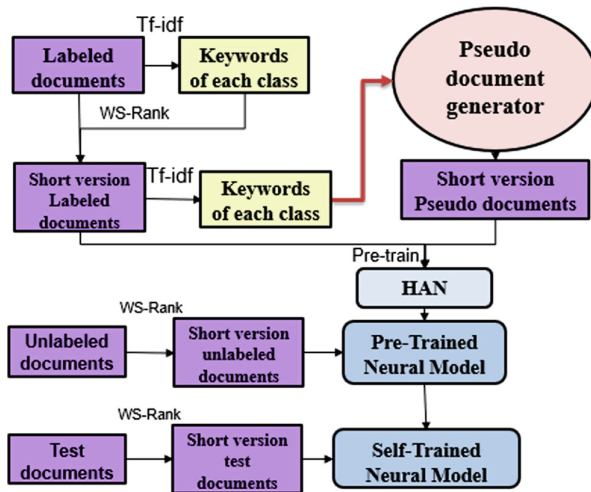


Fig. 1. Overview of WSTC with WS-Rank

3.2 WSTC-HAN with WS-Rank

In this section, we describe integration of WS-Rank into the WSTC model. The overview diagram is shown in Fig. 1. WSTC with WS-Rank can be divided into the following seven steps:

1. Apply `tf_idf` to extract keywords V_i for each class i from the labeled seed documents. V_i is supplied as additional information to WS-Rank.
2. Use full-length labeled seed documents to generate pseudo documents P , where the length of pseudo documents is set to k .
3. For each document d_i in the labeled seed documents and unlabeled documents, apply WS-Rank to obtain short version of these documents.
4. After Step 3 we obtain short version of labeled seed documents, denoted by D , and short version of unlabeled documents, denoted by U .
5. For each document t_i in the test dataset, we perform the same process in Step 3 to obtain short version T of the test documents.
6. Use P , D , and U to train the neural model.
7. Perform classification on T .

4 Experiments

4.1 Dataset Collection

We collect benchmark datasets from Wikipedia articles. One article named *Main topic classification* shows top-level categories of Wikipedia articles. From this level, we collect L1 Dataset, and from the sub- and sub-sub- categories, respectively, L2 and L3 Datasets are collected. Since WSTC assumes non-overlapping document classes, we omit articles that belong to multiple categories [1, 4]. The descriptions of our datasets are shown in Table 1. We expect that classification becomes harder as we descend the hierarchy down to L2 or L3, since categories at lower levels often contain overlapping topics and keyphrases.

We split each dataset into the training dataset and test dataset, according to the ratio of 8:2. We also extract $k = 10$ keywords for each class and regard them as class related keywords.

Table 1. Number of articles in each category

Dataset name	Category (No. of articles)	Total
L1 dataset	<i>Arts (843), Mathematics (2236), Law (1437), Language (1310), Sports (1427)</i>	7261
L2 dataset	<i>Architecture (2390), Communication design (888), Culinary Arts (268), Fashion design (1089), Performing arts (1202), Perfumery (540)</i>	6377
L3 dataset	<i>Advertising (1024), Document (1452), Graphic design (1258), Heraldry (766), Information visualization (201), Multimedia (642), Political communication (453)</i>	5823

4.2 Parameter Settings

The number of labeled articles for each class is 20. The background word distribution weight α is 0.2. The number of pseudo articles for each category β is 500. The self-training stop condition is $\sigma = 0.0001$. The dimension of word embeddings is 100. We choose $l = 15$ sentences as the size of shortened documents, since our preliminary experiments indicate that trained models show highest accuracies around $l = 15$, regardless of the summarization methods.

4.3 Experimental Results

Now we report our evaluation results. We use F1-Micro and F1-Macro scores to evaluate classification accuracy of each model.

We evaluate the effect of reducing input documents by WS-Rank for weakly-supervised text classification. Table 2 shows that when WS-Rank is used to extract top-15 key sentences, WSTC-HAN outperforms all the other models. Especially, when the dataset is L3, the F1-Macro score of WSTC-HAN with WS-Rank is 0.02 higher than WSTC-HAN with TextRank, and the F1-Micro score is 0.018 higher. If we only adopt the leading 15 sentences as the compressed version of one article, WSTC-HAN cannot catch important information written in the middle or end of articles. We can see that class-related information can help WS-Rank to extract key sentences which is useful for text classification. The F1 scores of L2 and L3 are lower than that of L1, which can be explained as the category level descends, the topics between each category become more similar each other, making classification harder.

Table 2. Results of different models for weakly-supervised Wikipedia article categorization.

Model	L1 dataset		L2 dataset		L3 dataset	
	F1-Macro	F1-Micro	F1-Macro	F1-Micro	F1-Macro	F1-Micro
WSTC-CNN	0.8523	0.85283	0.74929	0.74234	0.67047	0.6704
WSTC-HAN	0.81675	0.82828	0.7207	0.71799	0.62344	0.62198
WSTC-HAN (leading 15 sentences)	0.88582	0.90276	0.79397	0.80911	0.67194	0.6704
WSTC-HAN (TR@15s)	0.894	0.90966	0.80686	0.81697	0.68221	0.68675
WSTC-HAN (WSR_Sim@15s)	0.8962	0.9089	0.80867	0.82269	0.72087	0.73404
WSTC-HAN (WSR_Pos@15s)	0.8978	0.911	0.82658	0.83739	0.71333	0.7213
WSTC-HAN (WS-Rank)	0.8989	0.91403	0.83458	0.84039	0.72868	0.74128

In Table 2, TR@15 means TextRank is used to extract top-15 sentences as the short version of a long document. WSR_Sim@15s means that WS-Rank is used to extract top-15 sentences as the short version of long documents and the weight is similarity. WSR_Pos@15s means that WS-Rank is used to extract top-15 sentences as the short version of long documents, and the weight is position score. WSR_Pos@15s means

that WS-Rank is used to extract top-15 sentences as the short version of long documents, and the weight is position score plus similarity score.

5 Conclusion

In this paper, we discussed weakly-supervised document classification, particularly on corpora of Wikipedia articles that tend to become longer and topics are overlapping each other. We proposed WS-Rank to reduce long training and test articles, while giving emphasis on class-related keywords and sentence positions. Through experimental evaluations we find that our model which combines WS-Rank with WSTC-HAN is outperforming all other compared models. For future work, we will extend our research to hierarchical text classification.

References

1. Aouicha, M.B., Ali, M., Taieb, H., Ezzeddine, M.: Derivation of ‘is a’ taxonomy from Wikipedia Category Graph. *Eng. Appl. Artif. Intell.* **50**, 265–286 (2016)
2. Abdelghani, B., Al-Dhelaan, M.: Ne-rank: a novel graph-based keyphrase extraction in twitter. In: 2012 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology, vol. 1. IEEE (2012)
3. Ferreira, R., Freitas, F., Cabral, L.S., Lins, R.D.: A four dimension graph model for automatic text summarization. IEEE (2013)
4. Kittur, A.: What’s in Wikipedia? Mapping topics and conflict using socially annotated category structure, pp. 1509–1512. ACM (2009)
5. Kim, Y.: Convolutional neural network for sentence classification. In: EMNLP, pp. 1746–1751 (2014)
6. Lu, J., Shang, J., Cheng, C., Ren, X., Han, J.: Mining quality phrases from massive text corpora. In: SIGMOD (2015)
7. Liu, P., Qiu, X., Huang, X.: Recurrent neural network for text classification with multi-task learning. In: IJCAI 2016 (2016)
8. Lai, S., Xu, L., Liu, K., Zhao, J.: Recurrent convolutional neural networks for text classification. In: AAAI, vol. 333 (2015)
9. Mihalcea, R., Tarau, P.: TextRank: bridging order into texts. In: EMNLP, pp. 404–411 (2004)
10. Meng, Y., Shen, J., Zhang, C., Han, J.: Weakly-supervised neural text classification. In: CIKM (2018)
11. Parveen, D., Ramsel, H., Strube, M.: Topical coherence for graph-based extractive summarization. In: EMNLP, pp. 1949–1954 (2015)
12. Pengfei, L., Qiu, X., Huang, X.: Recurrent neural network for text classification with multi-task learning. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence. AAAI Press (2016)
13. Ramos, J.: Using TF-IDF to determine word relevance in document queries. In: ICML (2003)
14. Song, Y., Roth, D.: On dataless hierarchical text classification. In: AAAI (2014)
15. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A.J., Hovy, E.H.: Hierarchical attention networks for document classification. In: HLT-NAACL (2016)
16. Zhang, X., Zhao, J., LeCun, Y.: Character-level convolutional networks for text classification. In: NIPS (2015)



Supervised Classifiers to Identify Hate Speech on English and Spanish Tweets

Sattam Almatarneh^{1,2(✉)}, Pablo Gamallo¹, Francisco J. Ribadas Pena²,
and Alexey Alexeev³

¹ Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS),
Universidad de Santiago de Compostela, Santiago, Spain

{sattam.almatarneh,pablo.gamallo}@usc.es

² Computer Science Department, University of Vigo Escola Superior de Enxeñaría
Informática, Campus As Lagoas, 32004 Ourense, Spain

³ ITMO University, Saint-Petersburg, Russia

Abstract. Consistently with social and political concern about hatred and harassment through social media, in recent years, automatic hate-speech detection and offensive behavior in social media are gaining a lot of attention. In this paper, we examine the performance of several supervised classifiers in the process of identifying hate speech on Twitter. More precisely, we do an empirical study that analyzes the influence of two types of linguistic features (n-grams, word embeddings) when they are used to feed different supervised machine learning classifiers: Support Vector Machine (SVM), Gaussian Naive Bayes (GNB), Complement Naive Bayes (CNB), Decision Tree (DT), Nearest Neighbors (KN), Random Forest (RF) and Neural Network (NN). The experiments we have carried out show that CNB, SVM, and RF are better than the rest classifiers in English and Spanish languages by taking into account all features.

Keywords: Hate speech · Sentiment analysis · Linguistic features · Classification · Supervised machine learning

1 Introduction

Hate speech is defined as the language that attacks or diminishes, that incites violence or hate against groups, based on specific characteristics such as physical appearance, religion, descent, national or ethnic origin, sexual orientation, gender identity or other, and it can occur with different linguistic styles, even in subtle forms or when humor is used [10]. Hate speech identification is the sub-field of natural language processing that studies the automatic inference of offensive language and hate speech from textual data. The motivation behind studying this sub-field is to possibly limit the hate speech on user-generated content, particularly, on social media. One popular social media platform for researchers to study is Twitter, a social network website where people “tweet” short posts.

In machine learning, there are two main methods, unsupervised and supervised learning. Unsupervised learning approaches do not depend on the domain and topic of training data. Supervised learning approaches use labeled training documents based on automatic text classification. A labeled training set with pre-defined categories is required. A classification model is built to predict the document class based on pre-defined categories. The success of supervised learning mainly depends on the choice and extraction of the proper set of features used to identify the target object: hate speech. Even though there are still few approaches to hate speech identification, there are many types of classifiers for sentiment classification, which is the most similar task to hate speech detection.

The main objective of this article is to examine the effectiveness and limitations of supervised classifiers to identify Hate Speech detection in Twitter focused on two specific targets, women and immigrants in two languages: English and Spanish. The main contribution of this paper is to report on a broad set of experiments aimed at evaluating the effectiveness of the most influence linguistic features in a supervised classification task. Besides, we compare some supervised classifiers, namely Support Vector Machine (SVM), Gaussian Naive Bayes (GNB), Complement Naive Bayes (CNB), Decision Tree (DT), Nearest Neighbors (KN), Random Forest (RF) and Neural Network (NN) for our binary classification task.

The rest of the paper is organized as follows. In the following Sect. 2, we introduce some related work. Then, Sect. 3 describes the method. Experiments are introduced in Sect. 4, where we also describe the evaluation and discuss the results. We draw the conclusions and future work in Sect. 5.

2 Related Work

Early studies on hate speech detection focus mainly on lexicon-based approaches [12, 14]. As well, some researchers deal with the problem by employing feature (e.g., N-gram, TF-IDF) based supervised learning approach using SVM and Naive-Bayes classifier [11, 22].

Bag of words is often reported to be highly predictive and most evident information to employ, unigrams and larger n-grams are included in the feature sets by a majority of text classification task studies such as [5, 7]. In many works, n-gram features are combined with a large selection of other features. For example, in their recent work, [2, 18] report that while token and character n-gram features are the most predictive single features in their experiments, combining them with all additional features further improves performance. [1] compared different supervised machine learning classifiers: Naive Bayes (NB), Decision Tree (DT), and Support Vector Machine (SVM). The experiments show that SVM clearly outperforms NB and DT on the task of Search for Very Negative Opinions.

Table 1 lists the main components of some published studies: Reference, Year, Features, and Techniques Utilized.

Table 1. The main components of some published studies of supervised learning for Hate speech detection.

Reference	Year	Features	Techniques utilized
[13]	2004	BOW, N-grams, POS	SVM
[14]	2013	N-grams	NB
[6]	2014	BOW, dictionary, typed dependencies	SVM, NB, DT
[5]	2015	N-gram, typed dependencies	RF, DT, SVM, Bayesian logistic regression ensemble
[21]	2016	Dectionary	SVM
[2]	2019	TF-IDF, N-grams, Word embedding, Lexicon	SVM

3 The Method

To compare different classification algorithms, we build the corresponding classifiers by making use of the same training data in a supervised strategy. The characteristics of tweets are encoded as features in vector representation. These vectors and the corresponding labels feed the classifiers.

3.1 Features

Linguistic features are the most important and influential factor in increasing the efficiency of classifiers for any task of text mining. In this study, we included a number of linguistic features for the task of determining hate speech in tweets. The main linguistic features we will use and analyze are the following: N-grams and word embeddings features.

N-Grams Features: We deal with n-grams based on the occurrence of unigrams and bigrams of words in the document. Unigrams (1g) and bigrams (2g) are valuable to detect specific domain-dependent (opinionated) expressions. We assign a weight to all terms by using two representations: Term Frequency-Inverse Document Frequency (TF-IDF) and CountVectorizer.

CountVectorizer transforms the document into token count matrix. First, it tokenizes the document and according to a number of occurrences of each token, a sparse matrix is created. In order to create the matrix, all stopwords are removed from the document collection. Then, the vocabulary is cleaned up by eliminating those terms appearing in less than 4 documents to eliminate those terms that are infrequent.

Doc2Vec: To represent the tweets as dense matrices or embeddings, we make use of the *Doc2Vec* algorithm described in [15]. This neural-based model is efficient when you have to account for high-dimensional and sparse data [9, 15]. Doc2vec learns corpus features using an unsupervised strategy and provides a fixed-length feature vector as output. The output is then fed into a machine learning classifier. We used a freely available implementation of the Doc2Vec algorithm included in *gensim*,¹ which is a free Python library. The implementation of the Doc2Vec algorithm requires the number of features to be returned (length of the vector). Thus, we performed a grid search over the fixed vector length 100 [8, 16, 17].

4 Experiments

4.1 Dataset

The multilingual detection of hate speech (HatEval) task 5 at SemEval-2019 [3] provides a benchmark dataset. The proposed task focuses on two specific different targets, including immigrants and women in a multilingual perspective, for Spanish and English. The data for the task consists of 9000 tweets in English for training, 1000 for developing and 2805 for the test. For Spanish, 4469 tweets for training, 500 for developing, and 415 for the test. The data are structured in 5 columns: ID, Text, Hate Speech (HS), Target Range (TR) and Aggressiveness (AG). In our study, we consider only the first two columns to identify if the tweet is classified as hate speech or not.

4.2 Training and Test

Since we are dealing with a text classification problem, any existing supervised learning methods can be applied. We decided to utilize *scikit*², which is an open source machine learning library for Python programming language [19]. We chose SVM, GNB, CNB, DT, KN, RF, and NN as our classifiers for all experiments. Hence, in this study, we will compare, summarize and discuss the behavior of these learning models with the linguistic features introduced above. In order to provide a comprehensive comparison between classifiers, we adopted the default values for all classifiers on all experiments.

1. Support Vector Machines (SVM): SVMs are supervised learning methods used for classification and regression, working effectively in high dimensional spaces. SVM classifiers show excellent performance on the text classification task. In our experiments, we chose LinearSVC from the scikit-learn library.
2. Naive Bayes (NB): NB methods are a set of supervised learning algorithms based on applying Bayes theorem with the Naive assumption of conditional independence between every pair of features given the value of the class variable. We used two algorithms from the scikit-learn library:

¹ <https://radimrehurek.com/gensim/>.

² <http://scikit-learn.org/stable/>.

- Gaussian Naive Bayes classifier. The likelihood of the features is assumed to be Gaussian:

$$P(x_i | y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right) \quad (1)$$

The parameters σ_y and μ_y are estimated using maximum likelihood.

- The Complement Naive Bayes classifier. As described in [20]. The complement naive Bayes (CNB) algorithm is an adaptation of the standard multinomial naive Bayes (MNB) algorithm which is convenient for imbalanced data sets. More precisely, to compute the model’s weights, CNB utilizes statistics from the complement of each class. The parameter estimates for CNB is better than those for MNB. Also, CNB outperforms MNB on text classification tasks. The procedure for calculating the weights is by the following equation :

$$\begin{aligned} \hat{\theta}_{ci} &= \frac{\alpha_i + \sum_{j:y_j \neq c} d_{ij}}{\alpha + \sum_{j:y_j \neq c} \sum_k d_{kj}} \\ w_{ci} &= \log \hat{\theta}_{ci} \\ w_{ci} &= \frac{w_{ci}}{\sum_j |w_{cj}|} \end{aligned} \quad (2)$$

where the summations are over all documents j which are not in class c ; d_{ij} is either the count or tf-idf value of term i in document j ; α_i is a smoothing hyperparameter as that found in MNB; and $\alpha = \sum_i \alpha_i$.

3. Decision Trees (DT): DTs are a non-parametric supervised learning method for regression and classification where predicts the value of a target variable by learning simple decision rules inferred from the data features.
4. Nearest Neighbors (KN): The precept behind nearest neighbor methods is to find a predefined number of training samples closest in the distance to the new point and predict the label from these. The number of samples can be a user-defined constant (k-nearest neighbor learning) or vary based on the local density of points (radius-based neighbor learning).
5. Random Forest(RF): RF classifier is an ensemble algorithm where each tree in the ensemble is built from a sample drawn with replacement (i.e., a bootstrap sample) from the training set. We used RandomForestClassifier from the scikit-learn library, which combines classifiers by averaging their probabilistic prediction, instead of letting each classifier vote for a single class, in contrast to the original publication [4].
6. Neural Network (NN): We only examined one class of neural network algorithms from the scikit-learn library which is Multilayer perceptron (MLP). It is one of the more basic neural network methods.

4.3 Discussion

Table 2 shows that CNB, SVM, and RF are by far the best classifiers for identifying the hate speech in both languages English and Spanish. N-grams features

Table 2. Hate speech classification results of all classifiers, in terms of F1 scores for English and Spanish languages with linguistic features. The best F1 are highlighted (in bold).

	SVM		GNB		CNB		DT		KN		RF		NN	
	Eng.	Span.	Eng.	Span.	Eng.	Span.	Eng.	Span.	Eng.	Span.	Eng.	Span.	Eng.	Span.
TFIDF 1g	0.76	0.77	0.56	0.58	0.75	0.77	0.73	0.71	0.63	0.73	.75	0.75	0.71	0.72
TFIDF 2g	0.70	0.72	0.60	0.60	0.73	0.74	0.64	0.68	0.56	0.60	0.70	0.71	0.68	0.69
Countvect 1g	0.73	0.75	0.53	0.56	0.76	0.76	0.76	0.74	0.66	0.70	0.75	0.77	0.74	0.75
Countvect 2g	0.68	0.71	0.58	0.59	0.74	0.73	0.67	0.71	0.63	0.66	0.69	0.72	0.68	0.70
Doc2Vec	0.70	0.58	0.65	0.51	–	–	0.61	0.55	0.65	0.59	0.65	0.59	0.71	0.65

achieve the highest F1 scores in almost all tests regardless of whether the representations are CountVectorizer or TF-IDF. The performance of Gaussian Naive Bayes differs greatly depending on the number of features used in classification (see Table 2). GNB works better with a small number of features, more precisely the best scores are achieved when it only uses Doc2Vec. It is worth noting that the combination of heterogeneous features hurts the performance of this type of classifier. By contrast, the Complement Naive Bayes showed a good performance and consistent with the features of the bag of words, whether they are represented by TF-IDF or Countvectorizer. However, in MNB and CNB classifiers, the input value of features must be non-negative. Therefore, they cannot deal with Doc2Vec features, as they contains negative value which is useless if we normalize the value to (0,1) as shown in Table 2.

Concerning the linguistic features, the best performance of most classifiers is reached with TF-IDF and Countvectorizer, for both 1g or 2g.

The DT classifier has similar behavior to SVM in terms of stability, but its performance tends to be much lower than that of SVM on both languages.

5 Conclusions

In this article, we have compared different supervised classifiers for a particular task. More precisely, we examined the performance of the most influence features within supervised learning methods (using SVM, GNB, CNB, DT, KN, RF, and NN), to identify the hate speech on English and Spanish tweets.

Concerning the comparison between machine learning strategies in this particular task, Support Vector Machine, Complement Naive Bayes, and Random Forest clearly outperforms all the rest classifiers and show stable performance with all features.

In future work, we will compare other types of classifiers with more complex linguistic features, by taking into account the new deep learning approaches based on neural networks.

Acknowledgments. Research partially funded by the Spanish Ministry of Economy and Competitiveness through projects *TIN2017 – 85160 – C2 – 2 – R*, and by the Galician Regional Government under projects *ED431C 2018/50*.

References



1. Almatarneh, S., Gamallo, P.: Comparing supervised machine learning strategies and linguistic features to search for very negative opinions. *Information* **10**(1), 16 (2019). <http://www.mdpi.com/2078-2489/10/1/16>
2. Almatarneh, S., Gamallo, P., Pena, F.J.R.: CiTIUS-COLE at semeval - 2019 task 5: combining linguistic features to identify hate speech against immigrants and women on multilingual tweets. In: *The 13th International Workshop on Semantic Evaluation* (2019)
3. Basile, V., et al.: Semeval-2019 task 5: Multilingual detection of hate speech against immigrants and women in twitter. In: *Proceedings of the 13th International Workshop on Semantic Evaluation*, pp. 54–63 (2019)
4. Breiman, L.: Random forests. *Mach. Learn.* **45**(1), 5–32 (2001)
5. Burnap, P., Williams, M.L.: Cyber hate speech on twitter: an application of machine classification and statistical modeling for policy and decision making. *Policy Internet* **7**(2), 223–242 (2015)
6. Burnap, P., Williams, M.L.: Hate speech, machine classification and statistical modelling of information flows on twitter: interpretation and communication for policy decision making (2014)
7. Chen, Y., Zhou, Y., Zhu, S., Xu, H.: Detecting offensive language in social media to protect adolescent online safety. In: *2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Conference on Social Computing*, pp. 71–80. IEEE (2012)
8. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. *J. Mach. Learn. Res.* **12**(Aug), 2493–2537 (2011)
9. Dai, A.M., Olah, C., Le, Q.V.: Document embedding with paragraph vectors. arXiv preprint [arXiv:1507.07998](https://arxiv.org/abs/1507.07998) (2015)
10. Fortuna, P., Nunes, S.: A survey on automatic detection of hate speech in text. *ACM Comput. Surv. (CSUR)* **51**(4), 85 (2018)
11. Gaydhani, A., Doma, V., Kendre, S., Bhagwat, L.: Detecting hate speech and offensive language on twitter using machine learning: An n-gram and tfidf based approach. arXiv preprint [arXiv:1809.08651](https://arxiv.org/abs/1809.08651) (2018)
12. Gitari, N.D., Zuping, Z., Damien, H., Long, J.: A lexicon-based approach for hate speech detection. *Int. J. Multimed. Ubiquit. Eng.* **10**(4), 215–230 (2015)
13. Greevy, E., Smeaton, A.F.: Classifying racist texts using a support vector machine. In: *Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 468–469. ACM (2004)
14. Kwok, I., Wang, Y.: Locate the hate: detecting tweets against blacks. In: *Twenty-seventh AAAI Conference on Artificial Intelligence* (2013)
15. Le, Q., Mikolov, T.: Distributed representations of sentences and documents. In: *International Conference on Machine Learning*, pp. 1188–1196 (2014)
16. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint [arXiv:1301.3781](https://arxiv.org/abs/1301.3781) (2013)
17. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: *Advances in Neural Information Processing Systems*, pp. 3111–3119 (2013)
18. Nobata, C., Tetreault, J., Thomas, A., Mehdad, Y., Chang, Y.: Abusive language detection in online user content. In: *Proceedings of the 25th International Conference on World Wide Web, International World Wide Web Conferences Steering Committee*, pp. 145–153 (2016)

19. Pedregosa, F., et al.: Scikit-learn: machine learning in python. *J. Mac. Learn. Res.* **12**(Oct), 2825–2830 (2011)
20. Rennie, J.D., Shih, L., Teevan, J., Karger, D.R.: Tackling the poor assumptions of naive Bayes text classifiers. In: *Proceedings of the 20th International Conference on Machine Learning (ICML-2003)*, pp. 616–623 (2003)
21. Tulkens, S., Hilte, L., Lodewyckx, E., Verhoeven, B., Daelemans, W.: A dictionary-based approach to racism detection in dutch social media. *arXiv preprint [arXiv:1608.08738](https://arxiv.org/abs/1608.08738)* (2016)
22. Unsvåg, E.F., Gambäck, B.: The effects of user features on twitter hate speech detection. In: *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pp. 75–85 (2018)

Altmetrics



An Overview of Altmetrics Research: A Typology Approach

Han Zheng^(✉) , Xiaoyu Chen , and Xu Duan

Wee Kim Wee School of Communication and Information,
Nanyang Technological University, Singapore, Singapore
{han019, xiaoyu001, xu007}@e.ntu.edu.sg

Abstract. Altmetrics, novel metrics based on social media, have received much attention from scholars in recent years. As an emerging research area in information science, it is essential to understand the overview of altmetrics research. We extracted 731 altmetrics-related articles from the Scopus databases and adopted a text mining method (i.e., topic modeling) to develop a typology of altmetrics research. Six major themes were identified in our analysis, including altmetrics research in general, bibliometric and altmetrics, measuring research impact, metrics application, social media use, and performance evaluation. We interpreted the meaning of the six themes and their associations with altmetrics research. This paper is a first step in mapping the landscape of altmetrics research through uncovering the core topics discussed by scholars. Limitations and future work are also discussed.

Keywords: Altmetrics research · Typology · Topic modeling · Text mining

1 Introduction

Altmetrics refers to a series of novel metrics based on social media that can potentially evaluate research impact [1]. The research on altmetrics has received much attention from scholars due to its mixed effect on scholarly communication [2]. On the one hand, altmetrics could complement traditional bibliometric indicators (e.g., citation count) in terms of timely measuring research impact [3]. On the other hand, the validity and reliability of altmetrics are still debatable [4, 5]. As an emerging area, despite altmetrics research increasingly prevalent in the literature, the overview of altmetrics research is far from clear. We thus seek to fill this gap by developing a typology of altmetrics research based on the existing literature.

Abstracts of related literature can be used as a corpus for summarizing mainstream directions of a research area. This is because an abstract contains the most critical content and research focus of an academic article, which tend to be written by the authors with many efforts. Several works on bibliometric also acknowledged the role of abstracts in understanding the overview of a certain research area (e.g., [6, 7]). Through analyzing the abstracts of literature, we can identify core topics discussed most among researchers [8]. Additionally, the associations among the topics can help us to trace the current research directions [9].

Therefore, in response to developing a typology of altmetrics research, this study plans to do it as follows. First, we employ a topic modeling technique based on Latent Dirichlet Allocation (LDA) [10], enabling us to identify core topics from the abstracts of existing literature. Second, we interpret the meaning of identified core topics and their associations with altmetrics research, thereby forming a typology of altmetrics research. The preliminary results of this study will lay a foundation of picturing a landscape of altmetrics research.

2 Method

We used the Scopus database (<https://www.scopus.com/>) to retrieve the data for the study as it covers a sizable amount of high-quality research articles across multiple disciplines. The data were collected on 20 February 2019. Following prior work (e.g., [11]), three query words (i.e., altmetrics, alternative metrics, and social media metrics) were used in the search process. We limited our search in titles/abstracts/keywords of altmetrics-related articles that were published until February 2019. We limited the article language in English while we did not limit the document types in order to ensure the comprehensiveness of our collected data. The search yielded a total of 1,132 articles, and document-level information including authors, year, article title, journal title, abstract, and author keywords was retrieved. Although we used the three query words to extract altmetrics-related articles, we indeed found that some of the articles did not belong to this research area since the terms “alternative metrics” and “social media metrics” are also widely used in other disciplines (e.g., social media metrics for hoteliers). Thus, we checked our data manually to exclude irrelevant articles and included 731 articles in the final sample.

We conducted the data analysis using R statistical software version 3.5.1. We first created a corpus of all abstracts and standardized the terms by removing numbers, punctuations, and stopwords, and transforming the upper case to lower case. Tokenization was also performed to reduce the inflected words to their root form. Next, word cloud analysis was used to visually present the highly frequent-used words in altmetrics studies. Word cloud is widely employed in academia to identify the focus of a research area [12]. The more frequent a term is used in the documents, the larger the word appears in the image [13]. Before the generation of topic models, we assessed the perplexity score, an index of how well a probability distribution predicts the sample, to define the number of topics that describe the corpus. The lower perplexity score indicates less prediction error and the probability model fits the data well [8]. To identify the latent core topics in altmetrics research, we employed Latent Dirichlet Allocation (LDA) topic modeling, which views words as a bag of words and computes the probability that a word belongs in a topic, given that the probability of a topic belongs in an abstract [9]. Lastly, we discussed and interpreted the meaning of the topics and their relevance to altmetrics research.

3 Results

Figure 1 shows the distribution of altmetrics publications by year. As can be noted, there was a significant rise in the number of publications during the past decade, which demonstrates an increasing interest in this research area over the years. The word cloud analysis of abstracts is shown in Fig. 2. The maximum word limit of 100 was set. It can be seen that the most prominent words are ‘research,’ ‘altmetr,’ ‘metric,’ ‘impact,’ ‘social,’ and ‘use.’ This finding is consistent with the definition of altmetrics, which refers to metrics on social media that are used to assess research impact. Also, words such as ‘citat,’ ‘articl,’ and ‘bibliometr’ can be found in the image, indicating that altmetrics research focuses on traditional metrics as well, especially examining the relationship between altmetrics and bibliometric indicators (e.g., citation count). In conclusion, the word cloud of abstracts has shown the focus of altmetrics research from a general perspective.

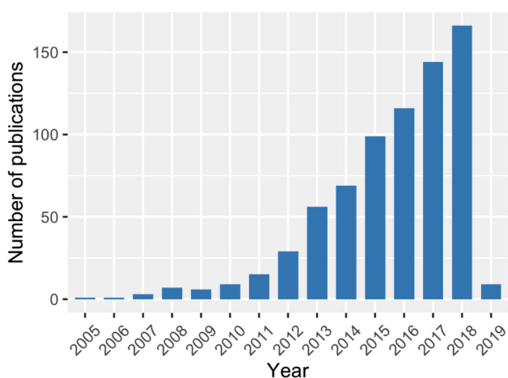


Fig. 1. The number of publications by year.



Fig. 2. Word cloud of abstracts.

Next, we used a topic modeling approach to investigate the latent core topics in altmetrics research. First, the distribution of perplexity score is shown in Fig. 3. It can be seen that the perplexity score decreases with the increase in the number of topics. When the number of topics equals to 10, it has the lowest perplexity score, indicating a good fit of the model to the data. Nevertheless, the decreasing trend of perplexity score starts to slow down when the number of topics is more than 4. To achieve a balance between human capability of interpreting topic meanings and sufficiency of words in a topic [14], three aspects were considered including the width and content of the corpus, the words included in each topic, and the perplexity score for a given number of topics. Specifically, after attempting different numbers of topics (i.e., from 2 topics to 10 topics) and assessing the perplexity score as well as the words in each topic, we determined that the optimal number of topics for interpretation and model fit is 6.

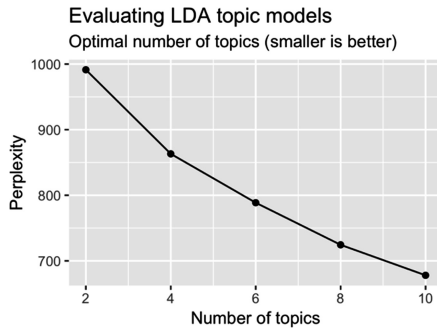


Fig. 3. Perplexity score for the given number of topics.

After running the LDA model, the 6 topics and the words included in each topic are shown in Fig. 4.

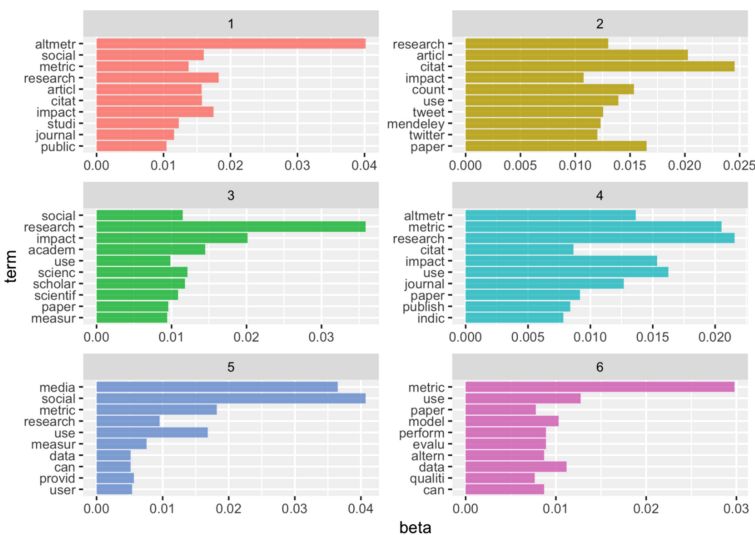


Fig. 4. Topics and associated words

The beta value refers to the probability of a word that is generated from a given topic. For example, the word ‘research’ has a 1.82% probability of being generated from topic 1 whereas it has a 3.59% probability of being generated from topic 3. To interpret the meaning of the topics, we assessed the common words in each topic. For instance, the most common words in topic 1 are ‘altmetr’ and ‘research,’ suggesting the topic is related to altmetrics research in general. In this way, we labeled the six topics as follows: altmetrics research in general, bibliometrics and altmetrics, measuring research impact, metrics application, social media use, and performance evaluation. The next section discusses how the six topics are relevant to altmetrics research.

4 Discussion and Conclusion

This study has found popular words and uncovered the latent core themes in altmetrics research. As an emerging research area in information science, the decomposition of the altmetrics literature helps to advance our understanding of the research foci comprehensively. Specifically, the six core themes generated in this study may provide valuable guidance for future research as follows.

First, topic 1 “altmetrics research in general” describes the investigation of altmetrics from a general perspective. This type of studies focuses on introducing and defining altmetrics as a novel tool in academia (e.g., [15]). Second, topic 2 “bibliometrics and altmetrics” includes core words such as ‘tweet,’ ‘mendeley,’ and ‘citat,’ which may depict the relationship between traditional metrics and altmetrics. Several studies have demonstrated that altmetrics can be considered as a complement of traditional metrics [16]. Specifically, [4] showed that the number of tweets is an indicator of highly cited articles within the first three days of article publication. Third, topic 3 “measuring research impact” indicates that scholars measure research impact of scientific papers because articles need research evaluation [17], and those with high citation count are considered to be influential, making a significant contribution to research area [15].

Fourth, topic 4 “metric application” contains core words such as ‘metric,’ ‘journal,’ and ‘paper,’ which means both traditional metrics and altmetrics are used at different document levels. For example, citation count is an article-level metric to assess the quality of the paper whereas journal impact factor (JIF) is a journal-level proxy for the quality of the papers published in a given journal [18]. Fifth, “social media use” refers to the adoption of social media tools among scholars for promoting research work. An example of this topic can be a survey study conducted by [19] to investigate the use of social media in the research workflow, and they concluded that social media has become increasingly important for promoting and discovering research. Similarly, academic journals also embark on using social media platforms to promote research outputs [20]. Lastly, “performance evaluation” indicates the use of metrics to evaluate performance. Scholars and research institutions currently rely much on traditional metrics to assess research performance while few have adopted altmetrics due to concern about quality [3]. In summary, this study develops a typology of altmetrics research through a pragmatic and data-driven classification approach, which can serve

as a complement of existing classifications and establish a solid foundation for future altmetrics research.

Our study is not without limitations. First, our study is limited by the number of abstracts used to build topic modeling. We only included several hundred abstracts in altmetrics research, and the results may not convey the foci of this research area perfectly. However, the articles indexed in the Scopus database represent altmetrics studies of high quality, which helps us understand the overall landscape of altmetrics research. Second, the six topics generated in the study are not mutually exclusive. Future research can look into how these topics are correlated and co-evolve with each other. Moreover, it would also be interesting to investigate the impact of these altmetrics studies (e.g., based on bibliometric and altmetrics analysis) to identify prominent researchers and active institutions in this area. Nevertheless, to our best knowledge, our paper is one of the first studies to adopt a text-mining approach to investigate the core topics in altmetrics research, paving the way for future research and development in this area.

References

1. Thelwall, M., Haustein, S., Larivière, V., Sugimoto, C.R.: Do altmetrics work? Twitter and ten other social web services. *PLoS ONE* **8**(5), e64841 (2013)
2. Harley, D.: Scholarly communication: cultural contexts, evolving models. *Science* **342** (6154), 80–82 (2013). <https://doi.org/10.1126/science.1243622>
3. Aung, H.H., Zheng, H., Erdt, M., Aw, A.S., Sin, S.J., Theng, Y.L.: Investigating familiarity and usage of traditional metrics and altmetrics. *J. Assoc. Inf. Sci. Technol.* (2019). <https://doi.org/10.1002/asi.24162>
4. Eysenbach, G.: Can tweets predict citations? Metrics of social impact based on Twitter and correlation with traditional metrics of scientific impact. *J. Med. Internet Res.* **13**(4), e123 (2011). <https://doi.org/10.2196/jmir.2012>
5. Yu, H., Xu, S., Xiao, T.: Is there Lingua Franca in informal scientific communication? Evidence from language distribution of scientific tweets. *J. Inf.* **12**(3), 605–617 (2018). <https://doi.org/10.1016/j.joi.2018.06.003>
6. Chua, A.Y., Yang, C.C.: The shift towards multi-disciplinarily in information science. *J. Am. Soc. Inf. Sci. Technol.* **59**(13), 2156–2170 (2008). <https://doi.org/10.1002/asi.20929>
7. Pal, A., Chua, A.Y.: Reviewing the landscape of research on the threats to the quality of user-generated content. *Proc. Assoc. Inf. Sci. Technol.* **53**(1), 1–9 (2016). <https://doi.org/10.1002/pra2.2016.14505301077>
8. Yoshida, M., Nakagawa, H.: Automatic term extraction based on perplexity of compound words. In: Dale, R., Wong, K.-F., Su, J., Kwong, O.Y. (eds.) *IJCNLP 2005*. LNCS (LNAI), vol. 3651, pp. 269–279. Springer, Heidelberg (2005). https://doi.org/10.1007/11562214_24
9. Papachristopoulos, L., et al.: Discovering the structure and impact of the digital library evaluation domain. *Int. J. Digit. Libr.* **20**, 125–141 (2017). <https://doi.org/10.1007/s00799-017-0222-x>
10. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. *J. Mach. Learn. Res.* **3**, 993–1022 (2003)
11. Erdt, M., Nagarajan, A., Sin, S.C.J., Theng, Y.L.: Altmetrics: an analysis of the state-of-the-art in measuring research impact on social media. *Scientometrics* **109**(2), 1117–1166 (2016)

12. Bungo, J.: Embedded systems programming in the cloud: a novel approach for academia. *IEEE Potentials* **30**(1), 17–23 (2011). <https://doi.org/10.1109/mpot.2010.938614>
13. Atenstaedt, R.: Word cloud analysis of the BJGP. *Br. J. Gen. Pract.* **62**(596), 148 (2012). <https://doi.org/10.3399/bjgp12X630142>
14. Noh, Y., Hagedorn, K., Newman, D.: Are learned topics more useful than subject headings. In: Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries, pp. 411–412. ACM (2011). <https://doi.org/10.1145/1998076.1998160>
15. Weller, K.: Social media and altmetrics: an overview of current alternative approaches to measuring scholarly impact. In: Welpel, I.M., Wollersheim, J., Ringelhan, S., Osterloh, M. (eds.) *Incentives and Performance*, pp. 261–276. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-09785-5_16
16. Zahedi, Z., Costas, R., Wouters, P.: How well developed are altmetrics? A cross-disciplinary analysis of the presence of ‘alternative metrics’ in scientific publications. *Scientometrics* **101** (2), 1491–1513 (2014). <https://doi.org/10.1007/s11192-014-1264-0>
17. Zheng, H., Erdt, M., Theng, Y.-L.: How do scholars evaluate and promote research outputs? An NTU case study. In: Erdt, M., Sesagiri Raamkumar, A., Rasmussen, E., Theng, Y.-L. (eds.) *AROSIM 2018*. CCIS, vol. 856, pp. 72–80. Springer, Singapore (2018). https://doi.org/10.1007/978-981-13-1053-9_6
18. Borgman, C.L., Furner, J.: Scholarly communication and bibliometrics. *Annu. Rev. Inf. Sci. Technol.* **36**(1), 2–72 (2002). <https://doi.org/10.1002/aris.1440360102>
19. Nicholas, D., Rowlands, I.: Social media use in the research workflow. *Inf. Serv. Use* **31**(1–2), 61–83 (2011). <https://doi.org/10.3233/isu-2011-0623>
20. Zheng, H., Aung, H.H., Erdt, M., Peng, T.Q., Sesagiri Raamkumar, A., Theng, Y.L.: Social media presence of scholarly journals. *J. Assoc. Inf. Sci. Technol.* **70**(3), 256–270 (2019)



User Motivation Classification and Comparison of Tweets Mentioning Research Articles in the Fields of Medicine, Chemistry and Environmental Science

Mahalakshmi Suresh Kumar, Shreya Gupta, Subashini Baskaran^(✉),
and Jin-Cheon Na

Wee Kim Wee School of Communication and Information,
Nanyang Technological University, 31 Nanyang Link,
Singapore 637718, Singapore
{mahalaks002, shreya015, subashin001}@e.ntu.edu.sg,
tjcna@ntu.edu.sg

Abstract. Modern metrics like Altmetrics help researchers and scientists to gauge the impact of their research findings through social media discussions. Twitter holds more scholarly and scientific discussions than other social media platforms and is extensively used to discuss and share research articles by domain experts as well as by the general public. In this study, we have analyzed the motivations of people using Twitter as a medium to propagate the research works. Tweets and the publication details from the field of medicine are collected from altmetric.com for journals with high impact factors and a Support Vector Machine classifier is developed with 85.2% accuracy to categorize the tweets into one of the six motivation classes. The model is then extended to observe the pattern of user motivations in chemistry and environmental science. Medicine and environmental science were found to have similar patterns in user motivations as they directly impact the general public. Chemistry, on the other hand, showed a peculiar pattern with a high percentage of self-citation and promotion. From this study, the domain is also found to play a vital role in measuring research impacts when alternate metrics are used.

Keywords: Altmetrics · Machine learning · Support Vector Machine ·
Medicine · Twitter · User motivation · Chemistry · Environmental science

1 Introduction

Altmetrics aim to gauge web-based scholarly discussions, essentially how research is tweeted, blogged about, discussed or bookmarked [9]. These non-traditional metrics help to measure the reach of scholarly work through social media activity. Robinson-García, Torres-Salinas, Zahedi, and Costas [22] revealed that out of all the social media mentioned articles, 87% were discussed on Twitter. The extensive use of Twitter by users worldwide to share, discuss and communicate research work and scientific findings help to determine the research impact among the general public. Furthermore, it is very important to capture the opinion and understanding of the research findings in

domains like medicine and biomedicine [7]. The discoveries in these domains rapidly disseminate to a larger audience of non-specialists in the form of health advice and recommendation [1, 6]. It is observed that authors and publishers tend to cite or promote their own work on Twitter. Sometimes their friends and organizations share, like, retweet their work on social media platforms. Often, tweets sharing research work are generated artificially using bots [21]. All these methods increase the citation count on altmetric.com (a service provider which tracks and analyzes the online activity around scholarly research outputs), but do not accurately describe the research impact. Therefore, it is imperative to explore the attitude and motivation of users citing research articles on Twitter to measure the real article impact and usefulness.

With our research study, we aim to analyze the twitter messages (tweets) and understand the motivation of these tweet authors. We build a machine learning model which can label the motivations of tweets by analyzing tweets and other relevant features. This is an extension to the earlier study [16] where a small dataset of around 3,000 tweets were manually labelled into 4 major categories, Discussion, Sharing, Promotion and Access, using content analysis research method. To have more refined motivations across domains, we have split these categories into 6 categories namely Access, Expressing Opinion, Interaction, Promotion, Sharing and Summarization. Our research aims to automate the labelling process by building a machine learning model that aids in answering the following research questions:

- (1) What are the motivations of users tweeting about research articles in the field of medicine?
- (2) How does the distribution of motivations vary across domains like medicine, chemistry and environmental science? Are the results generalized or domain specific?

2 Literature Review

With the rapid growth of social media, researchers are drawing attention to their work through social networking websites and their scientific outreach is no longer limited to the traditional academia [4]. In conjunction with conventional citation impact metrics such as Journal Impact Factor (JIF) and h-index, altmetrics that track the interaction of online items are used to measure research impact [18, 20].

Altmetrics as alternative metrics to assess the impact of the articles are under scrutiny as web-based metrics can be easily manipulated for self-promotion and commercialization of ‘likes’ and ‘shares’ on the social websites [19]. To determine the reliability and validity of altmetrics as supplementary metrics to citation count, various analysis and survey were conducted to gain an in-depth understanding of the processes that lead to sharing and mentioning of research publications on social media platforms.

Shema, Bar-Ilan, and Thelwall [24] studied the motivation for blogs citing research articles in the health category. It was inferred that 90% of the posts discussed the findings of the articles and bloggers rarely promoted their articles in their posts. Na and Ye [17] examined the validity of altmetrics as an impact factor by analyzing the scholarly interactions on Facebook. A content analysis on 2,783 posts citing psychological articles

revealed that nearly half the posts were simple sharing of articles with no user content while 20.4% of the posts were motivated by an intent to discuss and evaluate the articles. Self-citation and exchange of data sources were the least motivation factors observed from this study.

Research articles are widely discussed on Twitter and it is proven to be a trending medium for networking and learnings [10]. This microblogging site is used by scholars to broadcast their research and scientific discoveries [14]. A content analysis by Na [16] was carried out on 2,016 tweets citing articles from the psychology discipline, which revealed that 52.88% of the tweets encapsulated the important inferences of the articles. This reveals the fact that apart from sharing the article, users attempt to ingest the content and communicate it through the fastest media, thus indicating that tweets are a means of transmitting noteworthy findings. The study also revealed that 11.8% of tweets contained only the link and title, 19.05% retweeted the articles and only 7% were tweets from authors and publishers promoting their work.

From the study of user profiles, Htoo and Na [11] revealed a piece of interesting information. It was found that 77% of the tweets mentioning research articles were from individual users, and 67% of the individual users were found to be non-academic. This concludes that altmetrics as a tool for measuring the impact of publications reflect societal impact on the general public and tend to be non-scholarly. Existing studies by Ross, Terras, Warwick, and Welsh [23] and Knight and Kaye [15] showed that the purpose of tweeting scholarly articles is to increase the professional reputation of researchers and expand their networks. Researchers are also using Twitter as a medium of communication [8] to share their publication [3].

From these studies, we observed various patterns in the distribution of user motivations for sharing research articles. Manually labelling the motivations of the tweets is an expensive task which serves as a major obstacle to extend the existing studies to a larger set of data. In this research paper, we develop a machine learning classifier that detects the motivations of tweets citing publications in the medicine domain. Then the classifier is applied to various domains (chemistry and environmental science), and it provides us valuable insights on inter-disciplinary differences in the distribution of motivations which will enable us to clarify the validity of altmetrics.

3 Methodology

3.1 Domain and Journal Selection

According to the Thomson-Reuters' journal citation report for the year 2016, together with the impact and influence metrics of the previous year's data, journals in the domain of medicine emerged with high impact factors. Based on the trending topics and the number of tweets for the same year, medicine was found to be one of the most tweeted-about topics on Twitter [25]. For this study, tweets from top 20 journals from medicine were collected for analysis. To study the pattern of motivational differences across domains, we chose two other domains, chemistry and environmental science, to make a comparative analysis with medicine.

Like the field of medicine, another domain with more general public followers is considered for our study. Since environmental science is among the top 10 highly ranked categories and the findings of the environment is widely being posted on Twitter, we chose to study the environmental science domain. A dissimilar domain which requires expert knowledge to grasp the findings is considered for our analysis. Considering the journal impact factor and altmetrics attention, the domain chemistry was chosen. The top 20 journals in the selected domains were sorted out based on their high Impact Factors (IF) and their ISSN numbers were collected from the Journal of Citation Index reports.

3.2 Data Collection

Research articles from the collected journals published in 2016 were fetched from the Web of Science database in the form of Digital Object Identifiers (DOIs). Altmetric identifiers corresponding to the DOIs for each article were collected by invoking Altmetric API. The altmetric data (i.e. the tweets and number of mentions in Twitter) of these articles along with the tweet author's username were then scraped from altmetric.com. For this study, tweets from a total of 2,538 articles were collected in medicine, 10,891 from chemistry and 8,153 from environmental science (see Fig. 1 for distribution of tweets). Tweets collected in their raw form contain noise and various language tweets. In order to make them more readable for the machine, the non-English and empty tweets were discarded.

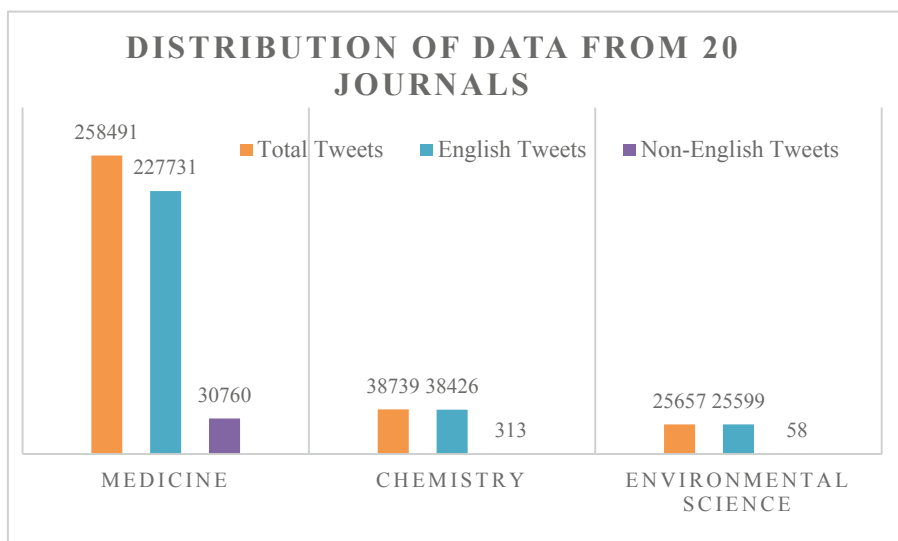


Fig. 1. Data distribution in Medicine, Chemistry and Environmental Science

3.3 User Motivation Analysis

We extend the work of Htoo and Na [11] to identify the key motivations of users and determine differences in the distribution of motivations among various disciplines. The various categories of motivations for this study along with their description, samples, and rules for detecting motivations are given in Table 1.

Table 1. Definitions and rules of different motivations

Motivation	Definition	Sample(s)	Rules
Sharing	Tweets that contain only the title and/or link to the article. These tweets do not express the opinion/experience of the user or summarize the findings of the articles. Tweets under this category may contain the full title, part of the title or some lines from the abstract. Retweets, Heard Through (HT) and MT are also considered under this category since they are sharing articles among the followers Priority: Low	- Ventricular Tachycardia Ablation versus Escalation of Antiarrhythmic Drugs. [URL] - RT @[TH]: Animated illustration explains Von Willebrand factor synthesis. [URL]	- Presence of: • Only the URL of a research article in the tweet, or • RT @ or HT @ - An exact match between the tweet and title of the research article
Summarization	When the user attempts to understand the information provided in the paper and summarizes the findings, the tweet falls under ‘Summarization’ category. These tweets are usually phrased using the users’ own words Priority: Medium	- Acetaminophen and Ibuprofen in Young Children with Mild Persistent #Asthma have similar outcomes [URL] - No diff in exacerbation rate in 1–5 yo’s w/mild persistent #asthma using acet. for pain vs ibuprofen for 1yr #NEJM [URL]	- Tweets containing textual content from the paper title or Abstract with less than 100% match or rephrased by the user without the usage of opinion words
Expressing Opinion	Tweets expressing the user’s opinion, sentiment, preference, approval, recommendation, criticism, and experience are classified under ‘Expressing Opinion’ Priority: Medium	- Great research! [URL] - This might be a further argument for using this in obesity treatment as well. [URL]	- Presence of: • Negative or positive opinion words in the tweet: e.g., “good read”, “not impressive results”, “a breakthrough”, etc. • Modals that express possibility: e.g., “could”, “might”, “may”, etc.

(continued)

Table 1. (continued)

Motivation	Definition	Sample(s)	Rules
Interaction	Tweets that tend to raise a question, promote conversation with other users or response to other tweets are labelled as ‘Interaction’. They promote discussion in the forum and have a deeper level of engagement with the articles’ content Priority: Medium	- What are the cardiovascular effects of #liraglutide, used in Rx of #diabetes? [URL] @[TH] - @[TH] This is the article I mentioned about. [URL]	- Presence of: Tag: e.g., @[TH] (but if the tagged person is the paper author or publisher, it is not interaction) The question mark “?” in order to probe the question: e.g., “what do you think of this article?”
Promotion	Tweets from the authors of the article or its publisher for promoting their work fall under the ‘Promotion’ category Priority: High	- Our recent work here quantifying monogenic risk for heart attack [URL]	- When the tweet author, i.e. the username of the twitter account, is either the author of the article or Publisher of the article
Access	Tweets that provide information for users to access the papers are classified under the ‘Access’ category. Users also tend to request access for articles Priority: High	- Here’s the full text of the new study about birth defects and Zika [URL]	- Presence of following phrases in tweets with the link to the article: “here”, “here it is”, “open access”, “full paper”, “full trial or ft”, “full text”, “full access”, “link”, or “links”

Note: [URL] typically refers to a paper url, and [TH] refers to a Twitter handle or username

4 Data Preparation

4.1 Manual Labelling and Inter-coder Reliability

A total of 5,839 tweets in the domain of medicine were labelled manually by 4 coders after removing non-English tweets and null values. Defined rules and their priority that were used to label the motivation of each tweet is given in Table 1. Among the collected tweets, 400 tweets were labelled initially by the 4 coders. These labels were validated by verifying each other’s labels. Conflicts in the labels were resolved by refining the rules for labelling and prioritizing the infrequently occurring categories to balance the distribution of motivation. Access and promotion were given higher priority as these two classes had the least number of tweets. Sharing was given the lowest priority as retweets and tweets with only URLs of the articles were more in number. Other categories such as Expressing Opinion, Summarization and Interaction were assigned medium priority as they had almost equal number of tweets. Inter-coder reliability between each pair of coders was evaluated and the average Cohen’s kappa value of 0.82 was obtained. These refined rules were then used to label the remaining tweets. With having considerable reliability among the coders, the tweets required for training the model was coded individually.

4.2 Data Pre-processing

Sentences in a tweet were tokenized on whitespace characters while emojis and emoticons present in the reviews were retained as they are often used to express an opinion on social media. Non-ASCII characters like © and £ were removed as they add noise to the corpus. Auxiliary verbs, conjunctions and articles appear frequently in most of the documents [13] were removed. Except the word ‘here’, whose presence in the tweet was used to detect a motivation class, the other stop words were removed. URLs in the tweet were replaced with the tag <URL>. Punctuation marks were also removed from the tweets except angle brackets, exclamation mark, at sign, period or full stop, comma and question mark. These punctuations are retained as their presence in the tweets was significant for classification. User mentions like ‘@WHO’ and ‘@XYZ’ in the tweet were then replaced with the keyword <TAG>. All the words in the tweet matching the title of the research article were removed from the tweet before creating features. This was done to ensure that sentiment words present in the title of the paper like “harmful effect” do not influence the sentiment of the tweet.

5 Feature Engineering

Features specific to each motivation given in Table 2 were used along with basic document and opinion related features.

Table 2. Motivation specific features

Feature	Observations from tweets	Relevant motivation
IsRetweet	Presence of “RT @” and “HT @” in the tweet means retweet (True or False)	Sharing
IsTag	Presence of <TAG> in the tweet (True or False)	Interaction
hasURL	Presence of <URL> in the tweet (True or False)	Sharing
IsFromPub	Tweet author is a publisher. (True or False)	Promotion
IsFromAuthor	Tweet author is a co-author of the article. (True or False)	Promotion
Tweet_title_match	Percentage match between the tweet and title of the article using the FuzzyWuzzy library (https://github.com/seatgeek/fuzzywuzzy). The words matching the title in the tweet are removed to prevent retaining emotion words in the tweet	Summarization
Tweet_abstract_match	Percentage match between the tweet and abstract using FuzzyWuzzy	Summarization
IsQuestion	Presence of “?” question mark in the tweet (True or False)	Interaction
IsAccess	Presence of phrases like “here”, “here it is”, “open access”, “full paper”, “full text”, “full trial”, “full access”, and “link” (True or False)	Access

5.1 Additional Features

- *Character count*: Motivations like Summarization and Expressing Opinion were observed to have longer tweet length. Since tweet length varies with motivation, the character count is added as an additional feature.
- *Word count*: The number of individual words in tweets with motivations of Summarization and Expressing Opinion was found to be greater than those in Sharing, Promotion and Access. Hence word count is added as an additional feature.
- *Word density*: The average number of characters per word is computed as word density and is added as a feature.
- *Upper case word count*: Users often use upper case words to highlight or stress on things they say. Usually, these tweets have a negative sentiment and count of such words can help the machine identify the sentiment of tweets.
- *Exclamation count*: The exclamation mark is very common with social media users. It is extensively used to express sentiment on Twitter like “WOW!!!!” and “Oh my God!!!!”.
- *Question mark count*: Tweets with question marks representing doubt and scepticism among users come across as a negative sentiment. The count of question marks in the tweet is important to understand the sentiment and hence is added as an additional feature.
- *Emoticons count*: As emoticons are widely used to express sentiment on social media platforms, the number of emojis or emoticons in the tweet will be crucial for detecting the sentiment.
- *Sentiment word count*: Using both the sentiment words obtained during manual labelling and SentiWordNet lexicon [5], the sentiment word count was calculated.
- *Subjectivity Indicator count*: As the presence or absence of adjective and adverbs in the text is a strong indicator of subjectivity, the predefined parts-of-speech from Penn Treebank tag set is used to determine the count of adjectives and adverbs.
- *Domain-specific polarity score*: Tweets contain a lot of slang words, emoticons, and acronyms. A lexicon and rule-based sentiment analysis tool, VADER (Valence Aware Dictionary and sEntiment Reasoner) [12], is used to compute the sentiment score of tweets.

6 Results and Discussion

6.1 Machine Learning Models

For the automated tweet classification, we experimented Naïve Bayes, Logistic Regression, Support Vector Machine and Ensemble boosting text classification algorithms which have been proven to work well in other natural language processing use cases [13]. 75% of the data was utilised for training the classifier and the remaining for testing its performance. The distribution of motivation in the two sets is shown in Table 3.

The best results of the four text mining models along with different document vectors are tabulated in Table 4. Among the best performing models, Support Vector

Machine was chosen and it was fine-tuned on its hyperparameters. Support Vector Machine with TF-IDF document vectors along with Radial Basis Function (RBF) kernel gave 85.2% accuracy. As there are six classes for prediction and they are imbalanced, weighted F1 score is used as an evaluation metric [2]. The results of the model are presented in Table 5.

Table 3. Distribution of Motivation in Training and Test set

Motivation	Training set	Test set
Sharing	1,648	550
Interaction	821	274
Summarization	772	258
Expressing Opinion	697	232
Promotion	323	107
Access	118	39

Table 4. Classification results of various models

Classifier	Validation accuracy (in %)
Naïve Bayes with Unigram Term Frequency (TF) Vector	63.42
Logistic Regression with N-gram TF-IDF Vector	71.78
Support Vector Machine with N-gram TF-IDF Vector	73.76
XGBoost with Unigram TF-IDF Vector	71.98

Table 5. Classification results of the SVM model

	Precision	Recall	F1-score	Support
Access	0.77	0.85	0.80	39
Expressing Opinion	0.76	0.70	0.73	232
Interaction	0.88	0.93	0.91	274
Promotion	0.98	0.92	0.95	107
Sharing	0.94	0.93	0.93	550
Summarization	0.69	0.72	0.70	258
Weighted average	0.85	0.85	0.85	1460

Upon analyzing the results, the model identified 85% of the Access category, 93% of Interaction and Sharing categories correctly (recall). Also, 92% of Promotion were identified accurately by the machine. Expressing Opinion and Summarization are the categories which were mostly misclassified by the model. The model identified 70% of Expressing Opinion category and 72% of Summarization category. The behaviour exhibited is in line with the results of manual coding where the human coders found it difficult to draw a fine distinction among these two categories. Since there is a borderline difference in the tweets where the users use opinion words to summarize the

contents of the article, there seem to be higher misclassification rates in this category and the same is also observed on evaluation. For example, tweets with phrases like “improved performance” in abstract are tagged as Expressing Opinion but they are Summarization.

6.2 Prediction of Unlabelled Data

The model is then applied over the unlabelled data set in three of the domains chosen for the study (see Fig. 2 for prediction of the model across domains).

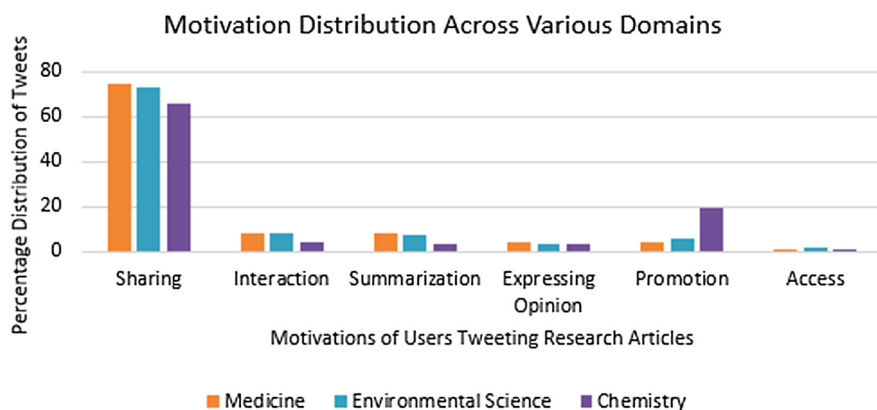


Fig. 2. Motivation distribution across the three domains

6.3 Evaluation on Medicine

The results of the prediction on medicine confirmed the existing belief that most people are tweeting with the intention to share new findings and processes with others. It was found that 74.3% of the tweets were only with the title of the article, link to the article and retweets which are intended for sharing, 8.3% of the tweets were with user mentions and questions, 7.9% of the tweets were posted with an intent to summarize the contents of the research article, 4.2% of the tweets represented sentiments the user had towards the article or as a means to praise or criticize the work, 3.8% of the tweets were made by the authors of the paper or the publisher of the journal to promote their work and 1.3% of tweets were made to seek or grant access to the research article. Although tweets were made with the motive of sharing the information with fellow users, there seems to exist interesting patterns where a considerable number of users not just shared information as such but also discussed the findings with others (Interaction). There also seems to exist a considerable number of users who gave an opinion about the articles and expressed their own insights on the content.

6.4 Evaluation on Chemistry

The trend of tweet motivation was quite different in chemistry where a significantly larger number of users used Twitter to publicize their research work. Promotion stood as the second highest factor for motivation as 19% of tweets were intended to promote the article and they were either from the publisher or from the authors. This interesting result emerged in this domain which doesn't seem to exist in the domain of medicine. The percentage of tweets sharing and retweeting the title and link of the article stands a clear majority in chemistry with 65.7%. Interaction in this domain holds 6.3% share. Summarization and Expressing Opinion attribute to 3.6% and 3.8% respectively. As observed in the medicine domain, people posting tweets to gain or provide access to research material is very less which is 1% in this case.

6.5 Evaluation on Environmental Science

From the results, it is apparent that environmental science follows a similar trend of the motivations of the medicine domain. Sharing continues to top the results with 73.3% of the tweets, and the results from this domain were in line with our assumption as more people tweeted about environmental science with motivations to interact with others and to summarize the findings with 8.3% and 7.6% respectively. Promotion and Expressing Opinion were not at considerable rates in this domain with only 5.5% and 3.2% share respectively and followed by Access with 1.9%. The model correctly classified unlabeled tweets, like "Did you know? <https://xxx>" and "This is a problem that isn't going away. <https://xxx>. #Cleanwater" as Interaction and Expressing Opinion respectively.

6.6 Domain-Motivation Relationship

In order to determine the existence of a correlation between article domains and tweet motivations, we performed a chi-square test with the following hypothesis:

H0 or Null Hypothesis: There is no statistical dependence between the tweet motivations and the domains of interest.

H1 or Alternate Hypothesis: There is statistical dependence between the tweet motivations and the domains of interest.

A value of 0.05 (α) is used as a level of significance to accept or reject the null hypothesis. Degree of Freedom is taken as 10 for this Chi-Square test of independence.

Table 6. Chi-square and P-value

	CHI Square Value (X2)	p-value
Medicine	2099.082	0.00**
Chemistry	9808.073	0.00**
Environmental science	156.552	0.00**

**Significantly small and very close to zero

From the Chi-square distribution of the domains for our study from Table 6, it is observed that the probabilities (p-values) associated with the computed Chi-square values are extremely small and close to zero compared to the significance level α and therefore the null hypothesis is rejected, and the alternate hypothesis holds good. Thus, it can be deduced that the motivational differences are dependent on the domains and the motivations vary with the domains. Though all the three domains displayed a significantly higher number of tweets under the ‘Sharing’ category and few tweets under Access category, the percentage of tweets in the other categories differed fairly across the three domains. For instance, Promotion, the second lowest category in both medicine and environmental science, was the second highest in chemistry. This peculiar pattern was unique to the chemistry domain.

6.7 Inferences

The results of this study emerge in a new dimension of patterns with respect to the altmetrics usage in research articles.

- In the previous study on the motivations of bloggers citing general and multi-disciplinary medical journals, discussions about health issues, social phenomena and research outcomes were found to be the main motivation. However, in our study carried out on large-scale Twitter data representing real-world scenario, the main motivation was found to be sharing the articles without user’s perceptions or opinions. This observation holds for all the domains which we have analyzed.
- From the volume of data obtained from the top 20 journals in the three disciplines, we found a notably high amount of Twitter data in medicine. As the outcomes of new research in medicine are beneficial to the general public, it is presumed to be a widely tweeted discipline. We also noted a considerably high volume of non-English tweets in medicine compared to other disciplines chosen for our analysis. From this observation, we infer that medicinal researches are widely disseminated across the globe.
- The motivational trend in medicine conveys that more people are using Twitter to share research breakthroughs followed by the users with the motivations to interact and express their opinion or emotion. From this pattern, it is inferred that the general public is highly involved in discussions such as methodological issues, lifestyle advises and treatment recommendations. As there are extensive on-going discussions in medicine, there is only a marginal proportion of authors or publishers promoting their work.
- As the discipline of chemistry has a less direct impact on people’s lives, the general public interaction in this domain is found to be minimal. We also observed a comparatively high proportion of promotion and this implies that authors and publishers are promoting their work for wider outreach.
- A comparative study of medicine and environmental science provides a common pattern in motivations among the users as both the disciplines share direct benefits to the general public.
- From the inferences, we can say that the motivations differ across domains and the research mention counts in tweets cannot be solely considered as a factor for

measuring research impact. Along with existing metrics for assessing research impact, considering the domain and its underlying motivation distribution plays a major role in accurately measuring the research impact.

7 Conclusion

With this study, we have developed a machine learning model to predict the motivations of the users tweeting about research articles. This study investigated tweets and relevant data associated with research articles like publisher name, author, abstract, title, etc. from altmetric.com. Manually labelled tweets from the medicine domain were used as a training dataset for the model. SVM outperformed other algorithms and achieved an accuracy of 85.2% with the test dataset. This model was applied to unlabelled tweets from medicine, chemistry and environmental science domains. On analyzing the motivational distribution, medicine was found to have more tweets with sharing, and very few with promotion and access. In chemistry, a greater number of tweets belonged to Sharing and Promotion. Environmental science, being a familiar discipline with the general public outreach, exhibited similar patterns in motivations as those of medicine. Chi-squared test on the results obtained for the three domains showed statistical differences between domains and motivations. From this study, we conclude that in addition to raw counts of Twitter mention, domain and its underlying motivation distribution should also be considered when measuring the research impact.




References

1. Campion, E.W.: Medical research and the news media. *New Engl. J. Med.* **351**(23), 2436–2437 (2004). <https://doi.org/10.1056/nejme048289>
2. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **16**, 321–357 (2002). <https://doi.org/10.1613/jair.953>
3. Collins, K., Shiffman, D., Rock, J.: How are scientists using social media in the workplace? *PloS One* **11**(10), e0162680 (2016). <https://doi.org/10.1371/journal.pone.0162680>
4. Erdt, M., Aung, H.H., Aw, A.S., Theng, Y.L., Rappale, C.: Analysing researchers' outreach efforts and the association with publication metrics: a case study of Kudos. *PloS One* **12**(8), e0183217 (2017). <https://doi.org/10.1371/journal.pone.0183217>
5. Esuli, A., Sebastiani, F.: SentiWordNet: a publicly available lexical resource for opinion mining. In: *The LREC*, pp. 417–422 (2006). 10.1.1.61.7217
6. George, D.R., Rovniak, L.S., Kraschnewski, J.L.: Dangers and opportunities for social media in medicine. *Clin. Obstet. Gynecol.* **56**(3) (2013). <https://doi.org/10.1097/grf.0b013e318297dc38>
7. Halevi, G., Schimming, L.: An initiative to track sentiments in altmetrics. *J. Altmetrics* **1**(1) (2018). <https://doi.org/10.29024/joa.1>
8. Haustein, S., Bowman, T.D., Holmberg, K., Peters, I., Larivière, V.: Astrophysicists on Twitter: an in-depth analysis of tweeting and scientific publication behavior. *Aslib J. Inf. Manag.* **66**(3), 279–296 (2014). <https://doi.org/10.1108/ajim-09-2013-0081>

9. Howard, J.: Scholars Seek Better Ways to Track Impact Online. *Chronicle of Higher Education* (2012)
10. Hsu, Y.-C., Ho, H.N.J., Tsai, C.-C., Hwang, G.-J., Chu, H.-C., Wang, C.-Y., Chen, N.-S.: Research trends in technology-based learning from 2000 to 2009: a content analysis of publications in selected journals. *J. Educ. Technol. Soc.* **15**(2), 354–370 (2012)
11. Htoo, T.H.H., Na, J.-C.: Who are Tweeting Research Articles and Why? (2017). <https://doi.org/10.1633/jistap.2017.5.3.4>
12. Hutto, C.J., Gilbert, E.: Vader: a parsimonious rule-based model for sentiment analysis of social media text. In: the Eighth International AAAI Conference on Weblogs and Social Media (2014)
13. Ikonomakis, M., Kotsiantis, S., Tampakas, V.: Text classification using machine learning techniques. *WSEAS Trans. Comput.* **4**(8), 966–974 (2005)
14. Ke, Q., Ahn, Y.-Y., Sugimoto, C.R.: A systematic identification and analysis of scientists on Twitter. *PloS one* **12**(4), e0175368 (2017). <https://doi.org/10.1371/journal.pone.0175368>
15. Knight, C.G., Kaye, L.K.: ‘To tweet or not to tweet?’ A comparison of academics’ and students’ usage of Twitter in academic contexts. *Innov. Educ. Teach. Int.* **53**(2), 145–155 (2016). <https://doi.org/10.1080/14703297.2014.928229>
16. Na, J.-C.: User motivations for tweeting research articles: a content analysis approach. In: *The International Conference on Asian Digital Libraries*, pp. 197–208 (2015). https://doi.org/10.1007/978-3-319-27974-9_20
17. Na, J.-C., Ye, Y.E.: Content analysis of scholarly discussions of psychological academic articles on Facebook. *Online Inf. Rev.* **41**(3), 337–353 (2017). <https://doi.org/10.1108/oir-02-2016-0058>
18. University of Pittsburgh: Altmetrics: What are Altmetrics? <https://pitt.libguides.com/altmetrics>
19. Priem, J., Hemminger, B.H.: Scientometrics 2.0: new metrics of scholarly impact on the social Web. *First Monday* **15**(7) (2010). <https://doi.org/10.5210/fm.v15i7.2874>
20. Priem, J., Taraborelli, D., Groth, P., Neylon, C.: Altmetrics: a manifesto. 2010 (2016)
21. Robinson-Garcia, N., Costas, R., Isett, K., Melkers, J., Hicks, D.: The unbearable emptiness of tweeting—About journal articles. *PloS One* **12**(8), e0183551 (2017). <https://doi.org/10.1371/journal.pone.0183551>
22. Robinson-Garcia, N., Torres-Salinas, D., Zahedi, Z., Costas, R.: New data, new possibilities: exploring the insides of Altmetric. com. *arXiv preprint arXiv:1408.0135* (2014). <https://doi.org/10.3145/epi.2014.jul.03>
23. Ross, C., Terras, M., Warwick, C., Welsh, A.: Enabled backchannel: Conference Twitter use by digital humanists. *J. Doc.* **67**(2), 214–237 (2011). <https://doi.org/10.1108/002204111111109449>
24. Shema, H., Bar-Ilan, J., Thelwall, M.: How is research blogged? A content analysis approach. *J. Assoc. Inf. Sci. Technol.* **66**(6), 1136–1149 (2015). <https://doi.org/10.1002/asi.23239>
25. Zawbaa, H.: 2017 Journal Citation Reports by Thomson Reuters (2017). <https://doi.org/10.13140/rg.2.2.31536.87049>



Understanding the Twitter Usage of Science Citation Index (SCI) Journals

Aravind Sesagiri Raamkumar^(✉) , Mojisola Erdt ,
Harsha Vijayakumar, Aarthi Nagarajan, and Yin-Leng Theng 

Wee Kim Wee School of Communication and Information,
Nanyang Technological University, Singapore 637718, Singapore
{aravind0002, mojisola.erdt, hvijayakumar, aarthyn,
tyltheng}@ntu.edu.sg

Abstract. This paper investigates the Twitter interaction patterns of journals from the Science Citation Index (SCI) of Master Journal List (MJL). A total of 953,253 tweets extracted from 857 journal accounts, were analyzed in this study. Findings indicate that SCI journals interacted more with each other but much less with journals from other citation indices. The network structure of the communication graph resembled a tight crowd network, with Nature journals playing a major part. Information sources such as news portals and scientific organizations were mentioned more in tweets, than academic journal Twitter accounts. Journals with high journal impact factors (JIFs) were found to be prominent hubs in the communication graph. Differences were found between the Twitter usage of SCI journals with Humanities and Social Sciences (HSS) journals.

Keywords: Twitter · Journals · Science Citation Index · Social network analysis · Social media

1 Introduction

The scientific community has been interested in Twitter due to the scope for extended information dissemination and promotion of research [1]. To enable discussions about research papers, academic journals use Twitter [2–4]. Furthermore, journals with social media accounts (e.g., Twitter) have been found to have higher research metrics [5]. The current exploratory study attempts to compliment two earlier Twitter studies conducted with tweets posted by academic journals. In the first study [6], the overall social media presence of academic journals was investigated, specifically for Twitter and Facebook platforms. In the second study [7], the Twitter usage of Arts & Humanities Citation Index (AHCI) and Social Science Citation Index (SSCI) journals was reported. This study reports the Twitter interaction dynamics of Science Citation Index (SCI) academic journals indexed in the Master Journal List (MJL). In the MJL, the journals are indexed in three main citation indices namely SCI, AHCI and SSCI. The following research questions are investigated in this study.

RQ1: Does the communication between journals in Twitter conversations of SCI journals differ from HSS (AHCI & SSCI) journals?

RQ2: Which network structure best represents the SCI journals Twitter communication graph?

RQ3: What type of Twitter accounts are top authorities in the communication graph of SCI journals?

RQ4: What disciplines do the top SCI journals in the communication graph represent and does their Twitter popularity reflect their ranking in the Journal Citation Report (JCR)?

2 Methodology

2.1 Base Dataset

A subset of the dataset used in a previous study [6] was used for this current study. The dataset preparation is described as follows. Journal titles from SCI were extracted from the MJL. After manually identifying the Twitter accounts of the journals, the tweets were extracted using the Twitter API. The basic Twitter API permitted extraction of only 3,000 tweets for each Twitter account during the time of data collection. Out of the 8,827 SCI journals in MJL, 857 were found to have Twitter accounts and 953,253 tweets were extracted for these journals.

2.2 Data Preparation

The data for addressing the four research questions was prepared through the following process. In this paper's context, the term 'mention' refers to a Twitter account tagged in a tweet. The tweets that contained mentions were shortlisted for the next step. From these tweets, the parent Twitter account name (journal) and the mention(s) were extracted. A combination of Twitter account name and the mention is referred to as a 'conversation' in this paper. It is to be noted that a single tweet could contain multiple mentions hence there could be multiple conversations within a tweet. Using the conversations data from the previous step, a Twitter communication graph was built. Gephi [8] was used to build this graph. In the communication graph, the source node is always the journal Twitter account whereas the target node is the mention (a Twitter account which is not necessarily a journal). The nodes with a degree value of one were filtered out, for handling the issue of sparsity. After building the graph, the community detection algorithm ForeAtlas2 was used in Gephi to identify the different communities in the graph. In this algorithm, communities are identified based on the frequency of interactions between the source and target nodes of the graph. A sub-graph of a particular set of nodes that interact more with each other than with other nodes, is regarded collectively as a community. The top 20 nodes (Twitter accounts) with the highest in-degree values and out-degree values in the communication graph were identified as a part of this step. The in-degree of a graph is the number of edges where the node is a target node while the out-degree of a node is the number of edges where the node is the source node. A node with a high in-degree value is referred to as an 'authority' while a node with a

high out-degree value is referred to as a ‘hub’. The nodes with the highest out-degrees (hubs) are the Twitter accounts managed by journals whereas the nodes with the highest in-degrees (authorities) could be any Twitter account.

3 Results

3.1 Twitter Conversations

SCI journals’ conversation stats are listed in Table 1. The number of conversations in SCI journals was high ($n = 509,062$) corresponding to the high number of tweets (refer Sect. 2.1). This metric was nearly five times more than AHCI ($n = 103,181$) and SSCI ($n = 121,099$) journals. To determine the interaction levels with other journals, we identified the mentions which were journal accounts (data listed in rows D, E, F of Table 1). SCI journals had the highest percentage of conversations at an intra-index level (18.16%). In [7], it was identified that HSS journals intra-index conversations had comparatively lower percentages (13.39% for AHCI, 13.25% for SSCI).

Table 1. SCI journals’ Twitter conversation statistics

Entity	n
Conversations [A]	509,062
Unique journal accounts [B]	768
Unique user-mentions [C]	88,021
Conversations where user-mentions are SCI journals [D]	92,421 (18.16% of A)
Conversations where user-mentions are AHCI journals [E]	88
Conversations where user-mentions are SSCI journals [F]	592

3.2 Communication Graph

The communication graph¹ generated for the SCI journals is illustrated in Fig. 1. The node color represents the parent community of a node. The size of the nodes was set according to the betweenness centrality values of the nodes. Representative labels have been added for each community based on the commonality of topics². Communities comprised of journal accounts belonging mainly to the disciplines of Chemistry, Bioscience, Ecology and Surgery. Technology & Engineering formed a very small community in this graph. Interestingly, there was one community which was entirely oriented towards journals from the Nature Publishing Group (NPG). The SCI graph structure resembles a “tight crowd” [9] mainly because of the central presence of Nature journals across many communities. This type of network has been observed in other related Twitter scholarly communication studies [10]. Unlike the SCI graph, the communities in the AHCI and SSCI graphs were represented by a greater number of

¹ Higher resolution image can be viewed at this link <https://bit.ly/2KP08EH>.

² The topics were identified based on the profile description in the Twitter accounts.

topics [7]. Both these graphs were classified as a “community clusters” due to the presence of multiple communities and minimal interspersed nodes.

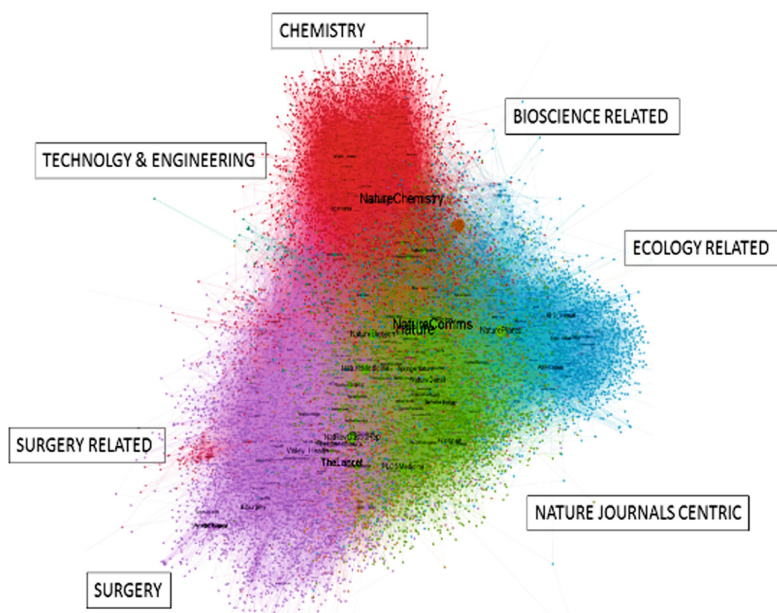


Fig. 1. SCI Twitter communication graph.

3.3 Top Authorities and Hubs

In Table 2, the top 20 authorities (based on in-degree values) from the SCI communication graph are listed, along with the account types which had been identified for these Twitter accounts. The major presence of news portals as top authorities was noticed (e.g., Nature News, The Guardian), with these accounts occupying 25% ($n = 5$) of the top 20 authorities. Scientific organization was another authority type identified, which was as popular ($n = 5$) as news portals. We used this account type as an umbrella term covering different organizations related to academia and scientific research. The next most popular authority type identified was journal. SCI had the most number of journals ($n = 4$), whereas AHCI and SSCI had only one and two journals in the top 20 authorities list respectively in the earlier study [7]. Agencies were the next most popular authority type ($n = 3$) identified. These Twitter accounts included government and global agencies (WHO, NIH & CDC). Magazine Twitter accounts ($n = 2$) were not as popular as authorities for SCI journals, unlike HSS journals where this account type was quite prevalent (approximately 35% of the top 20) [7].

Table 2. Top 20 authorities in the SCI communication graph

Twitter handle	Account name	Account type	<i>in</i> *
NatureNews	Nature News & Comment	News	209
guardian	The Guardian	News	172
sciencemagazine	Science Magazine	Journal	165
nytimes	The New York Times	News	159
sciam	Scientific American	Magazine	157
WHO	World Health Organization	Specialized Agency	142
newscientist	New Scientist	Magazine	130
nature	Nature	Journal	127
altmetric	Altmetric	Scientific Organization	123
NIH	NIH	Government Agency	122
YouTube	YouTube	Video Sharing	114
wellcometrust	Wellcome Trust	Scientific Organization	112
MayoClinic	Mayo Clinic	Scientific Organization	109
NEJM	NEJM	Journal	104
physorg_com	Phys.org	News	102
UniofOxford	Oxford University	Scientific Organization	102
guardianscience	Guardian Science	News	102
royalsociety	The Royal Society	Scientific Organization	101
CDCgov	CDC	Government Agency	100
TheLancet	The Lancet	Journal	96

Note: *in** refers to in-degree value

In Table 3, the top 20 hubs (based on out-degree values) from the SCI communication graph are listed. It is well-known that the JIF is measured and indexed along with the corresponding quartiles in the JCR [11] every year. We have included the JIF quartile data for the top 20 hubs in Table 3. The JIF quartile indicates the popularity of the journal within the citation network. For instance, if a journal belongs to the first quartile, it is considered to be among the top 25 percentile of journals for the respective research subject. Journals from the disciplines of medicine ($n = 3$), biology ($n = 4$) and chemistry ($n = 4$) were prominent. It can be observed that most of the journals ($n = 14$) in this list were from the first quartile of their respective research subjects. This finding indicates that the top journals in SCI seem to be most active in Twitter. Interestingly, engineering journal accounts were not present in this list.

Table 3. Top 20 hubs in the SCI communication graph

Twitter handle	Journal name	<i>out</i> *	<i>JQ</i> [^]
BiolJLinnSoc	Biological Journal of the Linnean Society	835	3
BMCMedicine	BMC Medicine	769	1
ChemMater	Chemistry of Materials	731	1
acsnano	ACS Nano	716	1
BiochemJ	Biochemical Journal	714	2
clin_sci	Clinical Science	682	1
ChemCatChem	Chemcatchem	661	1
PLOSPathogens	PLOS Pathogens	642	1
NatureChemistry	Nature Chemistry	636	1
FunEcology	Functional Ecology	626	1
PLOSMedicine	PLOS Medicine	605	1
NaturePlants	Nature Plants	603	4
JExpMed	Journal of Experimental Medicine	576	1
eLife	eLife	574	1
ASIHCOPEIA	Copeia	543	2
NatureGenet	Nature Genetics	526	1
GenomeBiology	Genome Biology	510	1
BiosciReports	Bioscience Reports	506	3
cenmag	Chemical & Engineering News	485	4
TheLancetInfDis	Lancet Infectious Diseases	485	1

Note: *out** refers to out-degree value; *JQ*[^] refers to JIF quartile

4 Conclusion

In the current study's context, SCI journals were found to have a marginally better presence than HSS journals on Twitter. For RQ1, the conversation statistics revealed certain differences between the journals from SCI and HSS. In our earlier study [7], the structure of the communication graphs of both AHCI and SSCI journals largely resembled a 'community clusters' network where there were multiple communities with high intercommunication. In the current study, the bigger SCI graph resembled a 'tight crowd' network. We can conclude that there are differences at the conversational level in Twitter between hard (SCI) and non-hard (AHCI & SSCI) sciences for RQ1 and RQ2.

The top authorities and hubs in the communication graph were identified for RQ3 and RQ4. News portals and scientific organizations were amongst the most authoritative Twitter accounts. Magazines such as the New Scientist and Scientific American were also present in this authorities list. Compared to HSS journals, SCI had more number of journals as authoritative accounts. With the participation of different account types such as news portals, scientific organizations, journals, magazines, and government agencies, the involvement of key players in the scholarly communication lifecycle of SCI journals is clearly perceived. This is in contrast to the HSS journals' top

authorities lists which comprised mostly of news portals and magazines which cater to the general public [7]. Interestingly, three Twitter accounts were found to be present in the top 20 authority lists of both SCI and HSS journals. These accounts were Guardian, NY Times and YouTube. The Top 20 Twitter hubs list comprised almost entirely of natural sciences and medicine related journals. This finding highlights the prominent outreach activities of journals from these disciplines on Twitter. With regards to the question of ascertaining whether the top hubs in Twitter were also top journals in the citation network (represented by JCR rankings), data shows that SCI had mostly first quartile journals in the list, unlike SSCI journals where journals from all four quartiles were present in the list [7].

Acknowledgements. The research project “Altmetrics: Rethinking And Exploring New Ways Of Measuring Research” is supported by the National Research Foundation, Prime Minister’s Office, Singapore under its Science of Research, Innovation and Enterprise programme (SRIE Award No. NRF2014-NRF-SRIE001-019).

References

1. Mohammadi, E., Thelwall, M., Kwasny, M., Holmes, K.L.: Academic information on Twitter: a user survey. *PLoS ONE* **13**, e0197265 (2018)
2. Gardhouse, A.I., Budd, L., Yang, S.Y.C., Wong, C.L.: #GeriMedJC: the Twitter complement to the traditional-format geriatric medicine journal club. *J. Am. Geriatr. Soc.* **65**, 1347–1351 (2017)
3. Leung, E., Siassakos, D., Khan, K.: Journal club via social media: authors take note of the impact of #BlueJC. *BJOG Int. J. Obstet. Gynaecol.* **122**, 1042–1044 (2015)
4. Chan, T.M., et al.: Ten steps for setting up an online journal club. *J. Contin. Educ. Health Prof.* **35**, 148–154 (2015)
5. Wong, K., Piraquive, J., Levi, J.R.: Social media presence of otolaryngology journals: the past, present, and future. *Laryngoscope* **128**, 363–368 (2018)
6. Zheng, H., Aung, H.H., Erdt, M., Peng, T.-Q., Sesagiri Raamkumar, A., Theng, Y.-L.: Social media presence of scholarly journals. *J. Assoc. Inf. Sci. Technol.* **70**, 256–270 (2018)
7. Raamkumar, A.S., Erdt, M., Vijayakumar, H., Rasmussen, E., Theng, Y.-L.: Understanding the Twitter usage of humanities and social sciences academic journals. *Proc. Assoc. Inf. Sci. Technol.* **55**, 430–439 (2018)
8. Bastian, M., Heymann, S., Jacomy, M.: Gephi : an open source software for exploring and manipulating networks. In: *ICWSM*, vol. 8, pp. 361–362 (2009)
9. Smith, M.A., Rainie, L., Shneiderman, B., Himelboim, I.: Mapping Twitter topic networks: from polarized crowds to community clusters. *Pew Res. Center* **20**, 1–56 (2014)
10. Lee, M.K., Yoon, H.Y., Smith, M., Park, H.J., Park, H.W.: Mapping a Twitter scholarly communication network: a case of the association of internet researchers’ conference. *Scientometrics* **112**, 767–797 (2017)
11. Thomson Reuters: Journal Citation Reports (JCR). <https://jcr.incites.thomsonreuters.com/>

Scholarly Data Analysis and Recommendation



Research Paper Recommender System with Serendipity Using Tweets vs. Diversification

Chifumi Nishioka¹(✉), Jörn Hauke², and Ansgar Scherp³

¹ Kyoto University Library, Kyoto, Japan
nishioka.chifumi.2c@kyoto-u.ac.jp

² Kiel University, Kiel, Germany

³ University of Essex, Colchester, UK
ansgar.scherp@essex.ac.uk

Abstract. So far, a lot of works have studied research paper recommender systems. However, most of them have focused only on the accuracy and ignored the serendipity, which is an important aspect for user satisfaction. The serendipity is concerned with the novelty of recommendations and to which extent recommendations positively surprise users. In this paper, we investigate a research paper recommender system focusing on serendipity. In particular, we examine (1) whether a user's tweets lead to a generation of serendipitous recommendations and (2) whether the use of diversification on a recommendation list improves serendipity. We have conducted an online experiment with 22 subjects in the domain of computer science. The result of our experiment shows that tweets do not improve the serendipity, despite their heterogeneous nature. However, diversification delivers serendipitous research papers that cannot be generated by a traditional strategy.

Keywords: Recommender system · Scientific publication · Experiment

1 Introduction

Various works have developed recommender systems for research papers to help researchers overcome the information overload problem [3]. Recommendations are generated based on, e.g., a user's own papers [21] or the papers a user has accessed in the past [16]. Most of the previous works have focused only on the accuracy of recommendations, e.g., on nDCG. However, several works in recommender systems in other domains (e.g., movies) argue that there are important aspects other than the accuracy [10, 14]. One of these aspects is the *serendipity*, which is concerned with the novelty of recommendations and in how far recommendations may positively surprise users [6].

In this paper, we study a research paper recommender system focusing on the serendipity. Sugiyama and Kan [22] have investigated serendipitous research

paper recommendations focusing on the influence from dissimilar users and co-author network to the recommendation performance. In contrast, our study investigates the following research questions: (RQ1) Do a user’s tweets enable serendipitous recommendations? (RQ2) Is it possible to improve the serendipity in a recommendation list by diversification? We run an experiment to empirically investigate the two research questions using three factors. For RQ1, we employ the factor *User Profile Source*, where two different user profile sources are compared: a user’s own papers and the user’s tweets. User’s own papers are used in existing recommender systems such as Google Scholar¹. We assume that user’s tweets produce recommendations that cannot be generated based on own papers, since researchers tweet about very recent developments and interests that are yet not reflected in their papers (e.g., what they found interesting at a conference [12] or in the social network). In the domain of economics, recommendations based on a user’s tweets received a precision of 60%, which is fairly high [18]. In addition, we analyze the factor *Text Mining Method*, which applies different methods for computing user profiles from different content, e.g., tweets, research papers, etc. As text mining methods, we compare the classical TF-IDF [19] with two of its recent extensions that are known to perform well in recommendation tasks, namely CF-IDF [7] and HCF-IDF [17]. For RQ2, we introduce the factor *Ranking Method*, where we compare two ranking methods: classical cosine similarity and the established diversification algorithm IA-Select [2]. IA-Select ranks candidate items with the objective to diversify recommendations in a list. Since it broadens the coverage of topics in a list, we assume that IA-Select delivers serendipitous recommendations compared to cosine similarity.

Along with the three factors, we conduct an experiment with 22 subjects. The result of the experiment reveals that using the user’s tweets for making scientific paper recommendations did not improve the serendipity. On the other hand, we confirm that the diversification of a recommendation list by IA-Select delivers serendipitous recommendations.

The paper is organized as follows: Sect. 2 describes the recommender system and the experimental factors. The evaluation setup is described in Sect. 3. Section 4 reports and discusses the results before concluding the paper.

2 Recommender System and Experimental Factors

In this paper, we build a content-based recommender system along with the three factors *User Profile Source*, *Text Mining Method*, and *Ranking Method*. It works as follows: (a) Candidate items of the recommender system (i.e., research papers) are processed by one of text mining methods and paper profiles are generated. (b) A user profile is generated based on his/her user profile source (i.e., own papers or tweets) by a text mining method, which is applied to generate paper profiles. (c) One of ranking methods orders recommendations to determine which papers are suggested. The design of the experiment is illustrated in Table 1, where each

¹ <https://scholar.google.co.jp/>.

cell is a possible design choice in each factor. In the following paragraphs, we describe the details of each factor.

Table 1. Factors and their choices panning in total $3 \times 2 \times 2 = 12$ strategies.

Factor	Possible design choices		
<i>User Profile Source</i>	Twitter		Own papers
<i>Text Mining Method</i>	TF-IDF	CF-IDF	HCF-IDF
<i>Ranking Method</i>	Cosine similarity		IA-Select

User Profile Source. In this factor, we compare the data sources that are used to build a user profile: own papers and tweets. As baseline, we use the own papers of the users. This approach is motivated from existing research paper recommender systems such as Sugiyama and Kan [21] and Google Scholar. In contrast, we assume that using tweets provide more serendipitous recommendations. It is common among researchers to tweet about their professional interests on Twitter [9, 12]. Thus, tweets can be used for building a user profile in the context of research paper recommender system. We hypothesize that a user’s tweets produce serendipitous recommendations, since researchers tweet about their most recent interests and information that are (yet) not reflected in their papers.

Text Mining Method. For each of the two sources on content, i.e., the user’s own papers or his/her tweets, we apply a profiling method using one of three text mining methods. This constitutes the second factor of the study. We compare three methods TF-IDF [19], CF-IDF [7], and HCF-IDF [17] to build paper profiles and a user profile. Since text mining method that works well differs depending on the type of contents to be analyzed (e.g., tweets, research papers), we introduce this factor. Thus, this factor is integrated into RQ1. First, we use TF-IDF [19], which is often used in recommender systems as baseline [7, 13]. Second, Concept Frequency Inverse Document Frequency (CF-IDF) [7] is an extension of TF-IDF, which replaces terms with semantic concepts from a knowledge base. The use of a knowledge base decreases noise in profiles [1, 8]. In addition, since a knowledge base can store multiple labels for a concept, synonyms are integrated into one feature. Finally, we apply Hierarchical Concept Frequency Inverse Document Frequency (HCF-IDF) [17], which is an extension of CF-IDF. It applies a propagation function [11] over a hierarchical structure of a knowledge base to give a weight to concepts in higher levels. Thus, it identifies concepts that are not mentioned in a text but highly relevant. HCF-IDF calculates a weight of a concept a in a text t as: $w(a, t) = BL(a, t) \cdot \log \frac{|D|}{|\{d \in D : a \in d\}|}$. $BL(a, t)$ is a propagation function Bell-Log [11], which is defined as: $BL(a, t) = cf(a, t) + FL(a) \cdot \sum_{a_j \in pc(a)} BL(a_j, t)$, where $FL(a) = \frac{1}{\log_{10}(nodes(h(a)+1))}$. The function $h(a)$ returns the level, where a

concept a is located in the knowledge base. $nodes$ provides the number of concepts at a given level in a knowledge base. $pc(a)$ returns all parent concepts of a concept a . We employ HCF-IDF, since it showed to work well for short texts such as tweets [18].

Ranking Method. Finally, we rank all candidate items to determine which items are recommended to a user. In this factor, we compare two ranking methods: cosine similarity and diversification with IA-Select [2]. As baseline, we employ a cosine similarity, which has been widely used in content-based recommender systems [13]. Top- k items with largest cosine similarities are recommended. Second, we employ IA-Select [2], in order to deliver serendipitous recommendations. IA-Select diversifies recommendations in a list to avoid suggesting similar items together. Although it has been originally introduced in information retrieval [2], it is also used in recommender systems to improve the serendipity [24]. The basic idea of IA-Select is that it lowers iteratively the weights of features in the user profile, which are already covered by selected recommended items. First, IA-Select computes cosine similarities between a user and each candidate item. Subsequently, it adds an item with the largest cosine similarity to the recommendation list. After picking the item, IA-Select decreases weights of features covered by the selected item in the user profile. These steps are repeated until k recommendations are determined.

3 Evaluation

We conduct an online experiment with $n = 22$ subjects to answer the two research questions. The setup of the experiment follows the previous works [5, 18]. In the subsequent paragraphs, we describe the procedure of the experiment, subjects, dataset, and metric, respectively.

Procedure. We have implemented a web application where human subjects evaluate the twelve recommendation strategies. First, subjects start on the welcome page, which asks for consent to the data collection. Thereafter, subjects are asked to input their Twitter handle and their name as recorded DBLP Persons². Based on their name, we retrieve a list of a user’s papers and obtain the content of the papers by mapping them to the ACM-Citation-Network V8 dataset, which is described later. The top-5 recommendations are computed for each strategy. Thus, subjects have to evaluate $5 \cdot 12 = 60$ items as “interesting” or “not interesting” based on relevance to their research interests. Items are displayed with the bibliographic information including authors, title, year, and venue. Furthermore, subjects can directly access and read the research paper by clicking on the link of an item. In order to avoid a bias, the sequence of the twelve strategies is randomized for each subject. The list of top-5 items of each strategy is randomized as well, to avoid the well-known ranking bias [4, 5]. After evaluating all strategies, subjects are asked to fill out a form about demographic information

² <https://dblp.uni-trier.de/pers/>.

such as age and profession. Finally subjects can state qualitative feedback about the experiment.

Subjects. $n = 22$ subjects were recruited through Twitter and mailing lists. Subjects are on average 36.45 years old (SD: 5.55). Regarding the academic degree, two subjects have a Master, thirteen a Ph.D., and seven are lecturer/professors. Subjects published on average 1256.97 tweets (SD: 1155.8). Regarding research papers for user profiling, on average a subject has 11.41 own papers (SD: 13.53).

Datasets. As a corpus of research papers, we use the ACM-Citation-Network V8 dataset provided by the ArnetMiner [23]. From the dataset, we use 1,669,237 of 2,381,688 research papers that have title, author, year of publications, venue, and abstract. We use title and abstract to generate paper profiles. As a knowledge base for CF-IDF and HCF-IDF, we use the ACM Computing Classification System (CCS) ³. It focuses on computer science and is organized in a hierarchical structure. It consists of 2,299 concepts and 9,054 labels.

Metric. To evaluate the serendipity of recommendations, we use the Serendipity Score (SRDP) [6]. It takes into account both of unexpectedness and usefulness of candidate items, which is defined as: $SRDP = \sum_{d \in UE} \frac{rate(d)}{|UE|}$. UE denotes a set of unexpected items that are recommended to a user. An item is considered as unexpected, if it is not included in a recommendation list computed by the primitive strategy. We use the strategy Own Papers \times TF-IDF \times Cosine Similarity as a primitive strategy, since it is a combination of baselines. The function $rate(d)$ returns an evaluation rate of an item d given by a subject. If a subject evaluates an item as “interesting”, it returns 1. Otherwise, it returns 0.

4 Result and Discussion

Table 2 shows the results of twelve strategies in terms of SRDP. We use the strategy Own Papers \times TF-IDF \times Cosine Similarity as a primitive strategy. Thus, mean and standard deviation of SRDP are 0.00 for that strategy as shown at the bottom in Table 2. An ANOVA is conducted to verify significant differences between strategies. The significance level for statistical tests is set to $\alpha = .05$. The Muchly’s test ($\chi^2(54) = 80.912$, $p = .01$) detects a violation of sphericity. Thus, a Greenhouse-Geisser correction with $\epsilon = 0.58$ is applied. The ANOVA reveals significant differences between the strategies ($F(5.85, 122.75) = 3.51$, $p = .00$). Furthermore, a Shaffer’s MSRB procedure [20] is conducted to find the pairwise differences. We observe several differences, but all of them are differences between the primitive strategy and one of the other strategies.

In order to analyze the impact of each experimental factor, a three-way repeated measures ANOVA is conducted. The Mendoza Test identifies violation of sphericity [15] for the global ($\chi^2(65) = 101.83$, $p = .0039$) and the factor

³ <https://www.acm.org/publications/class-2012>.

Table 2. SRDP and the number of unexpected items of the twelve strategies. The values are ordered by SRDP. M and SD denote mean and standard deviation.

	Strategy			SRDP	UE
	Text mining method	Profiling source	Ranking method	M (SD)	M (SD)
1	TF-IDF	Own papers	IA-Select	.45 (.38)	2.95 (1.05)
2	CF-IDF	Twitter	CosSim	.39 (.31)	4.91 (0.29)
3	TF-IDF	Twitter	IA-Select	.36 (.29)	4.91 (0.43)
4	CF-IDF	Twitter	IA-Select	.31 (.22)	4.95 (0.21)
5	CF-IDF	Own papers	CosSim	.26 (.28)	4.91 (0.29)
6	CF-IDF	Own papers	IA-Select	.25 (.28)	4.91 (0.29)
7	HCF-IDF	Own papers	IA-Select	.24 (.22)	4.95 (0.21)
8	HCF-IDF	Twitter	CosSim	.22 (.28)	5.00 (0.00)
9	TF-IDF	Twitter	CosSim	.20 (.24)	4.95 (0.21)
10	HCF-IDF	Twitter	IA-Select	.18 (.21)	5.00 (0.00)
11	HCF-IDF	Own papers	CosSim	.16 (.18)	5.00 (0.00)
12	TF-IDF	Own papers	CosSim	.00 (.00)	0.00 (0.00)

Text Mining Method × *Ranking Method* ($\chi^2(2) = 12.01, p = .0025$). Thus, the three-way repeated-measure is applied with a Greenhouse-Geisser correction of $\epsilon = .54$ for the global and $\epsilon = .69$ for the factor *Text Mining Method* × *Ranking Method*. Table 3 shows the result with F-Ratio, effect size η^2 , and *p*-value. Regarding the single factors, *Ranking Method* has the largest impact on SRDP, as an effect size η^2 presents. A post-hoc analysis reveals that the strategies using IA-Select make higher SRDP than those with cosine similarity. In addition, we observe a significant difference in the factors *User Profile Source* × *Ranking Method* and *Text Mining Method* × *Ranking Method*. In both factors, post-hoc analyses reveal significant differences, when a baseline is used in either of the two factors. When a baseline is used in one factor, |UE| becomes small unless a method other than baseline is used in the other factor.

Table 3. Three-way repeated-measure ANOVA for SRDP with Greenhouse-Geisser correction with F-ratio, effect size η^2 , and *p*-value.

Factor	F	η^2	p
<i>User Profile Source</i>	2.21	.11	.15
<i>Text Mining Method</i>	3.02	.14	.06
<i>Ranking Method</i>	14.06	.67	.00
<i>User Profile Source</i> × <i>Text Mining Method</i>	0.98	.05	.38
<i>User Profile Source</i> × <i>Ranking Method</i>	18.20	.87	.00
<i>Text Mining Method</i> × <i>Ranking Method</i>	17.80	.85	.00
<i>User Profile Source</i> × <i>Text Mining Method</i> × <i>Ranking Method</i>	2.39	.11	.11

The results of our experiment reveal that tweets do not improve the serendipity of recommendations. As shown at the rightmost column in Table 2, tweets deliver unexpected recommendations to users. However, only a small fraction of these serendipitous recommendations were interesting to the users. The results show further that the IA-Select algorithm delivers serendipitous research paper recommendations. Thus, we extend on the related works of using IA-Select to improve serendipity to the context of a research paper recommender.

5 Conclusion

In this paper, we investigate whether tweets and IA-Select deliver serendipitous recommendations. We conduct an experiment following the three factors. The result of the experiment reveals that tweets do not improve the serendipity of recommendations, but IA-Select does. This insight contributes to future recommender systems in such a sense that a provider can make informed design choices for the systems and services developed.

References

1. Abel, F., Herder, E., Krause, D.: Extraction of professional interests from social web profiles. In: Proceedings of ACM Conference on User Modeling, Adaptation and Personalization (2011)
2. Agrawal, R., Gollapudi, S., Halverson, A., Ieong, S.: Diversifying search results. In: Proceedings of ACM International Conference on Web Search and Data Mining (WSDM), pp. 5–14. ACM (2009)
3. Beel, J., Gipp, B., Langer, S., Breiting, C.: Research-paper recommender systems: a literature survey. *Int. J. Digit. Libr.* **17**(4), 305–338 (2016)
4. Bostandjiev, S., O’Donovan, J., Höllerer, T.: Taste-weights: a visual interactive hybrid recommender system. In: Proceedings of ACM Conference on Recommender Systems (RecSys). ACM (2012)
5. Chen, J., Nairn, R., Nelson, L., Bernstein, M., Chi, E.: Short and tweet: experiments on recommending content from information streams. In: Proceedings of SIGCHI Conference on Human Factors in Computing Systems (CHI). ACM (2010)
6. Ge, M., Delgado-Battenfeld, C., Jannach, D.: Beyond accuracy: evaluating recommender systems by coverage and serendipity. In: Proceedings of ACM Conference on Recommender Systems (RecSys), pp. 257–260. ACM (2010)
7. Goossen, F., IJntema, W., Frasincar, F., Hogenboom, F., Kaymak, U.: News personalization using the CF-IDF semantic recommender. In: Proceedings of International Conference on Web Intelligence, Mining and Semantics (WIMS). ACM (2011)
8. Große-Bölting, G., Nishioka, C., Scherp, A.: A comparison of different strategies for automated semantic document annotation. In: Proceedings of the 8th International Conference on Knowledge Capture (K-CAP), pp. 8:1–8:8. ACM (2015)
9. Große-Bölting, G., Nishioka, C., Scherp, A.: Generic process for extracting user profiles from social media using hierarchical knowledge bases. In: Proceedings of International Conference on Semantic Computing (IEEE ICSC), pp. 197–200. IEEE (2015)

10. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst. (TOIS)* **22**(1), 5–53 (2004)
11. Kapanipathi, P., Jain, P., Venkataramani, C., Sheth, A.: User interests identification on Twitter using a hierarchical knowledge base. In: Presutti, V., d’Amato, C., Gandon, F., d’Aquin, M., Staab, S., Tordai, A. (eds.) *ESWC 2014. LNCS*, vol. 8465, pp. 99–113. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-07443-6_8
12. Letierce, J., Passant, A., Breslin, J.G., Decker, S.: Understanding how Twitter is used to spread scientific messages. In: *Proceedings of Web Science Conference*. Web Science Trust (2010)
13. 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, Boston (2011). https://doi.org/10.1007/978-0-387-85820-3_3
14. McNee, S.M., Riedl, J., Konstan, J.A.: Being accurate is not enough: how accuracy metrics have hurt recommender systems. In: *CHI Extended Abstracts on Human Factors in Computing Systems*, pp. 1097–1101. ACM (2006)
15. Mendoza, J.L.: A significance test for multisample sphericity. *Psychometrika* **45**(4), 495–498 (1980)
16. Nascimento, C., Laender, A.H., da Silva, A.S., Gonçalves, M.A.: A source independent framework for research paper recommendation. In: *Proceedings of International ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, pp. 297–306. ACM (2011)
17. Nishioka, C., Große-Börling, G., Scherp, A.: Influence of time on user profiling and recommending researchers in social media. In: *Proceedings of International Conference on Knowledge Technologies and Data-Driven Business (i-KNOW)*. ACM (2015)
18. Nishioka, C., Scherp, A.: Profiling vs. time vs. content: what does matter for top-k publication recommendation based on Twitter profiles? In: *Proceedings of IEEE/ACM Joint Conference on Digital Libraries (JCDL)*, pp. 171–180. ACM (2016)
19. Salton, G., Buckley, C.: Term-weighting approaches in automatic text retrieval. *Inf. Process. Manag.* **24**(5), 513–523 (1988)
20. Shaffer, J.P.: Modified sequentially rejective multiple test procedures. *J. Am. Stat. Assoc.* **81**(395), 826–831 (1986)
21. Sugiyama, K., Kan, M.Y.: Scholarly paper recommendation via user’s recent research interests. In: *Proceedings of ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL)*. ACM (2010)
22. Sugiyama, K., Kan, M.Y.: Towards higher relevance and serendipity in scholarly paper recommendation. In: *ACM SIGWEB Newsletter (Winter)*, p. 4 (2015)
23. Tang, J., Zhang, J., Yao, L., Li, J., Zhang, L., Su, Z.: ArnetMiner: extraction and mining of academic social networks. In: *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pp. 990–998. ACM (2008)
24. Vargas, S., Castells, P., Vallet, D.: Explicit relevance models in intent-oriented information retrieval diversification. In: *Proceedings of International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, pp. 75–84. ACM (2012)



The Rise and Rise of Interdisciplinary Research: Understanding the Interaction Dynamics of Three Major Fields – Physics, Mathematics and Computer Science

Rima Hazra¹(✉), Mayank Singh², Pawan Goyal¹, Bibhas Adhikari¹,
and Animesh Mukherjee¹

¹ Indian Institute of Technology, Kharagpur, India
to_rima@iitkgp.ac.in, {pawang, animeshm}@cse.iitkgp.ac.in,
bibhas@maths.iitkgp.ac.in

² Indian Institute of Technology, Gandhinagar, India
singh.mayank@iitgn.ac.in

Abstract. The distinction between sciences is becoming increasingly more artificial – an approach from one area can be easily applied to the other. More exciting research nowadays is happening perhaps at the interfaces of disciplines like Physics, Mathematics and Computer Science. How do these interfaces emerge and interact? For instance, is there a specific pattern in which these fields cite each other? In this article, we investigate a collection of more than 1.2 million papers from three different scientific disciplines – Physics, Mathematics, and Computer Science. We show how over a timescale the citation patterns from the core science fields (Physics, Mathematics) to the applied and fast-growing field of Computer Science have drastically increased. Further, we observe how certain subfields in these disciplines are shrinking while others are becoming tremendously popular. For instance, an intriguing observation is that citations from Mathematics to the subfield of machine learning in Computer Science in recent times are exponentially increasing.

Keywords: Interdisciplinarity · Computer science · Mathematics · Temporal bucket signatures

1 Introduction

Science is built upon previous knowledge, which spans over ideas and concepts drawn from multiple disciplines. The availability of scientific literature from different disciplines of research is vastly expanding due to the advancement in the Internet infrastructure. Several recent studies show how these disciplines interact with each other. Organization for Economic Cooperation and Development (OECD, 1998) [1, 2] defines three classes of research based on several levels of interactions among disciplines: (i) multidisciplinary, (ii) interdisciplinary,

Table 1. Dataset description.

Sl. No	Fields	# papers	# subfields
1	Computer science	1,41,662	40
2	Mathematics	2,84,540	33
3	Physics	8,04,360	50

Table 2. Filtered datasets.

Time period	# papers per field		
	MA	PHY	CS
B1 (1995–1999)	726	7553	86
B2 (2000–2004)	2452	13912	104
B3 (2005–2009)	5913	19748	648
B4 (2010–2014)	5535	12609	4006
B5 (2015–2017)	10449	29896	3186

and (iii) transdisciplinary research. In multidisciplinary research paradigm, researchers from different fields collaborate together, but each of them confines their research to their disciplinary boundaries and exploit their own domain knowledge to address the problem. In contrast, in interdisciplinary research, researchers integrate concepts of different fields to solve their domain problems. The transdisciplinary research adds another dimension in which researchers create an intellectual group beyond their field.

The current work focuses on interdisciplinary research. The main objective of interdisciplinary research is to improve fundamental approaches or to solve problems whose solutions are not in the scope of a single field of research practice [3]. Interdisciplinary research is a common practice in science since early decades. Recent studies show how science is becoming highly interdisciplinary in the last few decades [1]. This work presents a thorough analysis of citation interactions to demonstrate interdisciplinarity among scientific disciplines. Citation interactions are represented by bibliographic relations such as “*who cites whom*”, “*when one cites other*”, etc. We focus on “*who cites whom*” relationships to quantify interdisciplinarity. Here, “*who*” represents citing field and “*whom*” refers to cited field. As a case study, for the first time, we conduct empirical experiments on three research fields Computer Science (*CS*), Physics (*PHY*), and Mathematics (*MA*) as these fields are closely interacting among themselves and exchanging their domain knowledge to address critical problems within their domains. Overall, the main objectives of this work are twofold:

1. Investigating patterns of citations across research fields.
2. Thorough analysis of citation interactions leading to the interdisciplinarity of research fields.

2 Datasets

We crawled *arXiv*¹, one of the well-known pre-print repositories and collected all the research articles—metadata as well as L^AT_EX source code—submitted between 1990–2017. It contains more than 1.2 million articles published in nine major fields. Each research field is further sub-divided into several subfields.

¹ www.arxiv.org.

For our experiments, we select the three major fields – Computer Science (*CS*), Mathematics (*MA*), and Physics (*PHY*). The total number of papers in each field and the respective number of subfields is noted in Table 1. Next, the citation network among papers is constructed by parsing the references present in “.bbl” files. For each candidate paper, we only extract those referenced papers that are available in *arXiv* by matching title string. In our experiments, we consider only those papers which have at least five extracted references.

3 Empirical Analysis

We conduct an in-depth temporal analysis of citation interactions among the three disciplines. We group citation interactions into multiple buckets based on the publication year of the citing paper. We report results for bucket size of five years². We, divide the entire dataset (see Table 2) into five buckets (i) **B1** (1995–1999), (ii) **B2** (2000–2004), (iii) **B3** (2005–2009), (iv) **B4** (2010–2014), and (v) **B5** (2015–2017). The total number of papers belonging to each time period, and each major field are noted in Table 2. Citations gained from articles published within the same field are termed as *self-field citations*. Incoming citations from other fields are termed as *non self-field citations*. We observe, empirically, that the proportion of self-field citations is significantly higher than non self-field citations. We, therefore, study citation interactions at two levels – (i) field and (ii) subfield level. Subfield level citation interactions present a more in-depth understanding of the interdisciplinary nature across these fields.

Sankey diagram is a graphical representation of flow from left to right in which width of the link is proportional to the amount of flow. In Fig. 1, we have shown the citation flow among *CS*, *MA* and *PHY*. The leftmost nodes are represented as the source and the rightmost nodes are represented as the target. In our case, leftmost nodes are denoted as “citer” field and rightmost nodes are denoted as “cited” field. The link between source to target denotes the citation flow from the citer field to the cited field.

Observations: Figure 1 shows citation flow among three fields. The temporal study uncovers several interesting observations. During initial time-periods, *CS* was poorly cited by the other two fields (with no citation from *MA* and *PHY* in **B1**). All of the non self-field citations from *PHY* went to *MA* and vice-versa. However, in later time-periods, *CS* started receiving attention from both *PHY* and *MA*. It is also clearly evident that in the initial time-periods, *PHY* was more cited by *CS* than *MA*, however, the trends are reversed in the later time-periods. Note that, here, we do not consider the flow of self-field citations. We observe similar self-field citation flow trends for each field. We next discuss bucket-wise observations:

B1 (1995–1999): During this time-period, *MA* entirely cites *PHY* and vice-versa. We do not observe in-/outflow of citations to/from *CS*.

² Our results hold for other bucket sizes also. As, papers between 1990–1995 are very less in number, we start buckets from the year 1995.

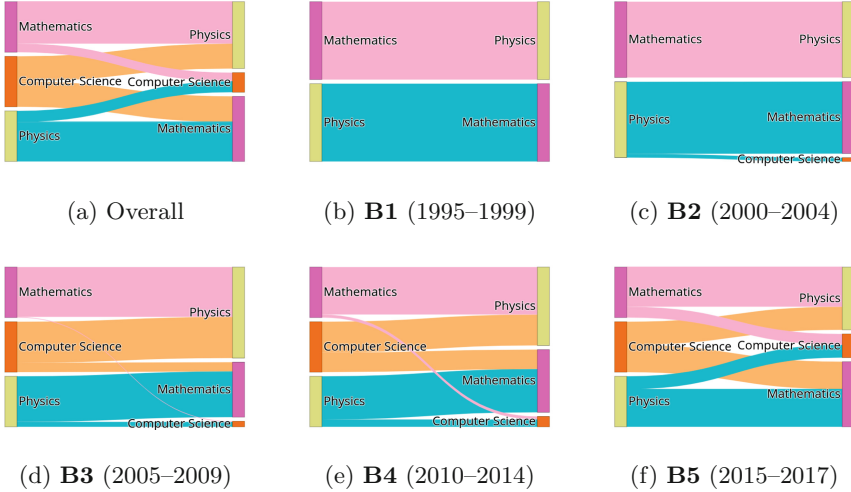


Fig. 1. Citation flow among three fields of science, Computer Science, Physics, and Mathematics over the (a) entire time-period, and (b-f) five temporal buckets.

B2 (2000–2004): During this time-period, a marginal number of citations flow to *CS* from *PHY*. *MA*, still, only cites *PHY*. Here, we do not observe outflow of citations from *CS*.

B3 (2005–2009): During this time-period, *CS* has started citing both the other fields; *CS* seems to have been drawing more ideas from *PHY* than *MA* in this period. In contrast to the previous bucket, the number of citations from *PHY* to *CS* has increased.

B4 (2010–2014): Interestingly, in this time span, we witness a complete shift in the citation patterns received by the *CS* papers. In particular, *CS* seems to have started receiving citations from *MA*.

B5 (2015–2017): *PHY* and *MA* both seem to be equally citing *CS* papers in this span. We posit that this interesting trend could be mostly attributed to the newly emerging topics like *Deep Learning*, *Machine Learning*, *Statistical Natural Language Processing*, etc. These topics significantly borrow many ideas from Mathematics.

Similar to Sankey diagrams, temporal bucket signatures (TBS) [4] present a novel visualization technique to understand the temporal citation interactions. TBS refers to a stacked histogram of the relative age of target papers cited in a source paper. Suppose we collect citation links into fixed-size buckets of temporal width T (e.g., $T = 5$ years). We partition the entire article set into these fixed buckets based on publication year. For each bucket, we compute the fraction of papers of other buckets that are cited by the papers of the current bucket. As self-field citations are significantly larger in number, we create TBS for self-field citations and non self-field citations separately. In the case of self-field citation TBS, we analyze how papers of the same field in older buckets receive citations

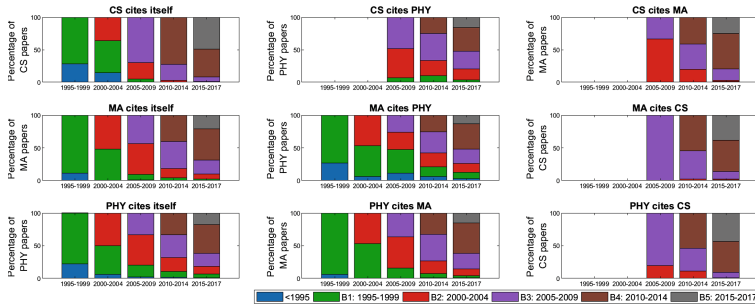


Fig. 2. Fraction of citation going from one field to another field over the years.

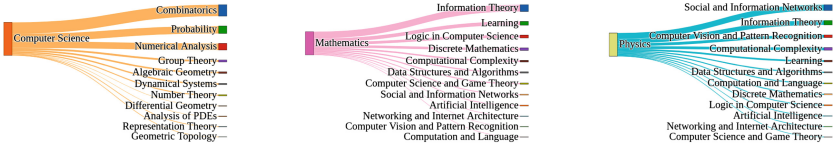
from the current bucket papers. In the case of non self-field citation TBS, we observe how papers belonging to other fields in older buckets receive citations from current bucket papers.

Observations: Figure 2 shows proportional citation flow from each field to other fields in different temporal buckets. *CS*, in contrast to *MA* and *PHY* cites current bucket papers more than the older bucket papers of its own field (evident from the higher proportion of the top segment in each temporal bucket). This observation reconfirms the common intuition that “*CS* is fast growing field”. In contrast, *CS* tends to cite older papers from the other two fields – *MA* and *PHY* (denoted by a lower proportion of top-most segment in each temporal bucket). *MA* and *PHY* predominantly cite older papers of each other and recent papers from *CS*.

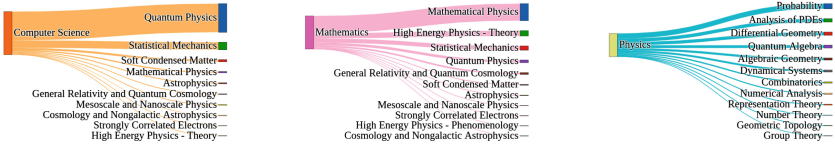
Next, we identify top 12 subfields of each individual field that received the highest number of citations between 1995–2017 (the supplementary material³ notes the different subfields). The popularity of subfields seems to be inconsistent at different time-periods. It is observed that several subfields have become obsolete over the time with a drastic decrease in their incoming citations. For example, *Computation and Language*, a subfield of *CS*, was among top-three most cited subfields during earlier time-periods (B1–B3), but its popularity drastically reduced during the later time-periods (B4–B5). In *CS*, we found no subfield that always exists in the most cited (top three) list. In case of *PHY*, two subfields are always in the most cited list: (i) *High Energy Physics - Theory* and (ii) *High Energy Physics - Phenomenology*. *MA* witnesses new subfields such as *Group Theory* and *Representation Theory* gaining high popularity whereas old subfields like *Logic* and *Classical Analysis and ODEs* depleting over the time.

Citation Flow Analysis: Next, we perform an analysis of citation flow from fields to subfields. We, again, leverage Sankey diagrams for graphic illustration of the flows. We conduct empirical analysis for the entire time-period along with different temporal buckets. Figure 3 shows citation flow from each field to other field’s subfields over the entire time period. *CS* mostly cites subfields such

³ <https://tinyurl.com/yxrhvufy>.



(a) CS citing MA subfields (b) MA citing CS subfields (c) PHY citing CS subfields



(d) CS citing PHY subfields(e) MA citing PHY subfields(f) PHY citing MA subfields

Fig. 3. Citation outflow from fields to other field’s subfields over the year range 1995–2017.

as *Combinatorics*, *Probability*, and *Numerical Analysis* from *MA* and *Quantum Physics* and *Statistical Mechanics* from *PHY*. Citation inflow from *CS* to *Quantum Physics* is significantly larger than to any other subfield of *PHY*. Similarly, *MA* mostly cites *CS* subfields such as *Information theory* and *Learning* and *PHY* subfields such as *Mathematical Physics* and *High Energy Physics - Theory*. In particular, *MA* cites *Mathematical Physics* in a significantly high proportion ($\sim 51.48\%$). *PHY* mainly cites *CS* subfields like *Social and Information Networks*, *Information Theory*, *Computer Vision & Pattern Recognition*, *Computational Complexity* and *Learning* and *MA* subfields such as *Probability* and *Analysis of PDE*. The subfield *Information Theory* in *CS* remains popular for both *PHY* and *MA*. Similarly, *MA*’s subfield *Probability* remains popular for both *CS* and *PHY*. The most interesting outcome here is the growing interest of *PHY* and *MA* in the *CS* subfield *Learning*. This possibly indicates that in recent times mathematicians and physicists have started taking interest in formulating the mathematical foundations of various machine learning techniques (e.g., theoretical foundations of the deep learning machinery)⁴.

4 Conclusion and Future Work

We study a large collection of research articles from three different scientific disciplines – *PH*, *MA*, and *CS* to understand how citation patterns from the core science fields to applied field have drastically changed. Besides, our work raises some fundamental questions such as which factors of a subfield are responsible for gaining citations from other disciplines? Is it related to the development of that subfield by borrowing ideas from other disciplines or due to the appearance

⁴ Please see supplementary material: <https://tinyurl.com/yxrhvufy>, for more results.

of a new idea in that subfield that attracts attention from other disciplines? This can be studied by identifying seminal contributions in that subfield and its relation to the significant ideas of other subfields which are cited by the original subfield.

References

1. Morillo, F., Bordons, M., Gómez, I.: Interdisciplinarity in science: a tentative typology of disciplines and research areas. *J. Am. Soc. Inf. Sci. Technol.* **54**(13), 1237–1249 (2003)
2. Organisation for economic cooperation and development (OECD) (1998)
3. Porter, A.L., Rafols, I.: Is science becoming more interdisciplinary? Measuring and mapping six research fields over time. *Scientometrics* **81**(3), 719 (2009)
4. Singh, M., Sarkar, R., Goyal, P., Mukherjee, A., Chakrabarti, S.: Relay-linking models for prominence and obsolescence in evolving networks. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2017, pp. 1077–1086. ACM (2017)



The Evolving Ecosystem of Predatory Journals: A Case Study in Indian Perspective

Naman Jain^(✉) and Mayank Singh

Department of Computer Science and Engineering,
Indian Institute of Technology Gandhinagar, Gandhinagar, Gujarat, India
{naman.j,singh.mayank}@iitgn.ac.in

Abstract. Digital advancement in scholarly repositories has led to the emergence of a large number of open access predatory publishers that charge high article processing fees from authors but fail to provide necessary editorial and publishing services. Identifying and blacklisting such publishers has remained a research challenge due to the highly volatile scholarly publishing ecosystem. This paper presents a data-driven approach to study how potential predatory publishers are evolving and bypassing several regularity constraints. We empirically show the close resemblance of predatory publishers against reputed publishing groups. In addition to verifying standard constraints, we also propose distinctive signals gathered from network-centric properties to understand this evolving ecosystem better. To facilitate reproducible research, we shall make all the codes and the processed dataset available in the public domain.

Keywords: Predatory journals · Publication ethics · Open access · Digital library

1 Introduction

Scholarly journals play an essential role in the growth of science and technology. They provide an information sharing platform for researchers to publish and access scientific literature. However, high monthly access costs, pay-per-view models, complicated and lengthy publication processes restrict researchers to leverage such knowledge. Open access journals (OAJ) emerged as a solution to abolish high access charges, with provision for unrestricted access to the latest findings. OAJs operate by charging article processing fees from the authors with a promising commitment of rigorous peer-review to maintain the quality and academic standard. The evolution of OAJs has been under investigation for at least a decade. Many fascinating studies have shown evidence of malpractices in large-scale. Majority of the research is involved in proposing a set of criteria to identify such malpractices and blacklist those violating these criteria. We review

a series of such criteria and show that malpractices have gradually evolved and are difficult to detect through standard small-scale manual studies. We conduct empirical experiments to present facts that clearly demarcate reputed journals from dubious journals.

Publication Ethics and Malpractices. Identification of malpractices in publishing is a challenging yet an important problem. Several small-scale manual studies have been performed to identify these publishers and blacklist them. Xia *et al.* [36] present evidence of malpractices in OAJ's fee-based model. They observed that majority of OAJs use '*enticing websites*' to attract researchers with a promise of pretentious peer-review and faster publishing process. Recently, Gopalkrishnan *et al.* [14] verified above claims by surveying 2000 authors that published in dubious journals. Due to a high disparity in the fee structure and other popular competitors, several OAJs adopted malpractices such as false editorial board, poor peer-review, and incorrect indexing information [2]. Some recent and more advanced malpractices include the formation of '*citation cartels*' [12] among journals to elevate the impact factor. Sorokowski *et al.* [32] demonstrated malpractices in recruiting editors by creating a fake profile of a researcher and applying for an editorial position at 360 journals. To their surprise, 48 bogus journals accepted the application without even verifying the information. Phil Davis [8] submitted articles—generated by a popular scholarly tool *SCIgen*—in several lousy quality journals which claim to perform peer-review for nominal processing charges. *SCIgen* is a program that generates random Computer Science research papers, including graphs, figures, and citations. Following a similar methodology, Bohannon [4] submitted grammatically incorrect papers to 304 OAJs. Out of 304 journals, 157 accepted the paper, 98 rejected, and the rest of them consider it for review or failed to respond. Most of these journals accepted articles based on formatting, layout, and language, thus, highlighting the malpractices in the peer-review process.

Predatory Publishing and Growth. Beall [2] coined the term '*predatory journals*' for those journals that charge high article processing fees from authors but fail to provide necessary editorial and publishing services. He proposed a list of subjective criteria for predatory classification based on a manual study performed on a limited set of journals. In addition, Beall curated a list of questionable journals and publishers (popularly known as "*Beall's list*") which includes hijacked journals, fake-metric companies, and predatory journals or publishers. Shen *et al.* [31] conducted a manual study of 613 journals from Beall's list and showed high growth in popularity within a span of five years. The publication count increased from 50,000 articles (in 2010) to 420,000 articles (in 2014).

Limitations in Current Research. Majority of the research in identification and curation of dubious journals revolves around Beall's list. However, multiple works have shown limitations in this approach. Sorokowski *et al.* [32] criticized Beall for relying on editorial appeal rather than analyzing the published content. Laine *et al.* [17] claimed that Beall's criteria list is not ranked in a preference order resulting in a geographical bias towards reputed journals being categorized

as predatory from developing countries. The list of journals used by Xia *et al.* [36] and Gopalkrishnan *et al.* [14] were sampled from the geographically biased Beall’s list. Sorokowski *et al.* [32] criticized Bohannon [4] for targeting only specific journals and for not comparing both reputed and predatory journals. Moreover, Beall’s list is currently discontinued due to unknown reasons [24, 33]. In addition, majority of the Beall’s criteria have been bypassed by the evolving predatory ecosystem which we have discussed in Sect. 4 through empirical experiments.

Landscape of Predatory Journals in India. David Moher [30] claimed that 27% of world’s predatory journal publishers are located in India and 35% corresponding authors are Indians. A similar study by Demir [9] on 832 predatory journals yields 62% journals being located in India along with authorship and editorship contribution of 10.4% and 57% respectively. In July 2018, leading Indian newspaper, The Indian Express, published an investigative story [37] claiming that Hyderabad is the Indian hub of predatory journals, operating more than 300 companies and publishing close to 1500 journals. A similar study by Patwardhan *et al.* [21] claimed that $\sim 88\%$ of the journals present in the University Grants Commission of India (UGC)’s ‘*approved list of journals*’ could be of low quality. Following widespread criticism, UGC has removed 4,305 journals that are either predatory or low quality.

Our Contribution. In our current study, we present an empirical study of one of the most popular Indian publishing group **OMICS** (OPG) that is largely considered as ‘predatory’ by several scholarly organizations and investigative journalists including The Guardian [15], Federal Trade Commission (USA) [11] and The Indian Express [37]. We present several anecdotal evidences to show how OPG dubiously bypasses the majority of Beall’s criteria and cautiously adapting itself to resemble highly reputed publishing groups. As an interesting example, a comparison study with BioMedical Central (BMC) reveals that OPG shares several characteristics with reputed publisher. In addition, BMC follows multiple predatory criteria proposed by Beall. Through our findings, we propose several empirical verification strategies to find statistical evidence in decision making. We claim that similar strategies can be applied to any scholarly publishers.

2 Datasets

Understanding the predatory nature of open-access scientific journals requires a rich bibliographic dataset. We, therefore, downloaded the entire OMICS publishing group (OPG) dataset [20]. It spans multiple subjects from broad research domains such as medical sciences, economics, and engineering and indexes 782 journals. OPG operates in 10 different publishing classes such as iMedPub LTD, Pulsus Group, and Allied Academics [35]. Table 1 presents journal and article distribution in OPG classes. Majority of the articles belong to OMICS class followed by RROIJ, Imedpub, and TSI. Table 2 presents general statistics of OPG, OMICS class (the parent group) and well-known BioMed Central publishing group. BMC [3], established in 2000, is a prominent United Kingdom-based publishing group. It is an open access group owned by Springer Nature. Currently,

Table 1. Journal and article distribution in OPG classes (second and third column). Fourth column represents per class count of OPG journals in SJR. Only 21 (~2.6%) journals in OPG are indexed in SJR

OPG classes	Journals	Articles	SJR
OMICS	498	70,247	15
Imedpub LTD	144	13,050	1
SciTechnol	55	4,734	0
RROIJ	36	18,703	0
Trade Science Inc (TSI)	22	12,823	1
Allied Academics	9	145	0
Open Access Journals	5	2,310	1
Scholars Research Library	4	399	0
Pulsus	3	91	3
Andrew John Publishing	2	12	0
Total	782	122,514	21

Table 2. Salient statistics about the full OPG, OMICS class in particular, and the BMC Group.

	OPG	
Number of Journals	782	
Number of papers	122,514	
Classes	10	
Research Fields	29	
	OMICS	BMC
Number of Journals	498	334
Number of papers	70,247	369,102
Research Fields	29	18
Total editors	16,859	21,859
Total unique editors	14,665	20,153
Total authors	223,626	2,223,945
Total unique authors	203,143	2,034,394

it publishes more than 300 scientific journals. We claim that similar study can be conducted for other reputed open access publishers such as PLOS, Nature, and Science.

3 OMICS Publishing Group Vs. BMC

In this section, we analyze the entire publishing eco-system of OPG and compare the interesting findings with BMC publishing group. We leverage several well-known online databases to analyze and authenticate the claims and beliefs.

3.1 The Meta Information

Indexing in Digital Directories and Scholarly Associations. Scholarly digital directories are created with an aim to increase the visibility and ease of use of open access scientific and scholarly journals. Some prominent directories include *SJR* (described in the previous section) and *Directory of Open Access Journals (DOAJ)*. Similarly, several global scholarly associations like *Open Access Scholarly Publishers Association (OASPA)*, *Committee on Publication Ethics (COPE)* and *International Association of Scientific and Technical & Medical Publishers (STM)*, maintain standards for strict regulations and ethical practices in open access publishing. Only 21 out of 782 (~2.6%) journals in OPG are indexed in SJR (see Table 1). Four classes (SciTechnol, RROIJ, Allied Academics and SRL) have no members indexed in SJR. Surprisingly, not a single OPG journal is present in DOAJ. Also, the three associations OASPA, COPE and STM, do not include any OPG journals in their list.

However, BMC shows a high acceptance in above prominent directories. Out of 334 BMC journals, 288 (~89%) journals are indexed in SJR. Similarly, 322

BMC journals are present in DOAJ. Also, the three associations OASPA, COPE and STM, have listed Springer Nature under its members.

Impact Factor. Among 782 OPG journals, only 432 journals have reported their impact factor. Out of 432 journals, 69 journals have cited their source agency for impact factor computation. Interestingly, the journal class *SciTechnol* reports impact factor for 39 out of 55 journals and cites *COSMOS* [6] as a source agency. We found that COSMOS's geographical coordinates points to a kindergarten school in Germany. To our surprise, COSMOS's founder and patron Mr. Jahanas Anderson credentials turned out to be incorrect. A basic Google reverse image search of his profile picture led to a genuine profile of Prof. Stephan Muller [23]. Interestingly, the certificate of acknowledgment provided by COSMOS contains a barcode that redirects to Food Safety and Standards Regulations of India (FSSAI). In addition, we found that *SciTechnol* falsely claimed to be indexed by reputed services such as Google Scholar, DOAJ, etc. We omit related investigative studies on other classes due to space constraint.

In contrast, BMC leverages services of popular Scopus database [27] for impact factor computation. It also provides several addition citation metrics like SCImago Journal Rank [26], CiteScore [10], Altmetric Mentions [1], etc.

Contact Information. Next, we studied the availability and authenticity of the postal address of the editorial offices of OPG journals. Interestingly, we find heavy usage of postal mailbox rental service called Prime Secretarial (PS) [22]. PS provides facilities for renting private street addresses with an extremely high level of confidentiality. Overall, we found 198 journals that used rented addresses, 500 journals do not provide any postal address information. 48 journals do not have even a contact page. They present a web-based form for official communication. Remaining 36 journals, all from RROIJ class, have provided the postal address of some buildings in the city of Hyderabad, India.

Similar to OPG, we study the availability and authenticity of the postal addresses of the editorial offices of BMC. None of the BMC journals provide postal address of operating offices. It either provides the email IDs of their employees or redirects to a submission portal.

Journal Names. Journal names add another dimension to interesting insights. The OPG's journal names are extremely long as opposed to BMC. The average number of words and characters in OPG's journal names are 8.3 and 60.2 respectively. For BMC, the corresponding values are 3.8 and 29.4. We claim that OPG uses longer names to show authenticity and research coverage. 12 OPG journals from *Trade Science Inc.* subclass are titled as "ABC: An Indian Journal" and claim to focus on Indian region where *ABC* includes keywords such as BioChemistry, etc. However, the editorship (10%) representation from India looks abysmal. In contrast to OPG, we find an entirely different trend in BMC. We find several journals focusing on different countries. All of these journals have high contribution from the their native country. For example, Chinese Neurosurgical Journal, Chinese Medicine, Israel Journal of Health Policy Research,

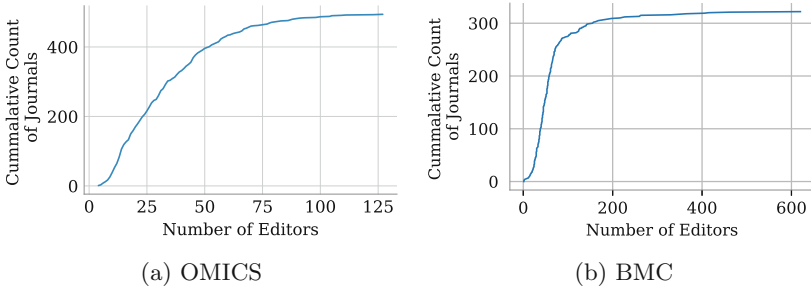


Fig. 1. Editor distribution in (a) OMICS subclass, and (b) BMC. 30.9% OMICS have more than 40 editors. Seven journals have editor count more than 100. (b) 67.06% BMC journals have more than 40 editors. 45 journals have editor count more than 100.

Irish Veteranaray Journal, and Italian Journal of Pediatrics have their respective country contributions as 64%, 78%, 70%, 65% and 52% respectively.

Next, we perform more nuanced set of experiments with only OMICS subclass (*hereafter as OMICS*) due to the difficulty in data curation (non-uniformity in format, unavailability of paper-specific data, etc.) for several OPG classes.

3.2 The Editorial Board

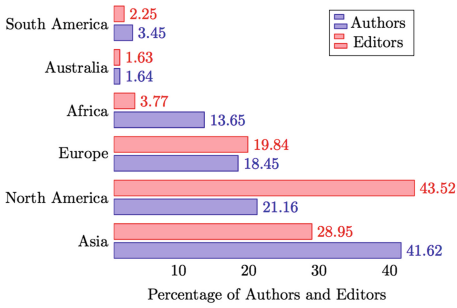
In this section, we study the editorship information by leveraging editors' metadata from the information available at editorial board pages of every journal.

Name Normalization. We find total 16,859 editors in 494 (=34.1 editors per journal) OMICS journals.¹ As opposed to Beall's hypothesis, the majority of the journals have provided complete information of editors—name, designation, affiliation and country—but no contact information such as email, personal homepage, etc. On empirical investigation, we find several instances of editors names with slight variations (acronyms in name, mid name vs. no mid name, etc.) but similar affiliations.² We normalize these variants by leveraging naming conventions and similarity in affiliations. Normalization results in 14,665 unique editors names. In contrast, we find total 21,859 editors in 334 (=65.4 editors per journal) BMC journals. Similar name normalization scheme, as described above, results in 20,153 unique editors names.

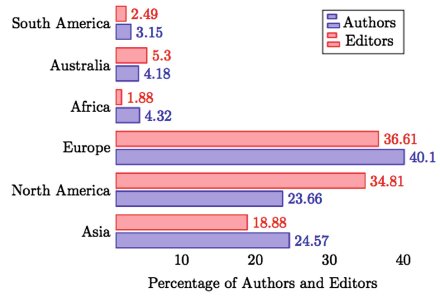
Editor Contribution. Figure 2a shows editor distribution in OMICS journals. More than 9% (=1,409) editors contribute to atleast one journal. 13 editors contribute to more than 10 journals. 30.9% journals have more than 40 editors. Surprisingly, we find seven journals having editor count more than 100. '*Journal of Tissue Science & Engineering Open Access Journal*' has maximum editors (=127). On an average, we find 34.1 editors per journal in OMICS. In contrast to the Beall's criterion, the average editors are significantly closer to well-known

¹ Four journal editor information links are dead.

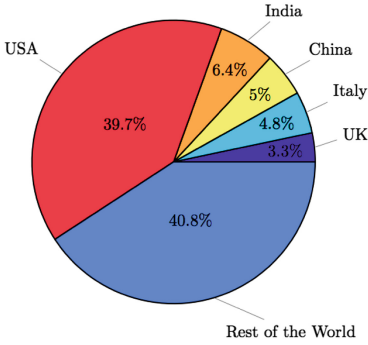
² 'Alexander Birbrair' might be present as 'A Birbrair' and 'Birbrair Alexander'.



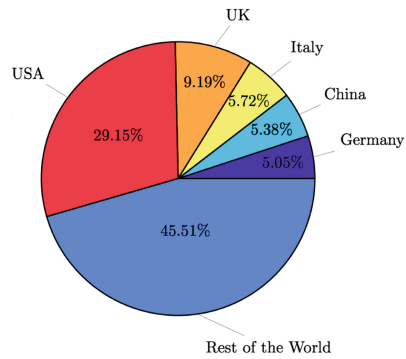
(a) OMICS - Continent



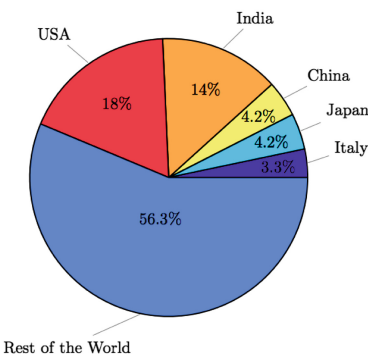
(b) BMC - Continent



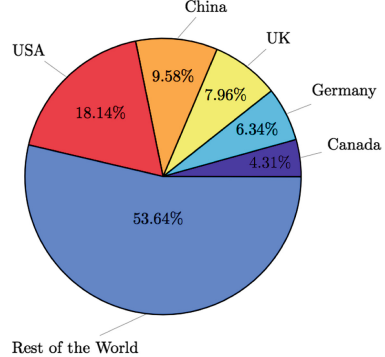
(c) OMICS - Country



(d) BMC - Country



(e) OMICS - Country



(f) BMC - Country

Fig. 2. Continent-wise distribution of editors and authors in (a) OMICS and (b) BMC. Country-wise distribution of editors in (c) OMICS and (d) BMC. Country-wise distribution of authors in (e) OMICS and (f) BMC.

publishers such as BMC (65.4) and Science (97.8). In BMC (see Fig. 2b), more than 6.47% (=1,316) editors contribute to atleast two journals of BMC journals. No editor contribute to more than 10 journals (maximum count is 7). 67.06% journals have more than 40 editors. Surprisingly, we find 45 journals having editor

count more than 100. ‘*BMC Public Health*’ has maximum number of editors (=620). Figure 2b shows editor distribution in BMC journals. It is observed that OMICS operates with high journal count and low editorship count while BMC operates with low journals count and high editorship count.

Geographical Analysis. We leverage affiliations to analyze the geographical distribution of editors. Figure 1a and c shows geographical distribution of OMICS editors. Continent-wise analysis (see Fig. 1a) shows that the majority of editors are affiliated to North American organizations (43.52%) followed by Asia (28.95%). Similar country-wise analysis show that editor affiliates to more than 131 countries across the world (see Fig. 1b), with USA being the major contributor (39.7%) followed by India (6.4%), China (5%), Italy (4.8%) and UK (3.3%). In BMC, continent-wise analysis (see Fig. 1b) shows that the majority of authors affiliates to European organizations (36.61%) followed by North America (34.81%). Editors belong to more than 136 countries across the world (see Fig. 1d), with USA being the major contributor (29.15%) followed by UK (9.19%), Italy (5.72%), China (5.38%) and Germany (5.05%). In contrast to OMICS, Indian editors have marginal contribution (1.51%). It is observed that both of the groups show similar geographical diversity in editor contribution.

Gender Analysis. Next, we leverage popular gender prediction tool *Gender Guesser* [18] to infer gender bias information from the names of the editors in OMICS and BMC. Given a name, the tools outputs a probability of its gender. The probability of an editor being a male came out to be 0.76. We find that 467 journals were heavily male dominated ($P(\text{male}) > 0.5$). 36 journals among 467, do not include any female member in the editorial board. In case of BMC, The probability of an editor being a male came out to be 0.72. We find that 302 journals were heavily male dominated ($P(\text{male}) > 0.5$). 12 journals among 302, do not include any female member in the editorial board.

3.3 The Authorship

In this section, we study the authorship information by leveraging authors’ meta-data from paper headers in the journal’s archive.

Name Normalization. In majority of the papers published in OMICS, we obtained well-structured metadata information—affiliation with the department/lab/center, the organization, city, and country name. As opposed to editors, phone number and email information of the first author is also available. We leverage this metadata information to normalize and index authors (similar to the editor name normalization). In OMICS, normalization results in 203,143 unique authors out of 223,626 authors. Similarly, BMC comprises 2,034,394 unique authors among 2,223,945 authors.

Author Contribution. Among 498 OMICS journals, the author information was available in 481 (~96%) journals. The rest of the journals do not possess an archive portal (12 cases), points to the same archive link (three cases), or contains only volume names but no paper information (two case). Figure 3a shows author

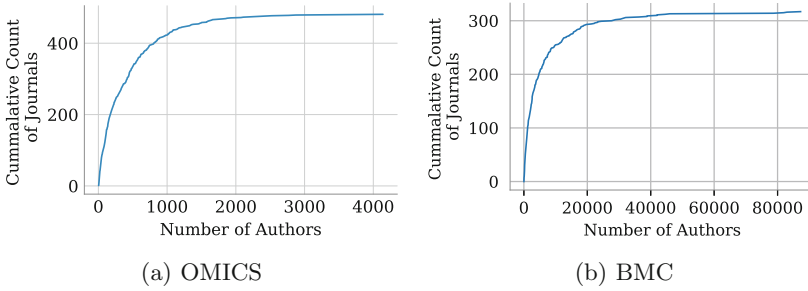


Fig. 3. Authorship distribution in (a) OMICS, and (b) BMC. 6.7% OMICS journals have published in more than one journal. 6.3% authors in BMC journals published in more than one journal.

publishing statistics. 6.7% authors published in more than one journal. We found 20 authors that published in more than 10 journals. 97 authors published in more than five journals. Similar to OMICS, BMC has the author information availability in 318 ($\sim 95\%$) journals. The remaining journals ($=16$) have not started published digitally yet. Figure 3b shows author publishing statistics for BMC. 6.3% authors published in more than one journal. We found 558 authors that published in more than 10 different journals.

Geographic Analysis. Next, we perform a geographical analysis of the authors. In contrast with editors, here we study the geographical distribution of the first author of the paper by leveraging the affiliation information. Contrary to the OMICS CEO’s claims [37], that ‘*Our articles from India are very few, less than 0.01%–99.99% of articles are from outside*’, we find that Indian authorship is $\sim 14\%$ of the global authorship. Also, we find that majority of authors publishing in OMICS class are from Asia(41.62%) (see Fig. 1a and e). In BMC, majority of the authorship comes from Europe(40.1%) followed by Asia(23.66%) and North America(24.57%). It is observed that the combined authorship from three continents, Asia, Africa and South America, having higher percentage of developing countries [34] contributes more towards OMICS than BMC.

Gender Analysis. We perform gender analysis of authors by inferring gender from author names (similar to editor gender identification). In OMICS, the probability of an author being a male came out to be 0.64. We found that 432 OMICS journals were heavily male dominated ($P(\text{male}) > 0.5$) and eight among 432 journals have no contribution from any female author. Surprisingly, in case of BMC, the probability of an author being a male came out to be 0.64. We found that 283 journals were heavily male dominated. However, each journal has non-zero female author count.

3.4 Novel Data-Driven Signals

In contrast to previously proposed subjective list-based evaluation techniques, we can leverage various co-authorship signals and network-centric properties to identify distinctive features between predatory and reputed publishers.

Author-Editor Commonness. OMICS shows significant fraction of editors that published their own work in their journal. 3.2% authors are the editors in the same journal. This phenomenon was observed across 426 Journals out of 498 Journal with an average of 16.8 editors per journal. 4.6% editors are authors in atleast one of the OMICS class journal other than the journal they are editor in. In total, an average of 37.3 editors publish in each journal of OMICS group. In contrast, above trend is negligible in BMC. 0.2% authors are the editors in the same journal.

Editor Network. Next, we perform network analysis over the editor graph. Editor graph $G_E(V, E)$ comprises editors as nodes V , whereas edges E connect editors in the same journal. We rank nodes based on three centrality measures: (i) Degree, (ii) Eigenvector, and (iii) Betweenness. The degree centrality ranks nodes with more connections higher in terms of centrality. A more popular editor has a high degree. However, in real-world scenarios, having more co-editors does not by itself guarantee that some editor is popular. We, therefore, also compute Eigenvector centrality which measures editor connections with other popular (not just any other editor) editors. Betweenness centrality measures how often a node occurs on all shortest paths between two nodes. In the current graph, it measures how frequently editorial decisions/recommendations are passed through that node. Overall, G_E of OMICS contains 14,665 nodes and 401,063 edges. The profiles of top-ranked editors (based on three metrics are verified by looking for their Google Scholar and university/organization webpage. Interestingly, the fourth-ranked editor as per degree centrality named ‘Alireza Heidari’ belongs to ‘California South University’ [5] which is claimed to be fake by Huffington Post [29]. Also, an editor named ‘Sasikanth Adigopula’ in *Journal of Cardiovascular Diseases and Diagnosis* [19] is claimed to be affiliated from Banner University, Colorado. On researching through internet sources, we found that there is no such institution and the name is of a cardiologist from Loma Linda University.

Similar graph construction for BMC editors yields 20,153 nodes and 1,419,776 edges. However, BMC shows no dubious signals in the top-ranked nodes. The editors either possess GS or ORCID user profiles or have authentic university profile pages. Detailed centrality values for Top-5 editors in each measure is shown in Table 3.

4 The Evolving Ecosystem

The previous section compares potential predatory publishing group OMICS with popular publishing group BMC. In this section, we revisit Beall’s list of subjective criteria for predatory classification. Table 4 shows a comparison between OMICS and BMC against the list of 35 Beall’s criteria that can be easily verified

Table 3. Top-5 central nodes in editor graph based on three centrality measure. The green color rows represent profiles verified by Google Scholar (GS), the blue rows represent profiles that are not present in GS but organization website contains their information and the red rows represent profiles whose information cannot be verified.

OMICS			BMC		
Editor Name	Centrality Values	Journal Count	Editor Name	Centrality Values	Journal Count
Degree					
George Perry	0.033	26	Sang Yup Lee	0.033	7
Rabiul Ahasan	0.059	20	Stefano Petti	0.028	2
Kenneth Maiese	0.051	10	Guy Brock	0.025	4
Alireza Heidari	0.045	13	Lalit Dandona	0.024	4
Rajesh Rajan Wakaskar	0.044	11	John Ioannidis	0.023	5
Eigenvector					
George Perry	0.001	26	Guy Brock	0.054	4
Rabiul Ahasan	0.0009	20	Stefano Petti	0.053	2
Sandeep Kumar Kar	0.0006	14	Jonathan Mant	0.053	3
Alexander Birbrair	0.0006	14	Shazia Jamshed	0.051	4
Alireza Heidari	0.0006	13	Florian Fischer	0.049	3
Betweenness					
George Perry	0.100	26	Sang Yup Lee	0.0002	7
Rabiul Ahasan	0.088	20	Fernando Schmitt	0.0002	5
Sandeep Kumar Kar	0.039	14	Jean-Louis Vincent	0.0002	5
Akbar Nikkiah	0.030	11	Suowen Xu	0.0001	5
Alexander Birbrair	0.030	14	Yong Wang	0.0001	5

through internet resources with a minimum requirement for manual processing. Some of the interesting insights are:

- 22 criteria are common between OMICS and BMC.
 - Five criteria are satisfied by both OMICS and BMC.
- 13 criteria are satisfied by OMICS but not by BMC.
- No criteria being satisfied by BMC but not by OMICS.

The predatory ecosystem is cautiously changing. The evolution in OMICS results in its operations similar to BMC publishing group. For example, recently, OMICS started its online submission portal similar to well-known publishers (Nature, BMC, and Science). Earlier, it accepts manuscripts through email. It is becoming extensively hard to distinguish between authentic and predatory journals using a standard list of criteria or rules. We need more data-driven identification methods that can detect false/misleading information in publishing groups. There are some freely available, reliable and important resources such as the Google Scholar API [25] for editor/author profile verification, Scopus API [28] for paper indexing verification, Image Search facility [13] for image authenticity on the website, DOI/ISSN API [7, 16] for electronic verification.

Table 4. Comparison between OMICS and BMC against Beall’s subjective criterion.

Beall’s Criteria	OMICS	BMC
The publisher’s owner is identified as the editor of each and every journal published by the organization.	×	×
No single individual is identified as any specific journal’s editor.	×	×
The journal does not identify a formal editorial / review board.	×	×
No academic information is provided regarding the editor, editorial staff, and/or review board members (e.g., institutional affiliation).	×	×
Evidence exists showing that the editor and/or review board members do not possess academic expertise in the journal’s field.	✓	×
Two or more journals have duplicate editorial boards	×	×
The journals have an insufficient number of board members	×	×
The journals have concocted editorial boards (made up names)	✓	×
The editorial board engages in gender bias (i.e., exclusion of any female members)	✓	✓
Has no policies or practices for digital preservation, meaning that if the journal ceases operations, all of the content disappears from the internet.	✓	×
Begins operations with a large fleet of journals, often using a common template to quickly create each journal’s home page.	✓	✓
Does not allow search engines to crawl the published content, preventing the content from being indexed in academic indexes.	×	×
Copy-proofs (locks) their PDFs, thus making it harder to check for plagiarism.	×	×
On its website, the publisher falsely claims one or more of its journals have actual impact factors, or advertises impact factors assigned by fake “impact factor” services, or it uses some made up measure	✓	×
The publisher falsely claims to have its content indexed in legitimate abstracting and indexing services or claims that its content is indexed in resources that are not abstracting and indexing services.	✓	×
Use boastful language claiming to be a “leading publisher” even though the publisher may only be a startup or a novice organization.	✓	✓
Operate in a Western country chiefly for the purpose of functioning as a vanity press for scholars in a developing country (e.g., utilizing a maildrop address or PO box address in the United States, while actually operating from a developing country).	✓	×
Have a “contact us” page that only includes a web form or an email address, and the publisher hides or does not reveal its location.	✓	✓
The publisher lists insufficient contact information, including contact information that does not clearly state the headquarters location or misrepresents the headquarters location (e.g., through the use of addresses that are actually mail drops).	✓	✓
The publisher publishes journals that are excessively broad (e.g., Journal of Education) in order to attract more articles and gain more revenue from author fees.	✓	×
The publisher publishes journals that combine two or more fields not normally treated together (e.g., International Journal of Business, Humanities and Technology)	×	×
The name of a journal does not adequately reflect its origin (e.g., a journal with the word “Canadian” in its name when neither the publisher, editor, nor any purported institutional affiliate relates whatsoever to Canada).	×	×
The publisher has poorly maintained websites, including dead links, prominent misspellings and grammatical errors on the website.	✓	×
The publisher makes unauthorized use of licensed images on their website, taken from the open web, without permission or licensing from the copyright owners.	✓	×
The publisher does not use standard identifiers (ISSN/DOI) or uses them improperly.	✓	×
The publisher uses names such as “Network,” “Center,” “Association,” “Institute,” and the like when it is only a solitary, proprietary operation and does not meet the definition of the term used or implied non-profit mission.	×	×
The publisher has excessive, cluttered advertising on its site to the extent that it interferes with site navigation and content access.	×	×
The publisher has no membership in industry associations and/or intentionally fails to follow industry standards.	✓	×
The publisher includes links to legitimate conferences and associations on its main website, as if to borrow from other organizations’ legitimacy, and emblazon the new publisher with the others’ legacy value.	×	×
The publisher or its journals are not listed in standard periodical directories or are not widely cataloged in library databases.	✓	×
None of the members of a particular journal’s editorial board have ever published an article in the journal.	×	×

(continued)

Table 4. (continued)

There is little or no geographic diversity among the authors of articles in one or more of the publisher’s journals, an indication the journal has become an easy outlet for authors from one country or region to get scholarly publications.	×	×
The publishers’ officers use email addresses that end in .gmail.com, yahoo.com, or some other free email supplier.	×	×
The publisher displays prominent statements that promise rapid publication and/or unusually quick peer review.	×	×
The publisher copies “authors guidelines” verbatim from other publishers.	✓	×

5 Conclusion and Future Work

In this work, we present an empirical analysis of a potential predatory open access publisher OMICS and compare it with well-known and highly reputed publisher BMC. We present facts with substantial evidence gathered from reputable sources to refute popular claims. We show that the entire predatory ecosystem is cautiously evolving to bypass the standard filters. Because the current work is only a preliminary attempt to study India-based potential predatory publishers extensively, future extensions could lead to similar studies on all possible dubious journals along with the development of web interfaces to visually represent these evidence.

References

1. Altmetric: Research Mentions (2019). <https://www.altmetric.com/about-altmetrics/what-are-altmetrics/>. Accessed 10 June 2019
2. Beall, J.: Predatory publishers are corrupting open access. *Nat. News* **489**(7415), 179 (2012)
3. BioMed Central: Open access publisher (2019). <https://www.biomedcentral.com/>. Accessed 10 June 2019
4. Bohannon, J.: Who’s afraid of peer review? (2013)
5. California South University: Private University (2019). <http://calu.us/>. Accessed 10 June 2019
6. Cosmos: Impact Factor Service Provider (2019). <http://www.cosmosimpactfactor.com/>. Accessed 10 June 2019
7. Crossref: Scholarly Reference Service (2019). <https://www.crossref.org/>
8. Davis, P.: Open access publisher accepts nonsense manuscript for dollars. *The scholarly kitchen* (2009). <http://scholarlykitchen.sspnet.org/2009/06/10/nonsense-for-dollars/>
9. Demir, S.B.: Predatory journals: who publishes in them and why? *J. Inf.* **12**(4), 1296–1311 (2018)
10. Elsevier: CiteScore: Journal Scoring Metric (2019). <https://www.elsevier.com/editors-update/story/journal-metrics/citescore-a-new-metric-to-help-you-choose-the-right-journal>. Accessed 10 June 2019
11. Federal Trade Commission: OMICS Group’s case (2019). <https://www.ftc.gov/enforcement/cases-proceedings/152-3113/federal-trade-commission-v-omics-group-inc>. Accessed 10 June 2019





12. Fister Jr., I., Fister, I., Perc, M.: Toward the discovery of citation cartels in citation networks. *Front. Phys.* **4**, 49 (2016)
13. Google Image: “Search by Image” service (2019). <https://www.google.com/imgph?hl=EN>
14. Gopalakrishnan Saroja, S., Santhosh Kumar, J., Hareesha, A.: India’s scientific publication in predatory journals: need for regulating quality of Indian science and education. *Curr. Sci.* **111**(11), 1759–1764 (2016)
15. Hern, A.: Predatory publishers: the journals that churn out fake science. *The Guardian*, August 2018. <https://www.theguardian.com/technology/2018/aug/10/predatory-publishers-the-journals-who-churn-out-fake-science>
16. ISSN Portal: Reference Service (2019). <https://portal.issn.org/> (2019)
17. Laine, C., Winker, M.A.: Identifying predatory or pseudo-journals. *Biochemia medica: Biochemia medica* **27**(2), 285–291 (2017)
18. Lead-Ratings: Gender guesser tool (2019). <https://github.com/lead-ratings/gender-guesser>. Accessed 10 June 2019
19. OMICS Online: Journal of cardiovascular diseases and diagnosis. OMICS group (2019). <https://www.omicsonline.org/cardiovascular-diseases-diagnosis.php>. Accessed 10 June 2019
20. OMICS Online: Open access publisher (2019). <https://www.omicsonline.org/>. Accessed 10 June 2019
21. Patwardhan, B., Nagarkar, S., Gadre, S.R., Lakhotia, S.C., Katoch, V.M., Moher, D.: A critical analysis of the ‘UGC-approved list of journals’. *Curr. Sci.* **114**(6), 1299 (2018)
22. Prime-Secretarial: “Hire An Address” service (2019). <https://www.prime-secretarial.co.uk/>. Accessed 10 June 2019
23. Public Website, University of Vienna: Prof. Muller’s profile (2019). <https://geschichte.univie.ac.at/en/persons/stephan-mueller-univ-prof-dr>. Accessed 10 June 2019
24. Raniwala, R.: ‘predatory’ is a misnomer in the unholy nexus between journals and plagiarism. *The Wire*, August 2018. <https://thewire.in/the-sciences/predatory-journals-fake-journals-plagiarism-peer-review-mhrd-ugc>
25. Scholarly: Google Scholar API (2019). <https://pypi.org/project/scholarly/>
26. Scimago: Journal Rankings Platform (2019). <https://www.scimagojr.com/journalrank.php>. Accessed 10 June 2019
27. Scopus: Journal index service (2019). <https://www.scopus.com/>. Accessed 10 June 2019
28. Scopus: Metric API (2019). <https://github.com/scopus-api/scopus>
29. Shakeri, S.: Fake school California South University is the university of Alberta’s evil twin. *Huffington Post*, March 2018. https://www.huffingtonpost.ca/2018/03/17/fake-school-california-south-university-is-the-university-of-albertas-evil-twin_a_23388464/
30. Shamseer, L., et al.: Potential predatory and legitimate biomedical journals: can you tell the difference? A cross-sectional comparison. *BMC Med.* **15**(1), 28 (2017)
31. Shen, C., Björk, B.C.: ‘Predatory’ open access: a longitudinal study of article volumes and market characteristics. *BMC Med.* **13**(1), 230 (2015)
32. Sorokowski, P., Kulczycki, E., Sorokowska, A., Pisanski, K.: Predatory journals recruit fake editor. *Nat. News* **543**(7646), 481 (2017)
33. Straumsheim, C.: No more ‘beall’s list’. *Inside Higher Ed*, January 2017. <https://www.insidehighered.com/news/2017/01/18/librarians-list-predatory-journals-reportedly-removed-due-threats-and-politics>

34. United Nations: Country Classification (2014). https://www.un.org/en/development/desa/policy/wesp/wesp_current/2014wesp_country_classification.pdf
35. Wikipedia: Omics publishing group (2019). https://en.wikipedia.org/wiki/OMICS_Publishing_Group (2019). Accessed 10 June 2019
36. Xia, J., Li, Y., Situ, P.: An overview of predatory journal publishing in Asia. *J. East Asian Libr.* **2017**(165), 4 (2017)
37. Yadav, S.: Fake science: face behind biggest of all - '40 countries, million articles'. *The Indian Express*, July 2018. <https://indianexpress.com/article/india/face-behind-biggest-of-all-40-countries-million-articles-fake-research-srinubabu-gedela-omics-5266830/>

Metadata and Entities



Identifying and Linking Entities of Multimedia Franchise on Manga, Anime and Video Game from Wikipedia

Kosuke Oishi¹, Tetsuya Mihara², Mitsuharu Nagamori²,
and Shigeo Sugimoto²

¹ Graduate School of Library, Information and Media Studies,
University of Tsukuba, 1-2 Kasuga, Tsukuba, Ibaraki, Japan
oishi.kosuke.ry@alumni.tsukuba.ac.jp

² Faculty of Library, Information and Media Science, University of Tsukuba,
1-2 Kasuga, Tsukuba, Ibaraki, Japan
{mihara, nagamori, sugimoto}@slis.tsukuba.ac.jp

Abstract. In Japan, manga, anime and video game (MAG) are viewed as popular culture. Many MAG works are published across media, i.e. shared characters, story, universe, etc. Entities and relationships describing multimedia franchise are helpful for general audiences to search MAG items. However, multimedia franchises are established by the cognition of audiences, it has been difficult for memory institutions to create authorized datasets of MAG multimedia franchise. In this paper, we propose a method to link institutional data and multimedia franchise entities from Wikipedia. Articles about MAG works in Wikipedia are created and edited by audiences. Many MAG articles are organized based on the structure of multimedia franchise. The proposed method is to identify of multimedia franchise entities automatically from patterns of relationships between these articles. This paper shows the result of the extraction and linking them to the bibliographic records of Media Arts Database (MADB) produced by the culture ministries of Japan.

Keywords: Popular culture database · Metadata aggregation · FRBRization

1 Introduction

Japanese comics (Manga), animation (Anime) and video games (Game) are globally known as Japanese popular culture. Hereafter in this paper, Manga, Anime, Game is collectively referred to as MAG unless otherwise noted. Popular titles in manga, anime or game are often used for creation in other media; for example, there are games and anime of Dragon Ball which was started as manga. This type of production across media is often found not only in MAG but also movies, novels, musicals, and so forth. Those works published across multiple media are called multimedia franchise, media franchise, cross-media, and media-mix.

There are some interesting questions about media franchise; how do users find a particular item such as a manga monograph and a game package from a comprehensive title of a multimedia franchise, and how do memory institutions organize bibliographic

information of the multimedia franchise works which would represent various different types of entities, i.e., a work as a conceptual entity representing an intellectual content of creation like FRBR Work and a work as a physical entity like FRBR Item. Tallerås [1] reports 89% of people have recognized the derivation relationship between the related multimedia franchise works and they have represented the hierarchical relationships of the works. Thus, users know relationships across multimedia franchise works but those relationships of works are out of the scope of the conventional bibliographic information created by memory institutions.

This study is aimed to develop a technology to automate the process of linking multimedia franchise entities and other bibliographic entities of MAG. The technology presented in this paper is rather simple and designed for Linked Open Data environments to help connect multimedia franchise products across the border of media. This study uses Wikipedia and Media Art Database (MADB) [2] being developed by Agency for Cultural Affairs (Bunka-cho) of the Japanese government.

2 Backgrounds and Related Works

2.1 Research Questions and Goals

MADB is a database service provided by Agency for Cultural Affairs covering four areas of popular culture in Japan – MAG and media arts. MADB provides a comprehensive set of metadata about MAG instances published and/or created in Japan. MADB aims an institutional metadata. Institutional metadata is created by memory institutions or created under guidelines which are similar to those given at memory institutions. Those institutional services are built primarily based on items of MAG while end-users tend to prefer higher-level such as FRBR *Work* and multimedia franchise titles. However, any of institutional metadata services including MADB satisfies this demand because How can we solve the gap between the end-users and MADB?

An answer to this question is to connect articles about MAG titles in Wikipedia to MADB which is an authoritative institutional metadata database created based on the items held by memory institutions or those items whose existence is proved by domain specialists. Thus, the linkage between Wikipedia/fan-created articles and MADB records is the main goal of this research.

This study uses the idea of Superwork model defined by Kiryakos [3, 12–14]. Superwork model is inspired by FRBR [4] and LRM [5]. In the Superwork model, *Superwork* is the top-most entity of the hierarchy. All entities in the Series layer and below are separately defined in each media. Kiryakos examined the structure of Wikipedia articles and relationships among them to propose *Superwork* as an entity to represent multimedia franchise. He also examined the structure of MADB to investigate consistency of the proposed model with existing item-based institutional metadata.

The basic trend of metadata for memory institutions is a shift to intellectual content-oriented description from item-oriented description. Each model is given a scope of description so that any single model cannot cover all media which brings us intellectual contents, e.g., books, anime and games. More importantly, in the Internet environment which is moving toward semantic linking based on the Linked Open Data, we need

good models to express intellectual contents and to link them to physically existing items which may be analog or digital. Thus, linking intellectual contents and physical items across media is an important issue but it is still lowly studied area.

2.2 Generating Datasets Based on Entity-Relation Data Model from Bibliographic Records

It is called FRBRization to automatically convert conventional bibliographic records to the data conforming FRBR. OCLC [6] proposes a Work-set algorithm, which is an algorithm for aggregating records with common keys, with pairs of normalized author names and titles. It is not easy to apply this method to MAG resources because this method requires an authority file, which is difficult to find in the MAG domain. He [7] shows FRBRization of Manga using the bibliographic records at the Kyoto International Manga Museum [8], complementing authority information for Manga by DBpedia. TelPlus project, V/FRBR project [9], Scherzo [10], DOREMUS project [11] are studies on FRBRization methods that create mapping rules between records and FRBR models and perform conversion based on them.

Unlike records grouping as in the Work-set algorithm, these methods can be applied in various domains and across domains because of their rule-based features. However, we do not see cross-domain attempts, in particular for MAG. One reason of this would be the lack of authoritative metadata resources in MAG.

3 Identifying Multimedia Franchises and Linking MAG Bibliographic Records

3.1 Identification of Multimedia Franchise Entities

In this study, we automatically generate a dataset from Wikipedia articles by extracting the structure of the articles. The organization of the dataset is defined based on the

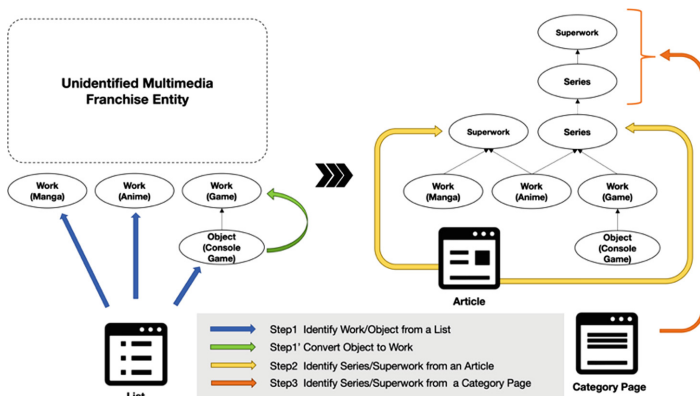


Fig. 1. Identification overview of entities and relationships using Wikipedia

Superwork model. The whole process is automated with heuristics and a few manually created datasets. Figure 1 shows an outline of the identification process of entities and relationships using the link structure of Wikipedia. We generate a hierarchical structure of MAG entities based on the Superwork model through this process. In the Japanese Wikipedia, entities which correspond to Works of manga and anime and *Object* of game of the Superwork model can be found in lists of MAG works: “Category: 漫画作品 (五十音別)”, “日本のテレビアニメ作品一覧”, “日本のアニメ映画作品一覧” and “Category: ゲーム機別ゲームタイトル一覧”. Those lists may be a stand-alone list article, a list embedded in an article or a table. Those Work and Object entities can be extracted from the lists (Step1). The listed articles describing these entities might have a link(s) to the articles about Series and Superwork entities of these Work/Object entities. Using this structure, the entities and relationships among the entities are identified (Step2). On the other hand, an article of Wikipedia has links to some categories in which the article is classified by authors of the Wikipedia articles. Since many authors create and use their own classifications in their articles, those categories are not always consistent with the contents of the Work. In order to avoid wrong categorization, some categories are selected as parents of Series/Superwork entities. “作品のカテゴリ”, “シリーズ作品”, “メディアミックス作品” are used as parents. Subcategories of these selected parent categories regard as candidates of Series/Superwork entities. However, these include categories that classify series articles such as “trilogy”, “series of battle action games” or “Super Smash Bros. ‘s appearance characters”. These inappropriate ones were excluded by using regular expressions and by MAG experts visually checking them individually. (Step3).

3.2 Linking Multimedia Franchise and MADB

In this study, item-oriented metadata defined as Object entity in Superwork model from MADB is linked to the hierarchical dataset of multimedia franchise after the creation shown 3.1. The proposed method is to identify the same work entity from the created hierarchical dataset same as these Work entities in MADB.

MADB has Object entities and Work entities in the metadata schema of manga and anime. It also has the relations of these two types of entities. On the other hand, it has only Object entities in the metadata schema of game. Thus, Work entity of game is created to find the set of Object entities which has the same title. To judge the linkage between these two entities from different data source, the similarity of two types of properties in each entity are measured. We use Levenshtein distance of normalized title and its Japanese syllabary (i.e. title yomi) for this similarity measurement. When the value of this indicator exceeds a certain threshold, link them as the same resource. this threshold value is set to 0.6.

4 The Result of Identification and Linking

This study attempts to create the practical dataset of multimedia franchise entities by applying the proposed method to the scraped data of Wikipedia on June 2018.

In the step.2 shown in 3.1, the method could misidentify Work/Series/Superwork entities from articles which is not for them, for example, described only about characters, or works are described in articles of a production company. To clarify the possibility of this misidentification, used 97 articles are checked visually by random sampling. 17 articles describe over two Work/Series/Superwork of MAGs. 9 articles describe Work/Series/Superwork of MAG, but it also contains works excluded MAG, e.g. novels or drama. They are regarded as correct results. On the other hand, 44 articles describe single work of MAG. 10 articles describe single work excluded MAG. 17 article are not existed. They are created only the link in the work lists.

Table 1. The number of entities and relationships obtained by implementation for Wikipedia

Entity Type		Number	Relationship Type	Number
<i>Superwork</i>	Linked more than 2 entities	3,469	<i>Superwork – Series</i>	4,043
	Linked only 1 entity	17,050	<i>Superwork – Work</i>	27,104
<i>Series</i>		5,649	<i>Series – Series</i>	2,375
<i>Work (Manga)</i>		14,666	<i>Series – Work</i>	10,269
<i>Work (Anime)</i>		5,849	<i>Work – Object (ConsoleGame)</i>	30,387
<i>Work (Game)</i>		24,340		
<i>Object (ConsoleGame)</i>		30,387		

(i) Entity

(ii) Relationship

Table 2. The number of relation aggregated MADB entities

	Type of count	Average	Median	Max
(i) Linking multimedia franchise entities	Relation	8.46	3	8,371
	MADB record	2.51	1	648
(ii) MADB records only	Relation	3.30	1	589
	MADB record	2.20	1	589

Table 1 shows the number of entities and entites obtained for the extraction. The multimedia franchise relationships are also checked visibly whether the upper and lower entities are correct or not from the label of the entity and other information in the article. As a result of the verification, 95/97 cases (97.9%) were correct. The one caused by the wrong linkage in Wikipedia, other one is linked to the page which is not about a work but a related person. As the result of linking work and entities in MADB to the extracted multimedia franchise entities, 16,176 manga, 7,881 anime and 40,268 game entities are linked. 97 links randomly Sampled from them are checked manually and 54/97 were correct. Table 2 shows the number of relations and linked MADB entities within one media franchise hierarchy. This compares two datasets of links, (i) the multimedia franchise entities and MADB entities which are identified as same ones,

and (ii) Aggregating entities from MADB by the relations in MADB only. The comparison of these two shows how multimedia franchise entities increase the aggregation of metadata in MADB by work. The average and median of the number of relationships are increasing. And in the 20,159 identified Superwork entities 1,718 cases were collecting 8 or more records of MADB, and the maximum was the *Gundam* series and it aggregates 648 records.

5 Discussion

As shown in Sect. 4, this study realizes to create the actual Object-to-Superwork hierarchy dataset by linking between media franchise entities and institutional metadata. 3,469 Superwork entities are confirmed as effective one to link work entities of deferent MAG works. Moreover, 1,718/20,519 Superwork entities aggregate 8 or more records. The ratio of effective Superwork entities (3,467/20,519) seems not to be high, there should be more effective one because the generated Superwork entities linked only 1 entity includes the one which links to works except MAG. Though this study focused on only for MAG, multimedia franchises consist of works in various media such as novels, drama and live action. They are also useful to link deferent multimedia works. From the result of the check for articles shown Sect. 4, about 25% Superwork entities are estimated to be effective ones.

The half of linking between the multimedia franchise entity and the MADB records is incorrect. This study applied simple method to used mainly the title and its Japanese syllabary for the linking to reduce calculation costs. Entities which are not same but has similar title were wrongly linked. Using additional parameters is expected to improve linking accuracy such as release date or release date of child elements.

There are several limitations of the proposed method caused by the data source. Because Wikipedia is being maintained manually, it depends greatly on whether there are fans who create articles about works. It is assumed that no article exists when the number of fans is small. In fact, for generated works, there are 14,666 in manga, 5,849 in anime and 24,430 in game. On the other hand, there are 88,017 in manga and 10,997 in anime and 23,514 in game (they can be created to aggregate records of deferent console but has same title) from MADB. It can be said that data from Wikipedia covers only 53.2% of animation and 16.7% of manga. The reliability of the described Wikipedia data is also crucial. It is impossible to completely exclude incorrect articles or links. Therefore, it is required to verify data by using data of a plurality of information sources.

6 Conclusion

In this study, we identify multimedia franchise entity based on Superwork model using article structure in Wikipedia and linking them to institutional metadata. In the future, it is expected that improvement in the scale and reliability of multimedia franchise data by combining multiple data sources other than Wikipedia. In addition, the multimedia franchise dataset can be expected to support academic researches on advanced recommendation and visualization of MAG information based on work content.

Acknowledgement. This work was supported by JSPS KAKENHI Grant Number JP18K11984 and JP18K18328.

References

1. Tallerås, K., Dahl, J.H.B., Pharo, N.: User conceptualizations of derivative relationships in the bibliographic universe. *J. Doc.* **74**(4), 894–916 (2018)
2. Media Art Database (Development Version). <https://mediaarts-db.bunka.go.jp/>
3. Kiryakos, S.: A study on improving bibliographic descriptions for objects of popular culture through multimedia franchise representation, hierarchical modeling, and metadata aggregation. Ph.D. thesis, Tsukuba University (2019)
4. IFLA Study Group on the Functional Requirements for Bibliographic Records: Functional Requirements for Bibliographic Records. <https://www.ifla.org/publications/functional-requirements-for-bibliographic-records>
5. IFLA Library Reference Model: A Conceptual Model for Bibliographic Information. https://www.ifla.org/files/assets/cataloguing/frbr-lrm/ifla-lrm-august-2017_rev201712.pdf
6. Bennett, R., Lavoie, B.F., O'Neill, E.T.: The concept of a work in WorldCat: an application of FRBR. *Libr. Collect. Acquisitions Tech. Serv.* **27**, 45–59 (2003)
7. He, W., Mihara, T., Nagamori, M., Sugimoto, S.: Identification of works of manga from bibliographic data of comic books using DBpedia. In: *Proceedings of Digital Library Workshop*, vol. 44, pp. 11–19 (2013)
8. Kyoto International Manga Museum. <https://www.kyotomm.jp/en/>
9. Manguinhas, H.M.A., Freore, N.M.A., Borbinha, J.L.B.: FRBRization of MARC records in multiple catalogs. In: *Proceedings of the 2010 Joint International Conference on Digital Libraries* (2010)
10. Notess, M., Dunn, J.W., Hardesty, J.L.: Scherzo: a FRBR-based music discovery system. In: *International Conference on Dublin Core and Metadata Applications*, pp. 182–183 (2011)
11. Achichi, M., Lisena, P., Todorov, K., Troncy, R., Delahousse, J.: DOREMUS: a graph of linked musical works. In: Vrandečić, D., et al. (eds.) *ISWC 2018. LNCS*, vol. 11137, pp. 3–19. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00668-6_1
12. Kiryakos, S., Sugimoto, S., Lee, J.H., Jett, J., Cheng, Y., Downie, J.S.: Towards a conceptual framework for superworks. In: *Proceedings of the 7th Conference of Japanese Association for Digital Humanities*, pp. 47–49 (2017)
13. Lee, J.H., et al.: Reconceptualizing superwork for improved access to popular cultural objects. *Proc. Assoc. Inf. Sci. Technol.* **55**(1), 274–281 (2018)
14. Kiryakos, S., Sugimoto, S.: The representation of a multimedia franchise as a single entity: contrasting existing bibliographic entities with web-based superwork portrayals. *LIBRES* **28**(2), 40–57 (2018)



Impact of OCR Quality on Named Entity Linking

Elvys Linhares Pontes^{1,2}(✉), Ahmed Hamdi², Nicolas Sidere²,
and Antoine Doucet²

¹ University of Avignon, Avignon, France
elvys.linhares-pontes@univ-avignon.fr

² University of La Rochelle, La Rochelle, France
{elvys.linhares-pontes, ahmed.hamdi, nicolas.sidere,
antoine.doucet}@univ-lr.fr

Abstract. Digital libraries are online collections of digital objects that can include text, images, audio, or videos. It has long been observed that named entities (NEs) are key to the access to digital library portals as they are contained in most user queries. Combined or subsequent to the recognition of NEs, named entity linking (NEL) connects NEs to external knowledge bases. This allows to differentiate ambiguous geographical locations or names (John Smith), and implies that the descriptions from the knowledge bases can be used for semantic enrichment. However, the NEL task is especially challenging for large quantities of documents as the diversity of NEs is increasing with the size of the collections. Additionally digitized documents are indexed through their OCRred version which may contains numerous OCR errors. This paper aims to evaluate the performance of named entity linking over digitized documents with different levels of OCR quality. It is the first investigation that we know of to analyze and correlate the impact of document degradation on the performance of NEL. We tested state-of-the-art NEL techniques over several evaluation benchmarks, and experimented with various types of OCR noise. We present the resulting study and subsequent recommendations on the adequate documents and OCR quality levels required to perform reliable named entity linking. We further provide the first evaluation benchmark for NEL over degraded documents.

Keywords: Named entity linking · Deep learning · Digital library · Indexing

1 Introduction

A named entity is a real-world object, such as persons, locations, organizations, etc. Named entities have been shown to be key to digital library access as they are contained in a majority of the search queries submitted to digital library portals. They were notably found in 80% of queries submitted to Gallica, the portal of the national library of France [2].

© Springer Nature Switzerland AG 2019

A. Jatowt et al. (Eds.): ICADL 2019, LNCS 11853, pp. 102–115, 2019.

https://doi.org/10.1007/978-3-030-34058-2_11

Collecting data from different sources leads to reveal the problem of duplicate and ambiguous information about named entities. Therefore they are often not distinctive since one single name may correspond to several entities. A disambiguation process is thus essential to distinguish named entities to be indexed in digital libraries.

Named Entity Linking (NEL) is the task of recognizing and disambiguating named entities by linking them to entries of a Knowledge Base (KB). Knowledge bases (e.g. Wikipedia¹, DBpedia [15], YAGO [23], and Freebase [1]) contain rich information about the world's entities, their semantic classes, and their mutual relationships. NEL is a challenging task because a named entity may have multiple surface forms, such as its full name, partial names, aliases, abbreviations, and alternate spellings [22]. Besides digital libraries, this task is important to several NLP applications, e.g. information extraction, information retrieval (for the adequate retrieval of ambiguous information), content analysis (for the analysis of the general content of a text in terms of its topics, ideas or categorizations), question answering and knowledge base population.

In a nutshell, NEL aims to locate mentions of an NE, and to accurately link them to the right entry of a knowledge base, a process that often requires disambiguation (see Fig. 1). A NEL system typically performs two tasks: named entity recognition (NER) and entity disambiguation (ED). NER extracts entities in a document, and ED links these entities to their corresponding entities in a KB. Until recently, the common approach of popular systems was to solve these two sub-problems independently. However, the significant dependency between these two tasks is ignored and errors caused by NER will propagate to the ED without the possibility of recovery. Therefore, recent approaches [13] propose the joint analysis of these sub-tasks in order to reduce the amount of errors.

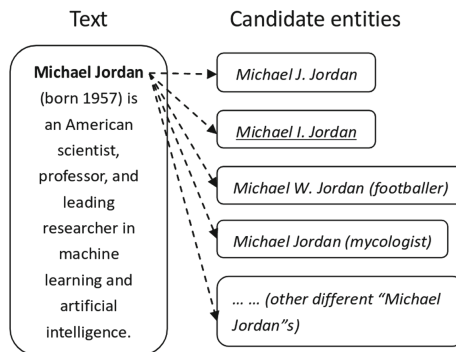


Fig. 1. An example of the named entity linking task. The named entity mention detected from the text is in bold face and the correct mapping entity is underlined (Shen *et al.* [22]).

¹ <http://www.wikipedia.org/>.

NEL in digital libraries is especially challenging due to the fact that most digitized documents are indexed through their OCRred version. This causes numerous errors due to the state of documents, following aging, bad storage conditions and/or the poor quality of initial printing materials. For instance, Chiron et al. [2] analyzed a collection of OCRred documents with 12M characters from 9 sources written in 2 languages. This collection is composed of documents from 1654–2000 and contains error rates that vary from 1% to 4%.

This paper aims to analyze the impact of OCR quality on the NEL task². Unfortunately, available corpora for NEL do not contain the noise presented in digital libraries. In order to overcome this problem, we simulated various OCRred versions of available data sets for NEL. Then, we tested state-of-the-art systems to investigate the impact of different OCR noises in NEL performance. The impact of OCR noise on NEL performance was as we expected: the greater the word and character errors, the greater the degradation of linking named entities to a knowledge base. However, we are now able to provide the DL community with recommendations on the OCR quality that is required for a given level of expected NEL performance.

Interestingly, the analyzed NEL approaches generated similar results for the character and the combination of all OCR degradation. This indicates that these systems have been able to analyze different OCR degradation without significantly reducing their performance.

The remainder of the paper is organized as follows: Sect. 2 makes a brief overview of the most recent and available NEL approaches in the state of the art. Then, available resources and our experimental evaluation are respectively described in Sects. 3 and 4. The paper is concluded in Sect. 5.

2 An Overview of Named Entity Linking

Given a knowledge base containing a set of named entities and a set of documents, the goal of named entity linking is to map each named entity in these documents to its corresponding named entity in a knowledge base [22]. NEL approaches can be divided into two classes:

- **Disambiguation approaches** only analyze gold standard named entities in a document and disambiguates them to the correct entry in a given KB [6, 14, 20].
- **End-to-end approaches** extract candidate entities from documents and then disambiguate them to the correct entries in a given KB [13].

Despite these two different classes, most works in the state of the art are based on three modules: candidate entity generation, candidate entity ranking, and unlinkable mention prediction [22]. More precisely, the first module aims to retrieve related entity mentions in KB that refer to mentions in a document.

² To the best of our knowledge, no studies have been conducted on NEL using OCRred documents.

Several works are based on name dictionary-based techniques [7], surface form expansion from the local document [25], and methods based on search engine [9] to identify candidate entities.

After selecting candidate entities, the second module attempts to rank the most likely link in KB for a mention. The main approaches are mainly based on supervised and unsupervised methods. These methods consider various techniques to analyze and rank entities, e.g. name string comparison [26], entity popularity [7], entity type [4], textual context [16], and coherence between mapping entities [3]. Finally, the last module validates whether the top-ranked entity identified in the candidate entity ranking module is the target entity for a mention.

Recent neural network methods [6,14] have established state-of-the-art results, out-performing engineered features based models. These methods use deep learning³ to analyze features, relationships, and complex interactions among features which allow a better analysis of documents and improve their performance. These methods combine context-aware word, span and entity embeddings with neural similarity functions to analyze the context of mentions and disambiguate correctly them to a KB.

Next subsections describe relevant and available NEL systems. Section 2.1 makes a brief description of disambiguation approaches and Sect. 2.2 focuses on the end-to-end approaches.

2.1 Disambiguation Approaches

Disambiguation approaches consider having already identified the named entities in the documents. In this case, these approaches aim to analyze the context of these entities to disambiguate them in a KB. In this context, Ganea and Hofmann [6] proposed a deep learning model for joint document-level entity disambiguation⁴. In a nutshell, they embed entities and words in a common vector space and use a neural attention mechanism over local context windows to select words that are informative for the disambiguation decision. Their model contains a conditional random field that collectively disambiguates the mentions in a document (Fig. 2).

Inspired on the Ganea and Hofmann’s approach [6], Le and Titov [14] treated relations between mentions as latent variables in their neural NEL model⁵. They rely on representation learning and learn embeddings of mentions, contexts, and relations in order to reduce the amount of human expertise required to construct the system and make the analysis more portable across languages and domains.

Raiman and Raiman [20] proposed a system for integrating symbolic knowledge into the reasoning process of a neural network through a type system⁶.

³ Including all types of deep learning techniques, such as transfer, reinforcement and multi-task learning.

⁴ The code is publicly available: <https://github.com/dalab/deep-ed>.

⁵ The code is publicly available: <https://github.com/lephong/mulrel-nel>.

⁶ The code is publicly available: <https://github.com/openai/deeptype>.

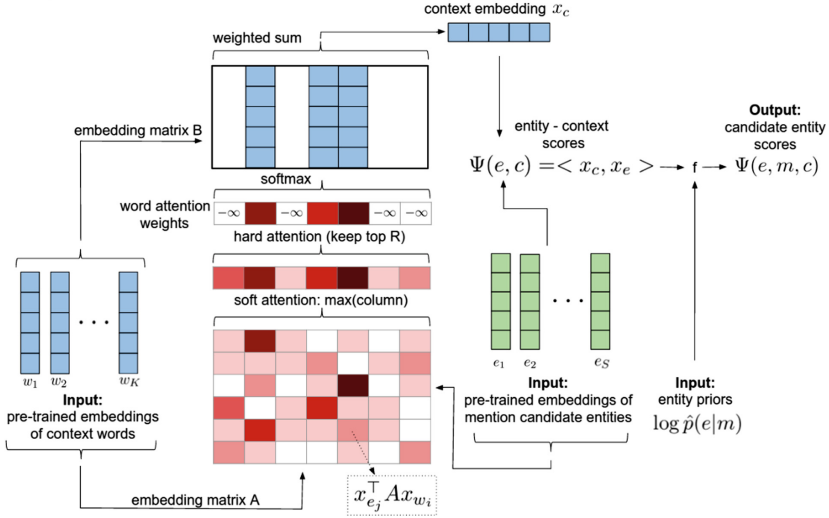


Fig. 2. Architecture of the Ganea and Hofmann’s approach. Their method uses a local model with neural attention to process context word vectors, candidate entity priors, and embeddings to generate the candidate entity scores (Ganea and Hofmann [6]).

They constrained the behavior to respect the desired symbolic structure, and automatically design the type system without human effort. Their model first uses heuristic search or stochastic optimization over discrete variables that define a type system informed by an Oracle and a learnability heuristic. Then, classifier parameters are fitted using gradient descent.

2.2 End-to-End Approach

Following the idea of jointly analyzing the NER and ED tasks, Kolitsas *et al.* [13] proposed a neural end-to-end NEL system that jointly discovers and links entities in a document⁷. Their model replaces engineered features by neural embeddings. They first generate all possible spans (mentions) that have at least one possible entity candidate. Then, each mention-candidate pair receives a context-aware compatibility score based on word and entity embeddings [6] coupled with neural attention and a global voting mechanism.

3 Resources

To the best of our knowledge, there is no publicly available corpora in the literature that are addressed to both named entity linking and post-OCR correction. There are either corpora where named entities are well recognized and linked,

⁷ The code is publicly available: https://github.com/dalab/end2end_neural.el.

but the text is not noisy or conversely, there are corpora where the text generated by an OCR process is aligned with the original text, but named entities are not annotated. Therefore we started from existing NEL corpora to build their versions in noisy context. We present in the following two subsections the available NEL corpora used in this work and a method to synthesize and inject OCR degradation into these corpora.

3.1 Original Data Sets

Available corpora for entity linking are mainly divided into six data sets:

- **AIDA-CoNLL** data set [10] is based on CoNLL 2003 data that was used for NER task. This data set is divided into AIDA-train for training, AIDA-A for validation, and AIDA-B for testing. This data set contains 1393 Reuters news articles and 27817 linkable mentions.
- **AQUAINT** data set [8, 18] is composed of 50 short news documents (250–300 words) from the Xinhua News Service, the New York Times, and the Associated Press. This data set contains 727 mentions.
- **ACE2004** data set [8, 21] is a subset of the ACE2004 coreference documents with 57 articles and 306 mentions, annotated through crowdsourcing.
- **MSNBC** data set [3, 8] is composed of 20 news articles from 10 different topics (two articles per topic: Business, U.S. Politics, Entertainment, Health, Sports, Tech & Science, Travel, TV News, U.S. News, and World News), having 656 linkable mentions in total.
- Finally, **CWEB** [5] and **WIKI** [21] data sets are composed of 320 documents each.

3.2 Simulated Data Sets

As we mentioned above, our work faces the lack of appropriate resources for named entity linking over OCRed documents. In order to overcome this problem, we started with corpora described in Sect. 3.1 and we then injected to each of them many text degradation caused by an OCR engine. We have built such ground truth as follow:

- clean texts have been converted into images
- images are contaminated using common degradation related to storage conditions and use of scanners [12]
- an OCR system is used to extract the text (we have used tesseract open source OCR engine v-4.0.0⁸).

We have used the DocCreator tool [12] to add four common types of OCR degradation that may be present on digital libraries material:

- **bleed-through** is typical degradation of double-sided pages. It simulates the ink from the back side that seeps through the front side.

⁸ <https://github.com/tesseract-ocr>.

- **phantom degradation** simulates degradation in worn documents. Following successive uses, some characters can be progressively eroded. The digitization process generates phantom ink around characters.
- **character degradation** simulates degradation due to the age of the document or the use of a scanner incorrectly set. It consists in adding small ink spots on characters and can induce the erasure of characters.
- **blurring** simulates a blurring effect, as can be encountered during a typical digitization process with focus issue.

These degradation allowed building four versions for each NEL corpus. We additionally define two versions that we call respectively LEV-0 and LEV-MIX. The LEV-0 version corresponds to re-OCRred version of original images with no degradation added. It aims to evaluate the OCR engine through sharp images. Whereas the LEV-MIX version is the result of simultaneously applying the four types of degradation to the original texts⁹. Figure 3 shows the impact of the different types of degradation on the sharpness of images comparing to the original one.

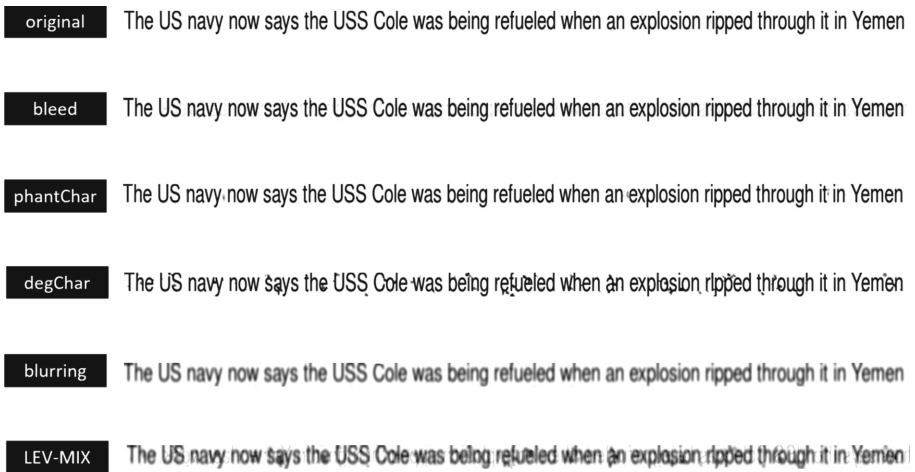


Fig. 3. Original image versus degraded images.

Following the text extraction by the OCR, noisy text have been aligned to its original version using the tool RETAS [24]. This tool allows to make this correspondence at character and word levels. Figure 4 shows an example of such alignment made between the ground truth and its OCRred version. The alignment reflects the various errors made by the OCR engine. The difference between the two texts are denoted by the presence of the character ‘@’. Each ‘@’ in the

⁹ Real-world documents contain simultaneously several OCR degradations. Therefore, the LEV-MIX version represents better the degraded documents on digital libraries.

ground truth indicates the insertion of one character by the OCR while ‘@’ in the noisy text indicates that one character has been deleted from the original text. All data used in this work are available for public¹⁰.

- **OCR:** The US navy now says the Uss@@ Cole was D@eing refueled when an explosion HOE@@@@ through it in Yenn@en
- **GT :** The US navy now says the U@@SS Cole was @being refueled when an explosion @@@ripped through it in Ye@@men
- **OCR:** last week, killing 17. The revi:@@@ accounting of the incident was given in a navy statement Friday
- **GT :** last week, killing 17. The revi@sed accounting of the incident was given in a navy statement Friday

Fig. 4. Text alignment: ground truth and OCR output.

In order to evaluate the quality of the text generated by an OCR engine, one of the most popular measures is Character Error Rate (CER) [11]. The CER is the proportion of errors made on characters compared to the original text. There are three types of errors:

- **Substitution** where one character has been replaced by another.
- **Deletion** where a character has simply not been recognized by the OCR.
- **Insertion** where an additional character has been wrongly added.

Table 1 details the OCR error rates at the character levels in the different data sets.

Table 1. CER on the OCRred versions of data sets.

OCR deg.	AIDA	ACE2004	AQUAINT	CWEB	MSNBC	WIKI	Average
LEV-0	1.0	0.8	0.4	1.3	1.1	0.6	0.9
Bleed	1.0	0.7	0.4	1.2	0.3	0.6	0.7
Blur	1.8	1.3	0.6	2.6	0.9	1.1	1.4
CharDeg	3.4	2.5	2.1	3.0	2.3	2.5	2.6
PhantChar	1.1	0.8	0.5	1.3	0.5	0.7	0.8
LEV-MIX	4.8	4.8	2.5	5.4	3.1	3.2	4.0

¹⁰ We provide for each test corpus the degraded images, the noisy texts extracted by the OCR and their aligned version with clean data by following this link: <https://zenodo.org/record/3490333>

The CER varies between 0.4% and 5.4% according to the documents and the type of degradation added. However, the noise distributed on the documents is homogeneous. It leads to a higher rate of error on words. The Word Error Rate (WER) is another measure used to translate the quality of a corpus [17]. Unlike the CER, this measure is the rate of correctly recognized words. Two words are different if they differ in at least one character. Table 2 describes the WER error rates in the different corpora.

Table 2. WER on the OCRed versions of data sets.

OCR deg.	AIDA	ACE2004	AQUAINT	CWEB	MSNBC	WIKI	Average
LEV-0	3.9	3.0	2.0	3.1	2.3	3.0	2.9
Bleed	3.9	3.0	1.9	2.7	1.6	3.0	2.7
Blur	5.4	3.8	2.2	5.8	3.2	3.5	4.0
CharDeg	16.9	14.4	13.4	13.6	14.3	14.1	14.4
PhantChar	5.6	4.6	4.0	4.3	3.4	4.5	4.4
LEV-MIX	18.2	15.4	13.7	16.6	15.4	14.4	15.6

As shown in Tables 1 and 2, among OCR degradation, the character degradation showed to be the most critical noise by generating the highest error rate at the character and word levels. Interesting, the bleed version of data sets generated similar or smaller CER and WER values than their LEV-0 version. Finally, the combination of OCR degradation (LEV-MIX) achieved the worst CER and WER.

4 Experimental Evaluation

In this section, we introduce the results of our experiments with different types of injected noise.

4.1 Automatic Evaluation

The main evaluation measures for entity linking systems are precision, recall and F1-measure. Precision is the fraction of correctly linked entity mentions that are generated by a system. Recall takes into account all entity mentions that should be linked and determines how correct linked entity mentions are with regard to total entity mentions that should be linked. Finally, the F1-measure is defined as the harmonic mean of precision and recall.

These measures can be calculated on a full corpus (micro-averaging) or averaged by document (macro-averaging)¹¹. Since knowledge bases contain millions of entities, for simplicity, we focused on mention-entity pairs for which the ground-truth related to known entity.

¹¹ In this paper, we relied on the micro F1 scores.

4.2 Training Settings

In order to analyze the effects of OCR degradation, we tested the systems of Ganea and Hofmann 2017’s [6] and Le and Titov’s [14]. We used the pre-trained models of each system. More precisely, Ganea and Hofmann trained their entity embeddings on the Wikipedia (Feb 2014) corpus and they used the pre-trained Word2Vec word vectors with 300 dimensions¹². Le and Titov used GloVe [19] word embeddings trained on 840 billion tokens and they selected the best 7 relations on the development scores. Both of them trained their systems on the AIDA-train data set.

4.3 Entity Linking over Clean and Noisy Data

Tables 3 and 4 respectively show the results of the systems of Ganea and Hofmann’s [6] and Le and Titov’s [14] on the OCRred data sets. As we expected, the CER and WER generated by OCR degradation are roughly correlated to the performance of NEL. LEV-0 and bleed data sets have low CER and WER levels which produced less impact on the F1 results. Character degradation has larger CER and WER values which caused a bigger drop in the NEL performance. Among all data sets, ACE2004 was the most affected by OCR degradation (a drop of almost 20% points for the character degradation).

Table 3. NEL results (micro F1 scores) using the Ganea and Hofmann 2017’s system [6] on the OCRred versions of data sets.

OCR deg.	AIDA	ACE2004	AQUAINT	CWEB	MSNBC	WIKI	Average
Clean	0.9144	0.8893	0.9021	0.7774	0.9350	0.7799	0.8663
LEV-0	0.9039	0.7906	0.8867	0.7372	0.9168	0.7491	0.8307
Bleed	0.9039	0.7906	0.8906	0.7348	0.9171	0.7490	0.8310
Blur	0.9044	0.7934	0.8856	0.7237	0.9106	0.7379	0.8259
CharDeg	0.9016	0.6873	0.7902	0.6938	0.8498	0.6597	0.7637
PhantChar	0.9039	0.7867	0.8802	0.7344	0.9101	0.7462	0.8269
LEV-MIX	0.9036	0.6856	0.8043	0.6798	0.8456	0.6560	0.7625

The LEV-MIX version of the data sets generated the highest CER and WER values with all kinds of character and word errors presented in the four OCR degradations. Despite these high error rates, the combination of these OCR degradations does not appear to have a significant change in the performance of two NEL systems. Both systems achieved satisfactory results (average micro F1-score greater than 0.75) for all OCRred versions.

The analysis of the relations of mentions as latent variables improved the analysis of candidates mentions and increased F1 scores for almost all clean

¹² <http://bit.ly/1R9Wsqr>.

Table 4. NEL results (micro F1 scores) using the Le and Titov’s system [14] on the OCRred versions of data sets.

OCR deg.	AIDA	ACE2004	AQUAINT	CWEB	MSNBC	WIKI	Average
Clean	0.9306	0.9014	0.8923	0.7773	0.9411	0.7783	0.8702
LEV-0	0.9071	0.8000	0.8683	0.7434	0.9314	0.7446	0.8325
Bleed	0.9091	0.8000	0.8723	0.7415	0.9318	0.7443	0.8332
Blur	0.9037	0.7934	0.8671	0.7282	0.9253	0.7344	0.8253
CharDeg	0.8984	0.7028	0.7795	0.6931	0.8610	0.6585	0.7655
PhantChar	0.9071	0.7915	0.8602	0.7401	0.9270	0.7393	0.8275
LEV-MIX	0.8985	0.6959	0.7936	0.6838	0.8613	0.6549	0.7647

data sets (except AQUAINT and WIKI data sets). However, errors produced by the OCR degradation were more critical for this analysis. Indeed, the mentions affected by OCR errors degraded the analysis of the relationships between other mentions that led to a decrease in the performance of this approach.

In order to better evaluate the impact of OCR quality on NEL results, we calculated for both systems used in this work the δ measure which represents the average decrease rate between the F1-score given in clean data and the F1-scores given in noisy data for each degradation. For both systems, δ do not exceed 11% in noisy data. The maximum decrease of NEL F1-score is given by the LEV-MIX degradation.

Figure 5 shows the evolution of the δ measure according to degradation. Types of degradation have been sorted according to OCR rates. CER and WER curves are also given for comparative reasons. We refer by δ_1 and δ_2 the decrease measures of NEL F1-scores given by Ganea and Hofmann system and Le and Titov’s system respectively. Both systems achieved good NEL performances (a

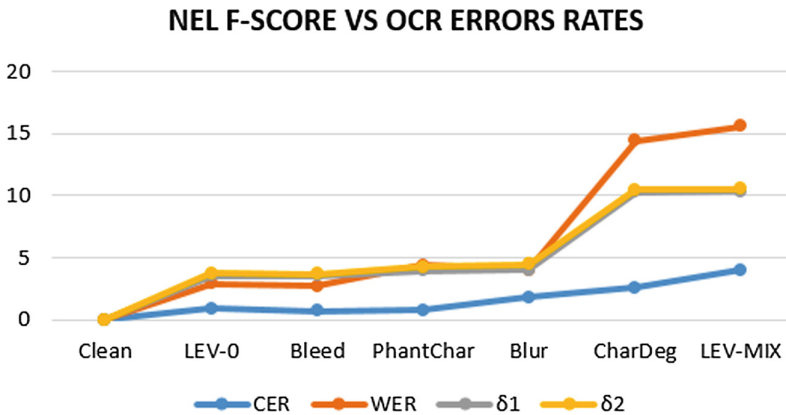


Fig. 5. NER F-score degradation according to OCR error rates

drop of up to 0.05 in F-score values) when the WER is below 5%. For higher WER values, the loss of NEL performance doubled.

Despite the complexity of the NEL task and the occurrence of several types of errors in the documents of digital libraries, the systems achieved interesting results. This proves that they can be used to distinguish ambiguous named entities in degraded documents. Some word correction strategies, such as auto-encoders, language models, and so on, could be used to decrease the impact of OCR degradation on NEL.

5 Conclusion

This paper is the first attempt that we know of that ever investigated the impact of document degradation on the task of named entity linking.

The broad collection of topics that may be covered in digital libraries implies that simply recognizing named entities is not sufficient to identify them. Named entity linking can contribute to solving this problem by analyzing the context of these entities and disambiguating them thanks to a knowledge base. We have tested two available NEL systems on six available data sets with OCR degradation. Despite the character and word error rates of OCR degradation, these two systems achieved satisfying results for NEL. Some of these OCR degradations generated high error rates, which reduced system performance.

We were able to provide the DL community with recommendations on the OCR quality that is required for a given level of expected NEL performance. We further provided the research community with the first data set and benchmark for the evaluation of NEL performance in a noisy context.

Further work is under progress to analyze the performance of end-to-end NEL systems on OCRred data sets. More precisely, we want to investigate how different approaches can overcome the problem of OCR degradation and provide correct predictions. We also intend to analyze and to test the performance of these systems using real data on low-resources languages.

Acknowledgments. This work has been supported by the European Union’s Horizon 2020 research and innovation program under grant 770299 (NewsEye).

References





1. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: a collaboratively created graph database for structuring human knowledge. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, pp. 1247–1250. ACM, New York (2008). <https://doi.org/10.1145/1376616.1376746>
2. Chiron, G., Doucet, A., Coustaty, M., Visani, M., Moreux, J.P.: Impact of OCR errors on the use of digital libraries: towards a better access to information. In: Proceedings of the 17th ACM/IEEE Joint Conference on Digital Libraries, pp. 249–252. IEEE Press (2017)

3. Cucerzan, S.: Large-scale named entity disambiguation based on Wikipedia data. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pp. 708–716. Association for Computational Linguistics, Prague, June 2007. <https://www.aclweb.org/anthology/D07-1074>
4. Dredze, M., McNamee, P., Rao, D., Gerber, A., Finin, T.: Entity disambiguation for knowledge base population. In: Proceedings of the 23rd International Conference on Computational Linguistics, COLING 2010, pp. 277–285. Association for Computational Linguistics, Stroudsburg (2010). <http://dl.acm.org/citation.cfm?id=1873781.1873813>
5. Gabrilovich, E., Ringgaard, M., Subramanya, A.: FACC1: Freebase annotation of ClueWeb corpora, Version 1 (Release date 2013–06-26, Format version 1, Correction level 0), June 2013
6. Ganea, O.E., Hofmann, T.: Deep joint entity disambiguation with local neural attention. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pp. 2619–2629. Association for Computational Linguistics (2017). <https://doi.org/10.18653/v1/D17-1277>. <http://aclweb.org/anthology/D17-1277>
7. Guo, S., Chang, M.W., Kiciman, E.: To link or not to link? A study on end-to-end tweet entity linking. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 1020–1030. Association for Computational Linguistics, Atlanta, June 2013. <https://www.aclweb.org/anthology/N13-1122>
8. Guo, Z., Barbosa, D.: Robust entity linking via random walks. In: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, CIKM 2014, pp. 499–508. ACM, New York (2014). <https://doi.org/10.1145/2661829.2661887>
9. Han, X., Zhao, J.: NLPR_KBP in TAC 2009 KBP track: a two-stage method to entity linking. In: Proceedings of Test Analysis Conference 2009 (TAC 2009). MIT Press (1999)
10. Hoffart, J., et al.: Robust disambiguation of named entities in text. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, pp. 782–792. Association for Computational Linguistics, Stroudsburg (2011). <http://dl.acm.org/citation.cfm?id=2145432.2145521>
11. Holley, R.: How good can it get? Analysing and improving OCR accuracy in large scale historic newspaper digitisation programs. *D-Lib Mag.* **15**(3/4) (2009)
12. Journet, N., Visani, M., Mansencal, B., Van-Cuong, K., Billy, A.: Doccreator: a new software for creating synthetic ground-truthed document images. *J. Imaging* **3**(4), 62 (2017)
13. Kolitsas, N., Ganea, O.E., Hofmann, T.: End-to-end neural entity linking. In: Proceedings of the 22nd Conference on Computational Natural Language Learning, pp. 519–529. Association for Computational Linguistics (2018). <http://aclweb.org/anthology/K18-1050>
14. Le, P., Titov, I.: Improving entity linking by modeling latent relations between mentions. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1595–1604. Association for Computational Linguistics (2018). <http://aclweb.org/anthology/P18-1148>
15. Lehmann, J., et al.: DBpedia - a large-scale, multilingual knowledge base extracted from Wikipedia. *Semant. Web J.* **6**(2), 167–195 (2015). <http://jens-lehmann.org/files/2015/swj.dbpedia.pdf>

16. Li, Y., Wang, C., Han, F., Han, J., Roth, D., Yan, X.: Mining evidences for named entity disambiguation. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2013, pp. 1070–1078. ACM, New York (2013). <https://doi.org/10.1145/2487575.2487681>
17. Lund, W.B., Kennard, D.J., Ringger, E.K.: Combining multiple thresholding binarization values to improve OCR output. In: Document Recognition and Retrieval XX, vol. 8658, p. 86580R. International Society for Optics and Photonics (2013)
18. Milne, D., Witten, I.H.: Learning to link with Wikipedia. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM 2008, pp. 509–518. ACM, New York (2008). <https://doi.org/10.1145/1458082.1458150>
19. Pennington, J., Socher, R., Manning, C.: Glove: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)
20. Raiman, J., Raiman, O.: Deeptype: multilingual entity linking by neural type system evolution. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI 2018), the 30th Innovative Applications of Artificial Intelligence (IAAI 2018), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI 2018), New Orleans, Louisiana, USA, 2–7 February 2018, pp. 5406–5413 (2018). <https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17148>
21. Ratinov, L., Roth, D., Downey, D., Anderson, M.: Local and global algorithms for disambiguation to Wikipedia. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, HLT 2011, vol. 1, pp. 1375–1384. Association for Computational Linguistics, Stroudsburg (2011). <http://dl.acm.org/citation.cfm?id=2002472.2002642>
22. Shen, W., Wang, J., Han, J.: Entity linking with a knowledge base: issues, techniques, and solutions. *IEEE Trans. Knowl. Data Eng.* **27**(2), 443–460 (2015). <https://doi.org/10.1109/TKDE.2014.2327028>
23. Suchanek, F.M., Kasneci, G., Weikum, G.: Yago: a core of semantic knowledge. In: Proceedings of the 16th International Conference on World Wide Web, WWW 2007, pp. 697–706. ACM, New York (2007). <https://doi.org/10.1145/1242572.1242667>
24. Yalniz, I.Z., Manmatha, R.: A fast alignment scheme for automatic OCR evaluation of books. In: 2011 International Conference on Document Analysis and Recognition (ICDAR), pp. 754–758. IEEE (2011)
25. Zhang, W., Sim, Y.C., Su, J., Tan, C.L.: Entity linking with effective acronym expansion, instance selection and topic modeling. In: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, IJCAI 2011, vol. 3, pp. 1909–1914. AAAI Press (2011). <https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-319>
26. Zheng, Z., Li, F., Huang, M., Zhu, X.: Learning to link entities with knowledge base. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, HLT 2010, pp. 483–491. Association for Computational Linguistics, Stroudsburg (2010). <http://dl.acm.org/citation.cfm?id=1857999.1858071>



Authoring Formats and Their Extensibility for Application Profiles

Nishad Thalhath¹^(✉), Mitsuharu Nagamori², Tetsuo Sakaguchi²,
and Shigeo Sugimoto²

¹ Graduate School of Library, Information and Media Studies,
University of Tsukuba, Tsukuba, Japan

nishad@slis.tsukuba.ac.jp

² Faculty of Library, Information and Media Studies,
University of Tsukuba, Tsukuba, Japan

{nagamori,saka,sugimoto}@slis.tsukuba.ac.jp,

<https://www.slis.tsukuba.ac.jp>

Abstract. Metadata application profiles act as a key element in interoperability of metadata instances. There are various accepted formats to express application profiles. Most of these expression formats such as RDF, OWL, JSON-LD, SHACL, and ShEx are machine-actionable, and formats like spreadsheets or web pages act as human-readable documentation. Due to limited options in the mutual conversion of these expression formats, they are often created independent of each other and thus makes the process of application profile creation and maintenance an expensive process requiring sophisticated skills and time. Proposals for convertible authoring formats to create application profiles have received less acceptance, mainly due to their inability in addressing various use cases and requirements. The authors describe a new authoring format based on YAML and evaluate the scope of extensibility of authoring formats by comparing this with the capabilities of two already available proposals; Simple DSP, Bibframe Editor. The comparison includes an attempt to recreate real-world application profiles with an extensible format. This study argues that the extensible authoring formats are suitable for creating application profiles with custom requirements and different use cases. This would promote the acceptance of application profiles by reducing the associated cost and skill requirements in creation, maintenance, and publishing of application profiles.

Keywords: Metadata · Metadata schema · Application profiles · Schema versioning · Semantic web · Linked data

1 Introduction

Metadata application profiles (MAP) are described as schemas of data elements from different namespaces combined and optimized for a specific local application [7]. They are a suitable mechanism to document and explain a metadata instance

by describing the set of metadata elements, policies, guidelines, vocabularies for that specific domain or implementation, schemas, policies, and constraints. Another vital role of an application profile is to provide the specification of term usage and ensure interoperability by expressing domain consensus, alignment, and the structure of the data [3,8].

1.1 Application Profile Expression and Authoring Formats

Dublin Core Metadata Initiative (DCMI) defines one of the earliest guidelines to express application profiles as Description Set Profiles (DSP), a constraint language for Dublin Core Application Profiles (DCAP) based on the Singapore framework for application profiles [10]. XML or RDF can be used as an expression format for DSP. Singapore framework recommends publishing the application profiles also as human-readable, with detailed usage guidelines to maximize reusability and interoperability. For the machine-actionable expressions, other than the XML and RDF, new standards are emerging and being widely accepted like JSON-LD, OWL, ShEx or SHACL.

An authoring format of application profile can be defined as a source for application profile publication, which helps to generate different application profile expression formats using a processor or a converter. Authoring formats cannot be treated as an expression of an application profile. There were different attempts to define and develop authoring formats or tools for creating application profiles. The first notable attempt is by DCMI. A translator with MoinMoin Wiki Syntax as an authoring format for DSP to integrate into webpages was introduced [6]. MetaBridge project used a spreadsheet-based application profile authoring format named Simple DSP (SDSP) [9]. One of the recent developments is an ongoing initiative named RDF-AP, which at the moment utilizes CSV-based notations to author application profiles [5]. The BIBFRAME project from the Library of Congress has developed web-based profile editors named BIBFRAME Profile Editor, which allows to modify or create profiles [1]. BIBFRAME editor is mainly intended to create the profiles of the BIBFRAME vocabulary but can be usable for other profiles as well. Linked Data for Production 2 (LD4P2) project published a modified BIBFRAME editor as a general online application profile authoring tool named Sinopia Profile Editor [12].

All the above-mentioned authoring formats are not extensible. Extensibility of a format is critical for its acceptance, which helps various communities to adopt a simple base format and bring in changes from the domain-specific requirements. Also, it helps to generate various standard formats from the same source document without just depending on the mutually inclusive elements.

1.2 Yet Another Metadata Application Profile (YAMA) and Extensibility

Yet Another Metadata Application Profile (YAMA) is proposed by the authors as an extensible authoring format for application profile, which addresses some of the shortcomings of the previous proposals [13]. YAMA uses YAML Ain't

Markup Language (YAML) [4] As an attempt to solve the absence of an application profile authoring workflow. YAMA is extensible with custom elements and structure. This will help to extend the capabilities of YAMA without any large-scale implementation changes. Any such extension is possible within the scope of YAML specification. YAMA is based on DC-DSP and includes a structured syntax to record modifications of a YAMA document named as change-sets. YAMA change-sets can be used to record changes of a MAP over any other versions and adapted from RFC 6902 JavaScript Object Notation (JSON) Patch [11]. In this paper, the authors evaluated the extensibility of YAMA by attempting to recreate a real-world application profile and compared the process with two other non-extensible application profile authoring formats.

2 Methods

To evaluate the advantage of extensible authoring formats for application profiles, three different formats were compared against the use cases for YAMA. The formats used for comparison are Simple DSP (SDSP) [9], Bibframe Editor [1] and YAMA [13]. The three formats use spreadsheets, web interface, and text editor as an authoring environment to express the source. Detailed comparison of these three authoring formats is given in Table 1.

Table 1. Feature comparison of three authoring formats used for evaluation.

	SDSP	Bibframe	YAMA
Based on	DC-DSP	Bibframe	DC-DSP
Format	CSV	JSON	YAML
Standard	RFC-4180	RFC-8259, ECMA-404	YAML 1.2
Strict to standard	No	Yes	Yes
Comments	Yes	No	Yes
Blank lines	Yes	Yes	Yes
Line Diff	Partial	Yes	Yes
Standard library compatibility	Partial	Yes	Yes
Native logic	No	Yes	No
Text editor compatibility	Partial	Yes	Yes
Native Syntax highlighting	No	Yes	Yes
Custom Editor	No	Yes	No
Proposed Editor	Spreadsheet	Bibframe Profile Editor	Text Editor
Schema Validation	No	Possible	Possible
Readability	Low	Low	High
Extensible	No	No	Yes

The DCAT Application Profile (DCAT-AP) for data portals in Europe [2] was used as sample application profiles for the evaluation, based on its

popularity, active maintenance, and the availability of previous versions. DCAT-AP is released as human-readable documentation as well in different machine-actionable expression formats such as JSON-LD, RDF, and SHACL. Since the source of this application profile is not available, it was assumed that DCAT-AP is created not using any authoring tools. The authors attempted to recreate DCAT-AP using all three authoring formats and generated standard application profile expression formats with similar tools. An analysis of the outputs was conducted by comparing the generated expression formats to its originally published versions and documentation to identify if an extensible authoring format can express real-world application profiles better than its non-extensible counterparts.

3 Results

Three authoring formats were used in creating a known application profile (DCAT-AP) sources from the documentation and their capability to include the documented application profile features and details were evaluated. The results shown from the investigation can be categorized into (a) the capability of including details from the documentation, (b) convenience as an authoring format, (c) capabilities of producing relevant expression formats.

These formats could represent part of the documentation, but mostly the textual explanations and general sections from the documentation is ignored for convenience, with an assumption that it is manually maintained. All three authoring formats are initially designed for the Dublin Core Application Profile (DCAP), and the application profile evaluated was not entirely Dublin Core Application Profiles. However, for the evaluation, the authors attempted to include maximum information without altering the DCAP structure, wherever possible.

A general observation from the expression capability is summarized as follows: all three formats could express the essential elements from the documentation and SDSP could express the minimal information of the AP, and could not express most of the components from the documentation. BibFrame Profile editor could express the full application profile, but semantics including the classes and its statements had to be represented as resources and statements. Also, specific hierarchical structures such as ‘mandatory,’ ‘recommended,’ and ‘optional’ status of the classification of the statements was skipped and represented the application profile structure more specific to BibFrame. YAMA is extensible due to its straight adaption of YAML, so every section from the application profile documentation could be included, and custom names could be used for the keys from the profile’s naming conventions.

In terms of the editing process involved, SDSP is found to be relatively easy as it is CSV, and a spreadsheet editor was used. However, BibFrame Profile Editor is a direct application profile editor with an interactive web-based user interface and advanced suggestion systems to help the profile creation process. Also, BibFrame profile editor supports selecting vocabularies for pre-populating the fields as well as it permits to edit the application profile in a well-structured

way. Both in SDSP and YAMA, the structure of the profile is logically built through the syntax specification, but in BibFrame editor the structure is built through visual interaction. Comparison of these authoring formats for YAMA use cases are given in Table 2.

Table 2. Comparison of general use cases for application profile authoring formats.

Use-cases	BIBFRAME	SDSP	YAMA
Actionable MAP authoring tool	Yes (Web UI)	Yes (CSV)	Yes (YAML)
Maintainable source for MAP	Yes (as JSON)	Yes (as CSV)	Yes (as YAML)
A structured textual format, for collaborative development and version control	Yes (JSON is structured and maintainable)	Limited (tabular format)	Yes (YAML is suitable with version control systems)
A structured authoring format makes it easy to validate the MAP and helps to eliminate errors and logical complexities	Yes (GUI Editor limits errors, and JSON can be validated)	No (schema validation is limited for CSV)	Yes (Any YAML schema validator can be used)
Change records as a timeline for changelog and metadata migrations	No	No	Yes (ChangSet format for actionable changelogs)
Generate human friendly documentation	No	No	Yes (with templates)
Template driven to customize the output formats	No (but JSON can be parsed for templating)	No (logical parsing is required)	Yes (optimized for templating)
Extensibility of the format to customize it to suit for different community/use-cases	No (limited to BibFrame)	Limited (not extensible)	Yes (extensible with custom key-values as YAML)
Inter-operable with editors/tools	Limited (output JSON can be edited as a text)	Limited (as spreadsheet)	Yes (YAML supports text editors, highlighting and indenting)
Adherence to Singapore Framework	No (Only basic application profile)	No (MetaBridge generates owl)	Partial (using templates & generators)

4 Discussion

The results suggest that authoring formats can significantly reduce the overall efforts in application profile creation as well as in compelling production of different expression formats. Three different types of authoring formats were evaluated and a considerably larger application profile is recreated using them. Compared to the other two constrained formats, as an extensible authoring tool for application profile, YAMA could express most of the documentation in an actionable manner. Limitations of the other two formats lead in dropping out most of the elements from the documentation and thus, permitting only a part of the human-readable documentation to be recreated. In YAMA, custom key-value pairs were used to include some aspects of the documentation, but in SDSP the structure is constrained so that it cannot be treated as an extensible option to add any custom information other than those specified in the SDSP structure. Similarly, the BIBFRAME profile editor does not permit to interact with the underlying structure and limits the user to follow a specific input process which the GUI is designed for. Sample files created as part of this evaluation are published at <http://purl.org/yama/examples>.

The limitation of this study is that the formats were only evaluated based on their ability to express existing documentation of an application profile. The results may vary in different scenarios, such as creating an application profile from scratch or modifying an existing authoring source. Even though YAMA is an efficient application profile authoring format, YAML syntax can pose a significant drawback. The structure may be challenging for editors, and they need to be comfortable in editing YAML in a text editor.

5 Conclusion

The developments in linked open data promote data exchange and interoperability of data among communities. Application profiles are the most suitable means for ensuring interoperability and for bringing in better use cases for data. Introducing easy to create and maintainable authoring formats will help different communities to adopt application profiles to express their data. There are evolving use cases for application profiles other than just explaining the metadata instance. Extensible authoring formats are expected to provide much more coverage for these emerging use cases, and it will also help communities to generate different formats or type of documentation from a single source which may serve much more purposes than just creating application profiles. The scope of an extensible format is futuristic, and also it can cater to various other demands related to metadata instances. Easy-to-use tools will also promote domain experts to create and publish application profiles in different machine-actionable expressions as well.



References

1. BIBFRAME Profile Editor (2018). <http://bibframe.org/profile-edit/>
2. The DCAT Application profile for data portals in Europe (DCAT-AP), April 2019. <https://github.com/SEMICEu/DCAT-AP>. Accessed 13 Sept 2017
3. Baca, M.: Introduction to Metadata, July 2016. <http://www.getty.edu/publications/intrometadata>
4. Ben-Kiki, O., Evans, C., döt Net, I.: YAML Ain't Markup Language (YAMLTM) Version 1.2, October 2009. <https://yaml.org/spec/1.2/spec.html>
5. Coyle, K.: RDF-AP, January 2017. <https://github.com/kcoyle/RDF-AP>. Accessed 12 Jan 2017
6. Enoksson, F.: DCMI: a MoinMoin Wiki syntax for description set profiles, October 2008. <http://www.dublincore.org/specifications/dublin-core/dsp-wiki-syntax/>
7. Heery, R., Patel, M.: Application profiles: mixing and matching metadata schemas. Ariadne (25) (2000). <http://www.ariadne.ac.uk/issue/25/app-profiles/>
8. Hillmann, D.: Metadata Standards and Applications. Metadata Management Associates LLC (2006). <http://managemetadata.com/>
9. Nagamori, M., Kanzaki, M., Torigoshi, N., Sugimoto, S.: Meta-Bridge: A Development of Metadata Information Infrastructure in Japan, p. 6 (2011)
10. Nilsson, M., Baker, T., Johnston, P.: DCMI: The Singapore Framework for Dublin Core Application Profiles, January 2008. <http://dublincore.org/specifications/dublin-core/singapore-framework/>
11. Nottingham, M., Bryan, P.: JavaScript Object Notation (JSON) Patch, April 2013. <https://tools.ietf.org/html/rfc6902>
12. Linked Data for Production 2 (LD4P2): Sinopia Profile Editor (2019). <https://profile-editor.sinopia.io/>
13. Thalhath, N., Nagamori, M., Sakaguchi, T.: YAMA: Yet Another Metadata Application Profile (2019). <https://purl.org/yama/spec/latest>

Digital Libraries and Digital Archives Management



Connecting Data for Digital Libraries: The Library, the Dictionary and the Corpus

Maciej Ogrodniczuk¹ and Włodzimierz Gruszczyński²

¹ Institute of Computer Science, Polish Academy of Sciences, Jana Kazimierza 5,
01-248 Warszawa, Poland

maciej.ogrodniczuk@ipipan.waw.pl

² Institute of Polish Language, Polish Academy of Sciences, al. Mickiewicza 31,
31-120 Kraków, Poland

wlodzimierz.gruszczyński@ijp.pan.pl

Abstract. The paper presents two experiments related to enhancing the content of a digital library with data from external repositories. The concept involves three related resources: a digital library of Middle Polish prints where items are stored in image form, the same items in textual form in a linguistically annotated corpus, and a dictionary of Middle Polish. The first experiment demonstrates how the results of automated OCR obtained with open source tools can be replaced with transcribed content from the corpus, enabling the user to search within individual prints. The second experiment links the print content with the electronic dictionary, filtering relevant entries with the dictionary of modern Polish to eliminate redundant results. Interconnecting all relevant resources in a digital library-centered platform creates new possibilities both for researchers involved in development of these resources as well as for scholars studying the Polish language of the 17th and 18th centuries.

Keywords: Digital library · Linguistic corpus · Electronic dictionary · Middle Polish

1 Introduction

Digital libraries are often isolated from external resources, even when the latter are thematically related and could serve as repositories of data enhancing the library items with additional information. In this paper we present the case study of such an enhancement, linking three electronic resources representing the same historical period into a common platform, facilitating analysis of historical data.

The work was financed by a research grant from the Polish Ministry of Science and Higher Education under the National Programme for the Development of Humanities for the years 2019–2023 (grant 11H 18 0413 86, grant funds received: 1,797,741 PLN).

© Springer Nature Switzerland AG 2019

A. Jatowt et al. (Eds.): ICADL 2019, LNCS 11853, pp. 125–138, 2019.

https://doi.org/10.1007/978-3-030-34058-2_13

The resources in question are: CBDU—The Digital Library of Polish and Poland-related Ephemeral Prints from the 16th, 17th and 18th Centuries, e-SXVII—The Electronic Dictionary of the 17th–18th Century Polish and KORBA—The Electronic Corpus of 17th and 18th Century Polish Texts.

All the above-mentioned resources were created independently, but due to their obvious inter-dependence they have already been used as data sources for one another. 40 digital library prints were transcribed into the corpus and corpus data served as a source of linguistic information and examples of word usage for the dictionary. Nevertheless, the management systems of these resources were not connected and all data relations had to be discovered manually. Only recently some extensions were added to the corpus search system to show entries corresponding to a given textual form and to the electronic dictionary to execute search for corpus examples documenting a given lemma [4], but the results were not linked to the source data in any way. At the same time, the digital library could benefit from referencing the dictionary entries to provide contemporary Polish explanations of words or phrases no longer in use. Last but not least, while the index of documents used in the corpus listed some basic metadata, much richer metadata was already available—but not used—in the digital library records for each given document.

The focus of this paper is providing the digital library with data coming from the two other resources: the dictionary and the corpus. After providing basing information on the resources used in the subsequent process (Sect. 2) and discussing existing approaches to the tasks relevant to our study (Sect. 3), we describe two experiments: using the transcribed data from the corpus to clean the output of automated OCR of library prints (Sect. 4) and linking print content with dictionary entries, filtered against a dictionary of contemporary Polish (Sect. 5). The concluding section summarizes our findings from the process and discusses the possibilities of its further improvement (Sect. 6).

2 Resources Used in the Study

Below is a brief description of the characteristics of the resources used in our experiment. Three of them are related to Middle Polish and one to contemporary Polish.

2.1 CBDU: The Digital Library of Polish and Poland-Related Ephemeral Prints from the 16th, 17th and 18th Centuries

The Digital Library of Polish and Poland-related Ephemeral Prints from the 16th, 17th and 18th Centuries (Pol. *Cyfrowa Biblioteka Druków Ulotnych polskich i Polski dotyczących z XVI, XVII i XVIII wieku*, hereafter abbreviated CBDU¹ [9, 14, 15] is a thematic digital library of approx. 2 000 Polish and Poland-related pre-press documents (ephemeral prints—short, disposable and irregular

¹ <https://cbdu.ijp.pan.pl/>.

informative publications) dated between 1501 and 1729 (all surviving documents of this kind described in scientific publications, particularly by Zawadzki [20]). From the moment of its creation, this digital library was used not only as a repository of digitized artefacts but also as a playground for experiments in rich description and linking of related data. At the time of this project, the CBDU management system, EPrints [1], was already used for extending item metadata with rich information provided by historians, philologists, librarians and computer scientists. This information included historical comments, glossaries of foreign interjections, explanations of lesser-known background details and relations between library objects (translations, adaptations, alterations of the base text, their alleged sources etc.) Still, existing resources relevant to the task (such as the dictionary, see Sect. 2.2, a source of explanations of Middle Polish terms difficult to understand for the modern reader) were only treated as a source of information, retrieved and duplicated rather than linked externally.

As for the content, CBDU prints are currently available only in image form, i.e. as PDF files containing scanned versions of original works². Even though the library creators intended to add transliterated or transcribed versions of those items, the task proved impossible within the project timeframe and only one sample text was transcribed³.

2.2 e-SXVII: The Electronic Dictionary of the 17th–18th Century Polish

The *Electronic Dictionary of the 17th–18th Century Polish* (Pol. *Elektroniczny słownik języka polskiego XVII i XVIII wieku*) [6, 7] is a continuation of the paper version of *The Dictionary of the Polish Language of the 17th and the first half of the 18th centuries*, with the first volume published between 1999 and 2004 [17]. In 2004 the decision to convert the paper dictionary to an electronic form was made and successive entries have since been made available on the website of the dictionary (<http://sxvii.pl>, henceforth e-SXVII; [8]). At present, the dictionary contains ca. 38k entries in various stages of development; few of the entries are complete, many of them are still being modified and supplemented (but contain semantic information), and most are still stubs (i.e. they only include grammatical forms of the lexeme and/or an example documenting use in text). There are ca. 76k grammatical forms among the entries. It must be noted that the inflectional paradigms cover only those forms which were found in the examined sources, and thus, in the majority of cases, the paradigms are not complete. The meanings of described lexemes are illustrated by quotations transliterated from over 1 000 sources.

² Content objects are currently available for over 1 400 prints and constitute to 11 585 pages.

³ See print 1264 at <http://cbdu.ijp.pan.pl/12640/>. The print also features a historical commentary and a manually created glossary explaining more difficult terms.

2.3 KORBA: The Electronic Corpus of 17th and 18th Century Polish Texts

The Electronic Corpus of the 17th and 18th Century Polish Texts (until 1772) (Pol. *Elektroniczny Korpus Tekstów Polskich z XVII i XVIII w. (do roku 1772)*) is also referred to as the *Baroque Corpus* (Pol. *Korpus Barokowy* – hence the acronym KORBA). Its data counts over 13M tokens (a relatively large number for any historical corpus). The aim of the corpus is to supplement the National Corpus of Polish [16] with a diachronic aspect, crucial for any investigation into the history of the Polish language. The corpus contains rich structural annotation (e.g., the location of the search phrase in the appropriate text page) and text annotation (e.g., tagging words from foreign languages). Part of the corpus (0.5M tokens) is manually annotated with morphosyntactic tags. This serves as a basis for automatic annotation of all data by a disambiguating tagger.

2.4 PoliMorf: A Morphological Dictionary of Modern Polish

PoliMorf [19]⁴ is currently the largest morphological dictionary of Polish, resulting from a merger of morphological data from two other prominent resources, SGJP—The Grammatical Dictionary of Polish [18]⁵ and Morfologik [13]⁶. The dictionary contains over 6.5M morphological interpretations and 3.8M different textual forms corresponding to 315 K lemmata. The large coverage of contemporary Polish results from the difference in lexicographical approaches to dictionary creation, each focused on different sets of word forms: while SGJP is the result of several years of work of linguists, Morfologik was developed by an online community of word game enthusiasts.

3 Related Work

The practical approach adopted in this work views digital libraries as institutions and services rather than collections of content [5] and as centers of effective integration of information resources *in order to provide readers with a systematic, complete, fast and accurate knowledge service* [21]. This view has already been present in CBDU [15], transformed from a conventional digital library into a research collaboration platform, featuring content augmented with rich metadata, gathered by the results of co-operation of researchers representing various fields.

Of particular interest to this study is the obvious relation of language corpora and dictionaries, most extensively analysed by Atkins and Rundell [2, p. 53 and further] and previously commented by many computational linguists, e.g. Kilgariff [12]. The most important aspect of this from the point of view of this work has already been tackled in the process of linking KORBA and e-SXVII [4].

⁴ <http://zil.ipipan.waw.pl/PoliMorf>, version 0.6.7.

⁵ <http://sgjp.pl/>.

⁶ <http://morfologik.blogspot.com/>.

The corpus users can already benefit from information about meanings, etymology, etc. of the displayed segments while dictionary users have gained an easy access to a much wider set of usage examples. This is also advantageous for the development of the resources: the integration allowed for identification of missing dictionary entries and simplification of usage examples for the existing ones, made finding a complete set of inflection forms possible and searching for multi-word expressions much easier.

A review of the subject literature shows that the problem of combining corpora with dictionaries is relatively frequent, especially the creation of tools for (semi) automatic data search in the corpus and their porting over to the dictionary. Such tools may be intended both for the editors of the dictionary, where they facilitate the process of entry creation (as e.g. with TshwaneLex Lexicography Software [11]⁷), or for the readers who may request and receive data from the corpus. This solution already functions in e-SXVII and is also used in large commercial publications⁸.

The integration of digital libraries (and particularly historical data repositories) with explanatory dictionaries is a much less popular search path. One of the goals of such integration may be the use of semantic explanations (definitions of meanings) contained in the dictionary to facilitate understanding of difficult, rare, or old words. This is somewhat similar to the problem of using bilingual dictionaries to aid the reader in understanding a foreign language text.

Part of the task involved in the project was to apply the OCR engine to historical data, so existing studies were consulted to select the best possible solution to this problem. A very recent comparison of two leading recognition engines, Tesseract⁹ and ABBYY FineReader¹⁰ is offered in the report [10] of the IMPACT project¹¹. The conclusion of that comparison found no single winner that would outperform the other program in all test cases, so the open source solution was selected for further tests. From the perspective of processing Polish historical data, the findings of the IMPACT project were also very important for the purpose of searching in the hidden layer of digital library items [3].

4 Experiment 1: From Dirty to Clean OCR with a linguistic Corpus

As mentioned before, CBDU prints were originally available only as graphical PDF files. The lack of textual layer made it impossible to use even the basic search capabilities within individual prints. Previous attempts at using automated OCR (so called *dirty OCR*, i.e. inaccurate optical character recognition) to add the textual layer to the prints were far from satisfactory and consequently abandoned by the library creators.

⁷ <https://tshwanedje.com/tshwanelex/>.

⁸ E.g. <http://www.macmillandictionaries.com/features/from-corpus-to-dictionary/>.

⁹ <https://github.com/tesseract-ocr/>.

¹⁰ <https://www.abbyy.com/finereader/>.

¹¹ <http://www.impact-project.eu/>.

When the KORBA corpus was created, several prints from CBDU were identified as representative sources of language data. Consequently, the prints of interest to the KORBA creators were manually transcribed and transliterated. The idea of the present experiment was to use this corpus data to create a high quality combination of image and text that would replace image PDFs and serve for further experiments on the textual layer, such as linking text fragments to dictionary entries.

4.1 Conversion Workflow

Presented below is the sequence of operations 38 prints (almost all of the prints transcribed and included in the corpus until the end of the first phase of the corpus creation project, i.e. until the end of April 2018¹²) were subjected to:

1. Obtaining a dirty OCR of a print

To retrieve the first approximation of OCR-ed text and, most importantly, the position of individual text regions within a print, we use OCRMYPDF¹³, a TESSERACT-based software for adding a textual layer to existing PDF files¹⁴.

2. Retrieval of corpus transcriptions

The transcriptions of corresponding prints are retrieved from the corpus. Even though KORBA contains rich TEI format for preservation of various characteristics of individual prints, only the transcription layer is used in the experiment. To facilitate further alignment, the texts are pre-processed to attach punctuation characters to preceding words.

3. Token alignment and fine-tuning

Beyond Compare¹⁵ scripts are used to automatically align the OCR tokens with tokenized texts retrieved from the corpus.

4. Token replacement and creation of a clean version of the print

CAM::PDF Perl library¹⁶ is used to replace the OCR results with clean data resulting from successful alignment.

4.2 Notes on OCR Quality

The quality of OCR (which could be assessed by comparing its results with the transliteration layer) was not fully investigated since the OCR engine was mostly used only to retrieve region-based information of the page layout. Still,

¹² For two prints out of the 40 transcribed ones the OCR process failed with segmentation fault.

¹³ <https://github.com/jbarlow83/OCRmyPDF>.

¹⁴ Note: all open source tools used in the process should be treated only as examples and similar results can be achieved with different software. For the OCR PDF conversion the authors of OCRMYPDF suggest several other open source programs for comparison such as PDF2PDFOCR or PDFSANDWICH; a commercial ABBYY FineReader suite is also frequently used for similar purposes.

¹⁵ <https://www.scootersoftware.com/>.

¹⁶ <https://metacpan.org/pod/CAM::PDF>.

some rough estimation was performed manually for a single page of one print¹⁷ to estimate whether the quality of OCR is sufficient enough to be used in the alignment process. Table 1 presents several human-evaluated (for information only) basic qualities of the process showing a token transfer accuracy of 85.7% in this sample.

Table 1. Statistics of OCR errors in a sample print (no statistical significance)

Quality	Token count
Actual number of textual tokens on the page	126
Number of tokens in dirty OCR	136
Number of properly OCR-ed tokens	22
Number of tokens in the corresponding corpus text	125
Number of correctly aligned tokens	124
Number of correctly transferred tokens	108

The number of tokens in dirty OCR is higher than the actual tokens on page since several words (e.g. ‘Kaznodziejskiego’) are broken at the end of a line (into ‘Ka-’ and ‘znodziejskiego’). In such cases we treat alignment as successful when one part of the dirty token (usually the longer one, depending on the alignment algorithm) is properly matched against the complete clean token.

Several tokens (22 in our example) were already properly recognized by the dirty OCR but, as expected, this number is low (17% of the total number of tokens on page).

4.3 Evaluation

The number of correctly transferred tokens was calculated by comparing the list of tokens in the input with the tokenized textual layer retrieved from the target PDF document (which forms the basis for searches for specific tokens in the document—the most relevant improvement from the point of view of the end user). This method of evaluation is by no means unquestionable since PDF is in general an output format, i.e. a map of locations of characters on the page (not even words) and the possibility of retrieving higher-level structural units is highly dependent on the specific rendering software. In this case the rendering

¹⁷ Page 3 of a print 1441: Relacja koronacji cudownego obrazu Najświętszej Marii Panny na Górze Różańcowej [w Podkamieniu] (*Report on the coronation of the miraculous image of the Virgin Mary on the Rosary Hill [in Podkamień]*; <https://cbdu.ijp.pan.pl/14410/>).

was produced by Adobe Acrobat Reader DC—the primary PDF reader, which approximates the environment available for an average user¹⁸).

Figure 1 presents the percentages of true tokens retrieved from the target PDF in Adobe Acrobat Reader¹⁹ and compared with the gold set using Beyond Compare scripts. The total number of tokens in the processed documents was 122 700. The average success rate of the conversion (the percentage of properly encoded tokens from the source) was 65.76% with the maximum reaching 86.89%.

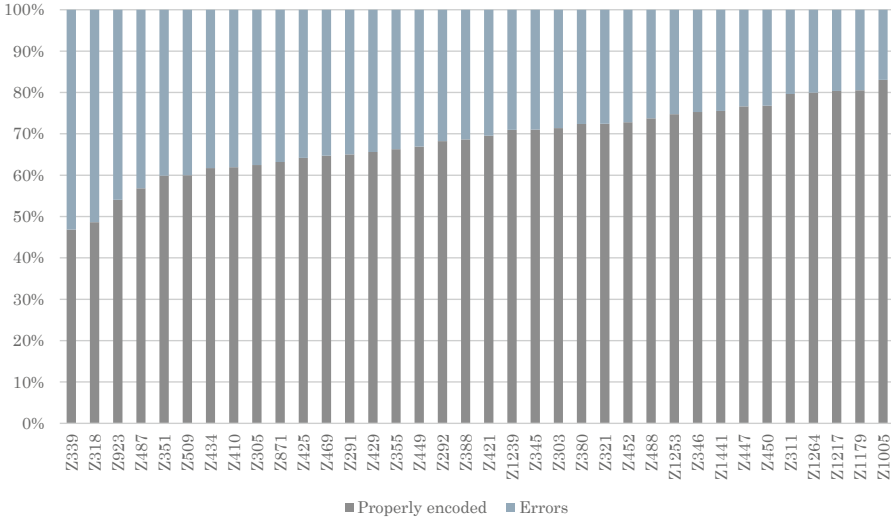


Fig. 1. Properly transferred tokens (dark) vs. encoding errors (light) for each text identified with the item number from Zawadzki’s bibliography [20]

Error analysis of the process showed that the primary reason for the difficulties in text replacement was the quality of the source scans, a direct consequence of the quality of their microfilm sources manufactured in the 1980s. In several cases, the transfer produced additional overlaps of regions—and, consequently, tokens—which might have been reflected in how the textual layer was rendered by the presentation software. One of the two prints with percentage of correctly transferred tokens lower than 50% (339) featured a two-column layout in part of the document, while the other (318) suffered from material damage (such as stains) and text bleeding through from the reverse page, which resulted in two and a half pages left completely unrecognized by the OCR engine. This finding

¹⁸ The most recent report available to the author estimates the market share for Adobe Reader 10 at 25% market share and Adobe Reader 11 at 55%, see <https://www.flexera.com/about-us/press-center/secunia-quarterly-country-report-pdf-readers-left-to-attacks-on-private-us-pcs.html>.

¹⁹ Version DC 19.012.20035.

goes in line with the IMPACT report [10] observation noting that Tesseract’s recognition accuracy can be decreased by noise.

4.4 Observations

The currently used algorithm calls for many functional improvements such as encoding line-broken words into two lines with soft hyphens to improve the search quality in the interface of the PDF reader. Currently such words are properly found in most cases, but only one part of the word is highlighted. In some cases (where the longer part of the hyphenated word is in the second line) we found out that our automatic alignment is responsible for overlap of regions, breaking the neighbouring tokens. Even where no soft hyphens are used, it might be more effective to align the clean token with the first part of the word (at the end of the line). Even if it would cause the token regions go beyond the end of the line, it would still provide correct location of the beginning of a given token.

5 Experiment 2: Annotating Prints with Electronic Dictionary

Annotating items with external information or linking to external resources from a digital library management system is straightforward, as long as one has programmatic access to its interface. This is rarely the case, but there are other reasons to not even try: even in the current version of CBDU, linking to an electronic dictionary from selected dictionary items stored in the metadata would seem awkward for users. It is much more natural to provide information directly in the text, as it often happens with ‘modern’ hyperlinked or annotated documents.

While manual linking from scanned prints to external resources is easy at the point of OCR correction (e.g. in a familiar FineReader interface), it requires an enormous amount of manual work. In contrast to this approach, the experimental method used here is completely automatic and takes into account several existing related resources mentioned in Sect. 2.

The main user-centered assumption of the work is that even though the dictionary can provide definitions for many words within the content, not every word in the text should be referenced because most of them are still perfectly understandable for the user (whether or not they are can be assessed by their presence or absence in contemporary dictionary of Polish). In our experiment we decided to filter out such words (even though the meaning could change over time) and exclude them from further processing. Presented below is the sequence of operations performed on prints processed in the previous experiment.

5.1 Filtering Out Modern Vocabulary

The list of entries in the e-SXVII dictionary which could provide relevant information for the user (i.e., are marked as complete or under construction, but still

contain information about meaning) is first filtered to remove lemmata absent from the KORBA corpus and then the ones present in PoliMorf, a dictionary of 20th-century Polish. The first analysis of the list of lemmata retained for further processing after the removal of the modern ones showed that some of them were included only due to different lemmatization methods in the dictionaries of old and modern Polish. For instance, PoliMorf and SGJP use verb infinitive as a lemma for gerunds while e-SXVII treats gerunds as separate entries and thus they could not be matched against the proper lemma. Based on this assumption the algorithm was improved for gerunds to include their nominative singular forms in the filter. Table 2 presents the statistics of the dictionary preparation to filter out irrelevant entries.

Table 2. Subsequent steps of dictionary filtering

Subset of the e-SXVII vocabulary	Number of entries
Meaningful dictionary entries	15 702
Lemmata present in the corpus	10 440
Lemmata absent from the XX-century dictionary	2 652
—with gerund correction	2 479

5.2 Creating Dictionary Annotations

Finally, the document content is compared to the dictionary list of entries to retrieve definitions and display them to the user directly in the text. The links are added with a custom Java program using iText7 library in its Community Edition²⁰.

In the current version of the algorithm, the definitions are included in the print in the form of PDF line annotations, presented as blue lines pointing at the beginning of the annotated word which display annotations retrieved from the dictionary when hovered over with the cursor. All annotations present in the document can be also displayed in the side panel of the PDF viewer (see Fig. 2).

5.3 Observations

Automated dictionary annotations produced 768 instances of textual forms linking to 164 distinct dictionary entries in all 38 prints. As noted before, in original version of CBDU only one print featured a manually created glossary of problem terms. Even with just this data, it was possible to compare it with the

²⁰ <https://itextpdf.com/en/products/itext-7/itext-7-community>, version 7.1.0. Again, there exist many commercial and non-commercial replacements for the software used, such as Evermap AutoBookmark Plug-in <https://www.evermap.com/autobookmark.asp> or Qoppa jPDFProcess Library <https://www.qoppa.com/pdfprocess/>.

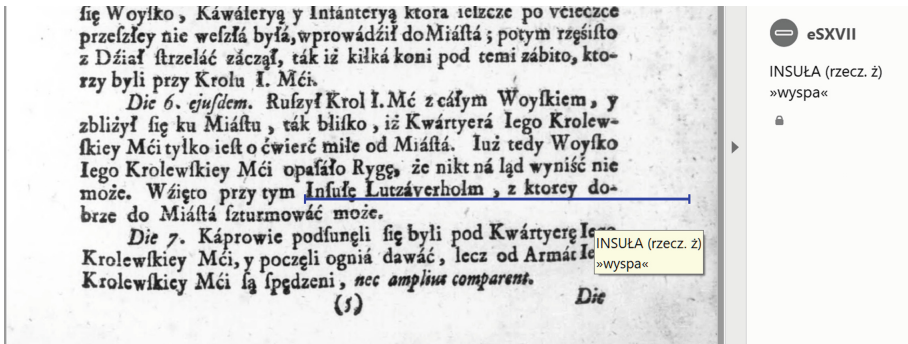


Fig. 2. Rendering of a PDF annotation with the dictionary definition in Acrobat Reader

result of automated filtering to see how it matches up with the editor's decision on which entries should be explained to the user, thus pointing towards areas where improvement is possible.

Out of all 29 entries available in the editor-created glossary for the print in question (1264):

- 14 entries are absent from e-SXVII, 5 entries are multi-word and 2 entries are stubs so they cannot be suggested by our mechanisms
- 3 entries are unexplained in the manual dictionary, probably because of their hapax legomenon status and not being clearly understood by the dictionary editor
- 4 entries can be found in the modern dictionary; one corresponds to an unused meaning (*bat*—a type of a boat), one to an understandable word (*oparkaniony*—surrounded by a fence) and two to outdated words (*ryszunek*—gear and *tentować*—to strive) not marked as such in the PoliMorf dictionary (but labeled as such e.g. in the recent update of the SGJP dictionary)
- one entry was present on the list of definitions suggested by our mechanism.

On the other hand, the list of suggested definitions contained 9 entries. Apart from the single entry constituting the intersection of both sets, all the remaining ones correspond to spelling variants of presently used words (*patrzeć*, *potym*, *wszedzie*, *wtym*) which explains why they were not of interest to the manual glossary. The general improvement to the process could apply some normalization to the list of dictionary entries before matching it against modern vocabulary to remove such alterations as diacritizations (*horyzont* vs. modern *horyzont*), consonant duplication (*ossobliwie* vs. *osobliwie*), a–o substitutions (*przelomac* vs. *przelamac*), denasalization (*cieżki* vs. *ciężki*). Some of the entries also familiar to the modern reader contain prefixes (*arcynos*) or non-stabilized spelling variants (*herbate* vs. *herbata*). Nevertheless, due to a large number of categories of such changes, resolving such issues would probably require tedious manual intervention and was not included in the current prototype.

6 Conclusions and Future Inquiries

The presented case study serves a number of purposes. Firstly, it can be used as a blueprint for future projects linking digital library data with external sources. The proposed workflow can be applied both to the entire collections of prints in digital libraries as well as individual items enhanced with links to data. Although the method uses PDF as a demonstration format, it can involve other formats provided they are capable of layered representation of data and storing linked information (such as DjVu). By establishing a link between the individual item and external resources, our technique is independent from any technical solution used for digital library management. Furthermore, the proposed procedure can serve as an example of co-operation between creators of linguistic resources, who are often unaware of opportunities such collaboration can provide to their users. Finally, it offers a general reflection on the central role digital libraries play in linguistic research, using a historic language as an example.

Procedures described in Sects. 4 and 5 can be applied to digital library data independently of each other and we believe that even then (e.g. by only linking dirty OCR to a dictionary, without cleaning the text first, as that can be infeasible for large projects) they can enhance user experience, particularly for foreign language, dialect or old language data. The proposed workflow would be useful in many configurations: both for processing individual prints and batches as well as for augmenting results of dirty and clean OCR. The tools used in the process can be replaced with different software.

Apart from the specific improvements already mentioned in the experimental sections, several general could be made regarding the use of our proposal in a digital library-centric collaboration platform. Similarly to how dictionary entries are linked to the corpus (providing examples of textual occurrences of a given lemma) and vice versa (providing useful definitions of words from query results), the corpus search interface could also be linked to prints available in the digital library, e.g. to show the whole page context of a given search result, helping corpus users locate the fragment directly in the source.

Last but not least, foreign-language prints and print fragments (in 11 languages) were not included by the process since e-SXVII dictionary only contains Polish entries. Dictionaries of other languages could be similarly integrated to provide translations of then-common Latin interjections and translations of foreign texts.

Acknowledgements. The authors would like to thank Grzegorz Kulesza for his diligent proofreading of this paper.

References

1. EPrints Manual (2010). http://wiki.eprints.org/w/EPrints_Manual
2. Atkins, B.T.S., Rundell, M.: The Oxford Guide to Practical Lexicography. Oxford University Press, Oxford (2008)

3. Bień, J.S.: Efficient search in hidden text of large DjVu documents. In: Bernardi, R., Chambers, S., Gottfried, B., Segond, F., Zaihrayeu, I. (eds.) AT4DL/NLP4DL-2009. LNCS, vol. 6699, pp. 1–14. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23160-5_1
4. Bilińska, J., Bronikowska, R., Gawłowicz, Z., Ogrodniczuk, M., Wieczorek, A., Żółtak, M.: Integration of the electronic dictionary of the 17th–18th century Polish and the electronic corpus of the 17th and 18th century Polish texts. Accepted for the Sixth Conference on Electronic Lexicography (eLex 2019) (2019)
5. Borgman, C.L.: What are digital libraries? Competing visions. *Inf. Process. Manag.* **35**(3), 227–243 (1999). [https://doi.org/10.1016/S0306-4573\(98\)00059-4](https://doi.org/10.1016/S0306-4573(98)00059-4)
6. Bronikowska, R., Gruszczyński, W., Ogrodniczuk, M., Woliński, M.: The use of electronic historical dictionary data in corpus design. *Stud. Polish Linguist.* **11**(2), 47–56 (2016). <https://doi.org/10.4467/23005920SPL.16.003.4818>
7. Gruszczyński, W. (ed.): *Elektroniczny słownik języka polskiego XVII i XVIII w.* (Electronic Dictionary of the 17th and the 18th century Polish, in Polish). Instytut Języka Polskiego PAN (2004). <https://sxvii.pl/>
8. Gruszczyński, W.: O przyszłości słownika języka polskiego XVII i 1. połowy XVIII wieku (On the future of the Polish dictionary of 17 and the first half of the 18th century, in Polish). *Poradnik Językowy* **7**, 48–61 (2005)
9. Gruszczyński, W., Ogrodniczuk, M.: Cyfrowa Biblioteka Druków Ulotnych Polskich i Polski dotyczących z XVI, XVII i XVIII w. w nauce i dydaktyce (Digital library of Poland-related old ephemeral prints in research and teaching, in Polish). In: *Materiały konferencji Polskie Biblioteki Cyfrowe 2010* (Proceedings of the Polish Digital Libraries 2010 Conference), Poznań, Poland, pp. 23–27 (2010)
10. Heliński, M., Kmiecik, M., Parkoła, T.: Report on the comparison of Tesseract and ABBYY FineReader OCR engines. Technical report, Poznań Supercomputing and Networking Center, Poznań (2012)
11. Joffe, D., MacLeod, M., de Schryver, G.M.: Software demonstration: the TshwaneLex electronic dictionary system. In: Elisenda Bernal, J.D. (ed.) *Proceedings of the Thirteenth EURALEX International Congress*, pp. 421–424. Institut Universitari de Linguística Aplicada, Universitat Pompeu Fabra, Barcelona (2008)
12. Kilgariff, A.: Putting the corpus into the dictionary. In: Ooi, V.B., Pakir, A., Talib, I.S., Tan, P.K. (eds.) *Perspectives in Lexicography: Asia and Beyond*, pp. 239–247. K Dictionaries (2009)
13. Miłkowski, M.: Developing an open-source, rule-based proofreading tool. *Softw. Pract. Exp.* **40**(7), 543–566 (2010). <https://doi.org/10.1002/spe.v40:7>
14. Ogrodniczuk, M., Gruszczyński, W.: Digital library of Poland-related old ephemeral prints: preserving multilingual cultural heritage. In: *Proceedings of the Workshop on Language Technologies for Digital Humanities and Cultural Heritage*, Hissar, Bulgaria, pp. 27–33 (2011). <http://www.aclweb.org/anthology/W11-4105>
15. Ogrodniczuk, M., Gruszczyński, W.: Digital library 2.0 – source of knowledge and research collaboration platform. In: Calzolari, N., et al. (eds.) *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, pp. 1649–1653. European Language Resources Association, Reykjavík (2014). http://www.lrec-conf.org/proceedings/lrec2014/pdf/14_Paper.pdf
16. Przepiórkowski, A., Bańko, M., Górski, R.L., Lewandowska-Tomaszczyk, B. (eds.): *Narodowy Korpus Języka Polskiego* (National Corpus of Polish, in Polish). Wydawnictwo Naukowe PWN, Warsaw (2012)
17. Siekierska, K. (ed.): *Słownik języka polskiego XVII i 1. połowy XVIII w.* (Dictionary of the 17th century and 1st half of the 18th century Polish, in Polish), vol. 1. Instytut Języka Polskiego PAN, Kraków (1999–2004)

18. Woliński, M., Kieraś, W.: The on-line version of Grammatical Dictionary of Polish. In: Calzolari, N., et al. (eds.) Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), pp. 2589–2594. European Language Resources Association, Portorož (2016). <http://www.lrec-conf.org/proceedings/lrec2016/pdf/1157.Paper.pdf>
19. Woliński, M., Miłkowski, M., Ogrodniczuk, M., Przepiórkowski, A., Szalkiewicz, L.: PoliMorf: a (not so) new open morphological dictionary for Polish. In: Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012, pp. 860–864. European Language Resources Association, Istanbul (2012). <http://www.lrec-conf.org/proceedings/lrec2012/pdf/263.Paper.pdf>
20. Zawadzki, K.: *Gazety ulotne polskie i Polski dotyczące z XVI, XVII i XVIII wieku* (Polish and Poland-related ephemeral prints from the 16th–18th centuries, in Polish). National Ossoliński Institute, Polish Academy of Sciences, Wrocław (1990)
21. Zhang, X.: Knowledge service and digital library: a roadmap for the future. In: Chen, Z., Chen, H., Miao, Q., Fu, Y., Fox, E., Lim, E. (eds.) ICADL 2004. LNCS, vol. 3334, pp. 104–114. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30544-6_11



Modernisation of a 12 Years Old Digital Archive: Experiences and Lessons Learned

Ilvio Bruder¹(✉), Martin Duffer², and Andreas Heuer³

¹ University Library Rostock, Rostock, Germany

² Hanse Softwarehaus, Rostock, Germany

³ Database Research Group, University of Rostock, Rostock, Germany
{ilvio.bruder,martin.dueffer,andreas.heuer}@uni-rostock.de

Abstract. Sustainability became a very important requirement in research projects, especially in digitalisation projects and their information systems. The problem in planning sustainability of data and systems is the unknown future of technical and organisational conditions for running these systems and services. In this paper, we want to look at a digital archive project about digitised historical music sheets which was built from 2003 till 2005 and generally modernised in 2017/2018. We want to present the challenges in rebuilding a 12 years old digital archive and discuss the lessons learned.

Keywords: Software evolution · Digital archive · Long-term preservation · Software monitoring

1 Motivation

Su-Shing Chen already described the suitable “Paradox of Digital Preservation” in 2001 [3]:

Traditionally, preserving things meant keeping them unchanged; however, our digital environment has fundamentally changed our concept of preservation requirements. If we hold on to digital information without modifications, accessing the information will become increasingly more difficult, if not impossible.

The digital preservation is also often described as the “Tamagotchi effect”. This means for a digital library or a digital archive project, that we have to maintain our system regularly. This requires a relatively big amount of resources, e.g. financial and personnel costs. Early digitalisation projects did not plan for later maintenance of their systems. Therefore, many funding authorities have extended their funding criteria to include sustainable solutions. Unfortunately, these criteria do not require, how this sustainability can be achieved practically.

Hence, preserving data as well as knowledge, e.g. described in [6], and preserving technical systems are different problems. Another problem is the economic sustainability which is explicitly addressed by digital libraries [7].

A good overview of different perceptions regarding sustainability in articles of recent years is given by [5]. Many aspects of sustainability were discussed, mostly technology, management, sharing and backups as well as costs and revenue. Many articles describe concepts which are important for planning digital archives. In this paper we want to discuss the real effects of weak maintenance of a digital archive running for 12 years.

The example we want to show, is a project of a digital archive of digitised handwritten music sheets. This archive is more a tool for analysing note scribes than a platform for sharing and exhibiting digital documents. Thus, the paper focuses on the problems migrating the functionality of the analysing tool on new hard- and software. The archive was built in 2003 and 2004 and had been constantly used over the years, but barely maintained. After over 12 years of usage, the system was very slow, crashed often, and the software included critical security risks. In 2017, we had to decide to either update this archive or shut down the service. We decided to rebuild the system and challenged many expectable and non-expectable issues. In the following sections, we want to give an overview of the old and the new system, of our problems rebuilding the system and of the lessons learned. The aspect of monitoring the system in the future will be discussed in particular.

2 Starting Point: The eNoteHistory Project

In 2002, about 1000 handwritten music sheets from the 17th and 18th century were digitised at the University of Rostock. After the digitalisation, a research archive for analysing such sheets was built in the eNoteHistory project [1, 2]. The project had the aim to build a digital archive, integrating knowledge components and specialised functions for writer identification in historical music scores.

Handwritten music scores were a way to record, copy and disseminate music during the late 17th century until the beginning of the 19th century. The information encoded in these music sheets, such as melody, title, time and place of origin, composer and scribe is of great interest to musicologists. Some of these data are seldom found in alphanumeric form in manuscripts, but have to be derived analysing other document features. Complex typographic and visual features such as handwriting characteristics, water marks etc. are used in practice for the analysis.

The archive consists of digitised sheets, metadata, as well as analysis tools and results. These data could be accessed in a web-based client at <http://www.enotehistory.de>. It was also possible to analyse new sheets with the help of feature trees or using an automatic analysis. In the manual analysis, specific features of the unknown music sheet are chosen from the feature trees, e.g. a specific handwriting of the treble clef.

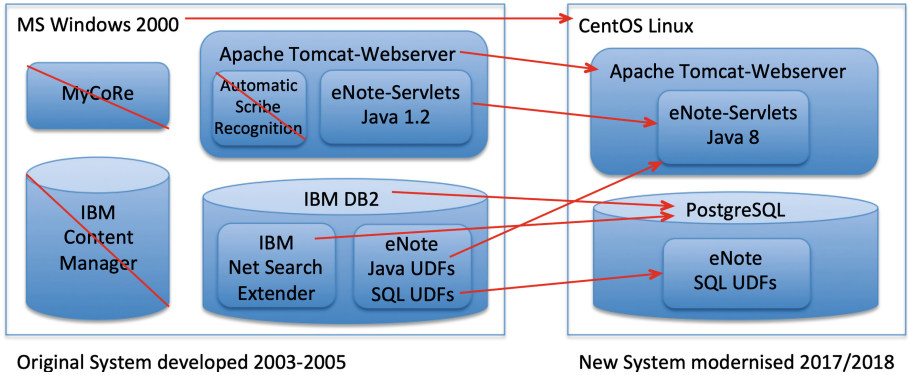


Fig. 1. Software components of the original and the modernised system.

The automatic analysis is based on the recognition of features like lines and their direction as well as the spatial arrangement of lines and points of the note heads and stems.

The system was built in a Windows 2000 Server environment on a separate server system. The programming language was mainly Java on server-side with Java Servlets for the web sites. There existed two storage solutions, first a realisation on an IBM DB2 database and second a realisation on an IBM Content Manager [4] installation with a digital library frontend based on MyCoRe¹. The DB2-realisation included the whole analysis and access possibilities. The MyCoRe-realisation only provided simple access and browsing functionality.

3 Working Steps for Modernisation

After over 10 years of minimum maintenance, the first step was an extensive and time-consuming analysis of the system because nobody of the original development team has been working at the institution anymore. One of the developers was contacted, but could not remember much of the design. In a second step, modernisation decisions had to be made about:

- what system (hard- and software) will be used,
- which components are left as old implementation, which components have to be reimplemented (see Fig. 1),
- which functions are essential, which functions could be switched inactive to reduce the effort, and
- how to migrate the data, metadata as well as user data.

In a third step, the new system was designed and implemented. After that, the data had to be transformed and migrated into the new system. The last step was the evaluation of the system.

¹ MyCoRe is a framework for building digital libraries: <http://www.mycore.de>.

3.1 Analysing the Old Implementation

The analysis was the most important step, before taking any implementation and modernisation decisions. The analysis consisted of examining hard- and software, testing the functionality as well as studying the original project and reading the documentations.

As mentioned above, the eNoteHistory archive was a Java implementation on a Windows 2000 Server. The last implementation and administration steps were done over 10 years ago. There were two different implementations: based on an IBM DB2 database (version 8) and based on IBM Content Manager with a MyCoRe frontend. The DB2-based version has been running the last 10 years. The Content Manager version was available, but was obvious fragmentary and could not be reactivated without massive effort.

The executable version using IBM DB2 database was somewhat broken, too. Functionalities like search, navigation, and analysis were not accessible. Moreover, the Windows 2000 Server was a virtual server and was working very unstable.

Documentation and source code were available in different versions and it was not clear which one described the last version. Furthermore, the source code was partly incomplete. The material was overall large, but inappropriately structured for long-time maintenance. After analysing all of the eNoteHistory code variants, we decided to use the IBM DB2 version as the initial point for the restoration. Contrary to IBM DB2, the IBM Content Manager is a database and information system software, which is not further available in this form at IBM.

The Windows 2000 Server is obviously not up to date and a potential security risk. It is important to update the underlying operating system before the restoration of the eNoteHistory software itself. The plan was to use a Linux-based operating system (CentOS) in the future for a better integration into the institutes hardware environment. This means, we had to check all software components of eNoteHistory for compatibility within a Linux environment. In several cases, it was necessary to find similar software or software libraries for the Linux operating system.

Besides the database system itself, there were used extensions of the DB2 database, such as the netSearchExtender which are not available for current versions of IBM DB2. Furthermore, the eNoteHistory software used self-implemented extensions, e.g., User Defined Types and User Defined Functions. These extensions are implemented in Java with version 1.2 and corresponding Java libraries. Such extensions are not executable in modern environments with Java version 8 or 11² and above.

The web site as a frontend was implemented using Java Servlets. These Java Servlets were executed and content was delivered by a Tomcat web server. The version of the Tomcat-Server was 5.5 with Java 1.2. This Tomcat-Server is not supported anymore. Some of the used functions in the Servlets are not available

² Java version 8 and 11 are long-term support (LTS) versions.

in current versions of Tomcat anymore. Therefore, we needed new functions and a migration or transformation of the old functions.

3.2 Upgrading the Database from DB2 to PostgreSQL

One major task in modernising the eNoteHistory system was the upgrade of the database system. Sustainability, expenses and spread of current database systems are important criteria for the decision, which database will be used in the future. We decided to use the open source database system PostgreSQL which is the university-wide deployed and supported database solution.

In a first step, the database schema, the special user-defined types, and user-defined functions were analysed. The SQL standard evolved over the years and the database manufacturer have often implemented special language adaptations of SQL. The schema had to be adapted, so that some DB2-specific data types were mapped onto PostgreSQL-specific data types. For instance, the DB2 data type DATALINK is not available in PostgreSQL. These changes were made before implementing the software. Some of the problems occurred and were solved during the implementation and transfer of the data. Such problems are sizes of data types, special formats or functional differences between the database systems. Due to the significant differences between the old and new database system, a simple export from the old system and import into the new system was not possible. The original database migration tools provided by IBM for DB2 version 8 were tested, but were not applicable. Therefore, we implemented an own migration tool for the schema and data transformation, which goes through the data step by step and repeats steps on failure. For instance, if an error, that a value is out of range of a domain, is produced, the step is undone, the schema is manually adapted, and the step is repeated. A failure can occur in every row of a table. The developed tool uses an old DB2 database driver in combination with a current database driver for PostgreSQL. The transformation of schema and data with this tool was a trial and error approach.

The eNoteHistory application and clients needed access to the database. The old DB2 driver was replaced by the new PostgreSQL driver in these implementations too. Furthermore, all of the SQL-queries in the implementations had to be checked, whether they were functioning as intended or they had to be transformed and adapted into a new SQL-query for the new system.

Particularly complex were user-defined functions in the DB2 system because they use special mechanics of this old database system. No user-defined function was applicable in current database systems due to the outdated and proprietary implementation. The solution was to analyse the source code of a particular function and reimplementing the function in the new database, reimplementing the function in the application or client outside of the database, or trying to use standard queries to achieve the same functionality. A specific problem at the eNoteHistory project was, that some functions did not work correctly anymore, which increased the complexity of the analysis significantly.

ENoteHistory used two kinds of user-defined functions: user-defined functions based on SQL and user-defined functions implemented in Java. User-defined

functions based on SQL had to be translated to the current SQL dialect of PostgreSQL and had to consider the made schema changes if needed. User-defined functions based on Java could not be transferred to PostgreSQL, firstly, due to the used outdated Java version and secondly, due to the lack of Java support in user-defined functions in PostgreSQL. Indeed, there is an open source add-on for Java user-defined functions in PostgreSQL, however, installation, configuration, as well as administration and updating is complicated and a significant source of failures and security issues. We decided to reimplement these Java user-defined functions within the application code of eNoteHistory outside of the database environment. This is not less complicated, but we have a better separation of database and the functionality which is more maintenance-friendly. The main advantage of user-defined functions, the performance of database-near implementation could be neglected regarding the average visitor numbers of the eNoteHistory service.

One of these Java user-defined functions was the calculation of the most similar scribes of a given music handwriting. The user-defined functions was reimplemented as a normal Java function within the eNoteHistory application (not as a user-defined function in the database). With this changes, such function was not more available within an SQL statement, but can be called within the program code. This means, that we needed new so called side tables for intermediate data, which were created automatically by the database before. Furthermore, we had to check all SQL statements, whether they used this user-defined function and possibly changed the SQL statement and the function calls.

3.3 Upgrading the Web Server and the Web Site

The original implementation of the eNoteHistory web service was based on Windows 2000, a DB2 V8 database, an Apache Tomcat Server v5.5 and Sun Java v1.2. The Windows 2000 server is replaced by a modern Linux distribution, CentOS in a current, long-term supported version. The DB2 V8 database system is replaced by PostgreSQL version 9.6 and Sun Java 1.2 by Oracle Java version 8. The Tomcat Server is used in version 8.5 with a current servlet specification.

All these updates and upgrades are significant modifications of the eNoteHistory application. Hence, many functionalities did not work anymore. Nearly every function had to be analysed and adapted, repaired or reimplemented step by step. For instance, the search functionality was implemented using the IBM DB2 Net Search Extender which is not supported anymore and was replaced by IBM DB2 Text Search. All programming code related to the Net Search Extender had to be found, analysed, and adapted to a search functionality, PostgreSQL as the new database system provides. Similar problems arose by using the current Java version due to missing some older, in eNoteHistory used code libraries. Some of these missing libraries could be replaced by current ones. Unfortunately, for few libraries, there were no current alternatives and we had to reimplement the functionality. We paid special attention to not use programs or software libraries from third party developers due to the uncertain sustainability.

Tomcat is used as the web server and contains the framework of the eNoteHistory application. The framework consist of Java components. The source code was available in different versions, but no version management was used. The code structure was barely described, but there were source code comments. The main problem was the first compilation of the source code after all these years. Unfortunately, it was not possible to compile the source code due to too many failures with outdated Java libraries and database interfaces. It took time and was difficult to eliminate all the failures getting a first compilation. Afterwards, reimplementation and corrections could be made.

It made sense to review every line of the source code. However, the given resources allowed only the most important and broken components to be analysed and possibly reimplemented. Further maintenance cycles may concentrate on other source code components.

3.4 Changes in Addition to the Old Application

During the rework of the eNoteHistory application, it made sense to overview other aspects of the system too. We have simply altered some textual information, corrected dead web links and provided new contact information.

Furthermore, we changed the procedure, how the files for the digitised notes are handled. In IBM DB2, the files are handled by the database using a DB2-specific functionality. In PostgreSQL, this particular functionality is not available, but a different one. Nevertheless, we decided to use a simpler way: handling the files by the operation system in the file system and using file locators as strings in PostgreSQL. This is not the recommended way, but it is less complex to maintain and is possibly better sustained. Moreover, the eNoteHistory server is dedicated only to this application and should not interfere with others.

Another aspect of the eNoteHistory application was the user management. The old user management required an authentication for some special search and analysis functionality. Nowadays, it is common to provide such functionality without any authentication. Only the altering of data should be possible for particular user. Therefore, we changed the user management significantly.

4 Lessons Learned and Measures

4.1 Lessons Learned

In the area of software evolution, there are many approaches, guidelines, information, and lessons learned about sustainability. However, most of them are from a software design (e.g. [11]) or an architectural (e.g. [8]) point of view. After finishing this work, we have learned few lessons, especially from a practical modernisation point of view:

Diversity of Failures: After running a public software over 14 years, you can find logical and content-based failures beside software errors, too.

Hard- and Software Provider: Using software, operation system, etc. you should pay attention to providers history (bigger and traditional companies are often more sustained) and to available long-term supported versions.

Standards: If possible, standards should be used, but not always the newest. For instance, SQL: no database manufacturer fulfils the newest version of SQL. Modernising eNoteHistory, we had to disable most of the newer SQL concepts like special user defined functions and the file handle concept. Sometimes, it makes sense to not extensively use available software libraries due to the dynamic and constant further development.

Maintenance Strategy: It is necessary, not only to maintain a system, but also to modernise it and maybe to reimplement some aspects from time to time.

Documentation: Not only the system needs to be documented, but also the software versions needs to be organised and described. This is also relevant for the technical organisation and technical information (which server, ports, file systems are used for what functionality).

Maintenance Plans: It is important to plan the maintenance for a minimum period of 5 to 10 years or expect severe failures after a couple of years.

Sustainability vs. Performance: From a technical point of view, the optimisation of a system regarding sustainability is more important than regarding performance.

Resources for Sustainability: Sustainability requires much resources. Projects which take into account sustainability and needed resources early, get a significant advantage. Resources like maintenance costs, hard- and software costs as well as working time and costs should be considered.

Maintenance Schedule: Hard- and software systems have to be updated regularly. A maintenance schedule depends on the planned system lifetime. Based on the experiences in eNoteHistory, the following maintenance schedule can be derived:

- system lifetime till 5 years: least effort, existing hard- and software should last the lifetime;
- lifetime 5–10 years: intermediate effort, the virtualisation of the hardware might be necessary, software and security issues might arise;
- lifetime more than 10 years: system has to be fundamentally updated (regularly, every 2–3 years on average). The longer the maintenance intervals are, the higher is the particular effort.

Monitoring the System: Moreover, the system has to be observed regularly. Failures and unavailability are bad for the user experience. It is even worse, if unavailability of services is unperceived by the operator. Nowadays, automatic tools are available for software monitoring.

4.2 Measures for Monitoring the System

Establishing a permanent observation of the eNoteHistory system was the first measure taken. Due to the software analysis of eNoteHistory, we were aware of the problems and requirements on an adequate monitoring. Besides, we had to notice that the system became more and more unstable and a permanent observation by a person was not possible. Service unavailability is caused by program failures, directory problems or missing program libraries (e.g., due to systems software updates) as well as failure of components (e.g., after a restart due to a power failure). Another problem is the reaction time until the service is available again.

Based on the experiences over the years, the following requirements emerge:

- automatic monitoring routines and reactions, as much as possible;
- periodic availability test of all web pages;
- periodic test of main functionality: login, search, and analysis steps;
- check of all web links;
- alerting per email or messenger post;
- reaction possibilities, e.g., set a maintenance status on the website or restarting the system.

The information whether the service is running or something is wrong, is very important for such a system. Therefore, we investigated in software systems for the monitoring of provided services. There are a couple of systems to be considered. Three systems are closely analysed: Selenium (<https://www.seleniumhq.org>), Cypress (<https://www.cypress.io>), and Scrapy (<https://scrapy.org>). Selenium and Cypress are tools explicitly for testing web sites. Scrapy is a web scraping tool for extracting data from web pages.

All of the three tools provide the functionality for checking web sites. Scrapy has rich concepts for getting data, information, and content out of a web page, but lacks features for testing and checking web sites periodically. Selenium and Cypress provide very similar, general functionality, but pursue different approaches. Selenium uses the interface of different browsers for accessing and interacting with web sites. Cypress uses a browser add-on which interacts directly with web pages. The disadvantage of Cypress is that it supports currently only the browser Chrome. The browser interface approach of Selenium is more flexible, but less powerful. Selenium is the older and technically more mature tool, which was the major reason to use it.

4.3 Website Tests Using Selenium

Selenium [9, 10] provides many different test procedures. It is primarily designed as an enhancement for software tests, especially web applications. Supporting periodic tests, Selenium is also appropriate for monitoring web sites. It uses so called WebDrivers to interact with browsers. Selenium supports several browsers and is suitable for cross browser testing. The architecture is based on a Client/Server approach. Hence, it is possible to manage and operate a couple of test

browsers by one Selenium host and one test script parallelly (e.g., for stress tests). Due to the exclusive usage of browser interfaces, Selenium can not directly handle events appearing within the browser.

The following tests are provided:

- Responsiveness and adaptability test regarding screen sizes, element sizes, and element visibility;
- Tests on elements: availability, visibility of elements; interference of elements, clickability; arrangement of elements (e.g. in a grid);
- Special media tests: size of images, correct scaling;
- Tests on interacting with elements, especially form elements: buttons, radio buttons, selection elements, input elements, result checking;
- Link tests: clickability, accessibility of links (URL checks);
- Cookie tests: check of stored data.

All of these tests and checks are described by test scripts, which can be implemented using a programming language. Selenium supports a couple of different programming languages. The analysis of the test results have to be explicitly implemented as well. Furthermore, Selenium provides different alerting and reaction possibilities, e.g., posting a message on twitter or altering the web server showing a maintenance site to users.

4.4 Selenium Example

Due to many different test possibilities of Selenium, we want to discuss one specific test example. Based on a series of clicks and checks, the manual analysis shall be tested. Selenium acts like a user by choosing features and checking the results. There are two reasonable test categories: First, specific features are given and the result is checked against the expected one. Second, features are randomly chosen by Selenium and the clickability as well as availability are checked.

Figure 2 shows the web page for analysing scribes of notes manually. The analysis consists of choosing properties of the notes to be analysed. On the left side, there is a tree representation of the upper three or four level of features. In the middle respective on the right side, the tree can be further explored and particular features can be chosen. After describing the available features, similar scribes can be searched. The picture shows the selection of a treble clef notation. The three upper levels of the feature tree is chosen on the left side (red circles in Fig. 2) and the other twelve levels to the final treble clef notation is chosen in the middle representation. The whole path is displayed above the choosing area (red rectangle). Besides the treble clef, the tilt of a quarter note and the note flag notation of an eighth note (visualised by a green square behind a specified feature in the tree) are specified.

In the test, a couple of features and the result is predefined. Selenium accesses the analysis web page and clicks on the topmost level of the first feature. The result is checked and should be unfolding the subtree of this element. The check also consists of the correct representation of the small feature images. After

The screenshot shows the eNoteHistory application interface. At the top, there is a logo for eNoteHistory with the tagline "Writer Identification in historical Music Scores". The user is logged in as "Guest" with the language set to "Lang.: [UK] [DE]". The main navigation bar includes "Home", "Browse", "Search", "Analyze", and "Login".

The central area is titled "Analyze a page manually." and contains a list of analysis options on the left and a detailed view of the selected option on the right. The selected option is "1. Schlüssel/ G-Schlüssel/ einzüiges Grundelement/ G2-Schlüssel/ rechtsläufig/ mit Einrollung/ innen/ aufwärts/ 1.1.1.1.2.1.1. . . 1. (gerade)/ 1.1.1.1.2.1.1. . . 1.2. (links)/ geschlossen/ 1.1.1.1.2.1.1. . . 1.2.2.1. (fallend)/ links am G-Punkt vorbei/ Kehre/ 1.1.1.1.2.1.1. . . 1.2.2.1.1.1. (rechts)". A red box highlights this text, and a red circle highlights the "1. Schlüssel" option in the left sidebar.

Below the analysis options, there is a section "Wert gespeichert:" with a musical staff showing a treble clef and a circled feature value: "1.1.1.1.1.2.1.1. . . 1.2.2.1.1.1.1. (rechts)". A "Zurück zur Auswahl" button is located below this section.

At the bottom of the interface, there is a table of search results:

FNode: 4.1.1 [50] -- 4.1.1.6.	FNode: 8.1 [69] -- 8.1.3.	FNode: 1.1.1 [0] -- 1.1.1.1.2.1.1. . . 1.2.2.1.1.1.1.	ND	Distanz	Schreiber	Werk
			0.125	0.0231	(Johann I)	Musica Saec. XVII.18.-51. ^39a
			0.125	0.1425	(Johann I)	Musica Saec. XVII.18.-45. ^9
			0.125	0.1969	(Johann I)	Musica Saec. XVIII.-61. ^7a

Fig. 2. Test the eNoteHistory manual analysis. (Color figure online)

unfolding all the levels of the regarding feature on the left side, the next feature levels are clicked in the middle area of the page until the given feature value is shown and chosen. Again, the result of every step is checked. If any given feature is looked up and chosen, the button “identify writer” is pressed by Selenium. Following, the results are compared with the expected. In case they differ, the administrator gets an e-mail with the precise error message.

5 Conclusions

The eNoteHistory application built from 2003 till 2005 was not maintained since then and has been modernised in 2017 and 2018. After 12 years, there existed many problems running such an old application. The problems were primarily

instabilities of the software, data and system security issues, and the increase of service failures and malfunctions.

The eNoteHistory system was based on Windows 2000, IBM DB2 v8, Sun Java 1.2, and Tomcat v5.5. After an analysis of the components, we decided to use a current Linux CentOS, PostgreSQL, Oracle Java, and Tomcat for the new eNoteHistory server. Thus, some fundamental, complex, and extensive changes were necessary.

Overall, the effort for modernisation of such a system after 12 years is tremendous. The question whether updating and upgrading or reimplementing is highly eligible and is difficult to decide. The eNoteHistory software system is running sufficiently stable and secure at the moment. An important task after the modernisation is a permanent and automatic monitoring of the services. We used the web software test suite Selenium for the monitoring and defined a couple of test procedures for checking the services, the correctness of the web pages, the web links, and the special analysis functions.

References

1. Bruder, I., Finger, A., Heuer, A., Ignatova, T.: Towards a digital document archive for historical handwritten music scores. In: Sembok, T.M.T., Zaman, H.B., Chen, H., Urs, S.R., Myaeng, S.-H. (eds.) ICADL 2003. LNCS, vol. 2911, pp. 411–414. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-24594-0_41
2. Bruder, I., Ignatova, T., Milewski, L.: Knowledge-based scribe recognition in historical music archives. In: Heery, R., Lyon, L. (eds.) ECDL 2004. LNCS, vol. 3232, pp. 304–316. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30230-8_28
3. Chen, S.: The paradox of digital preservation. *IEEE Comput.* **34**(3), 24–28 (2001)
4. Choy, D.M.: Integration of structured and unstructured data in IBM content manager. In: Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data. ACM (2005)
5. Eschenfelder, K.R., Shankar, K., Williams, R., Lanham, A., Salo, D., Zhang, M.: What are we talking about when we talk about sustainability of digital archives, repositories and libraries? In: Proceedings of the 79th ASIS&T Annual Meeting: Creating Knowledge, Enhancing Lives Through Information & Technology. American Society for Information Science (2016)
6. Giaretta, D.: *Advanced Digital Preservation*. Springer, Berlin (2011)
7. Hamilton, V.: Sustainability for digital libraries. *Library Rev.* **53**(8), 392–395 (2004)
8. Koziolok, H.: Sustainability evaluation of software architectures: a systematic review. In: Proceedings of the Joint ACM SIGSOFT Conference on QoSA and ACM SIGSOFT Symposium ISARCS. ACM (2011)
9. Sirotkin, A.: Web application testing with selenium. *Linux J.* **2010**(192) (2010)
10. Vila, E., Novakova, G., Todorova, D.: Automation testing framework for web applications with selenium webdriver: opportunities and threats. In: Proceedings of the International Conference on Advances in Image Processing, ICAIP 2017, pp. 144–150. ACM (2017)
11. Zdun, U., Capilla, R., Tran, H., Zimmermann, O.: Sustainable architectural design decisions. *IEEE Softw.* **30**(6), 46–53 (2013)



Investigating the Feasibility of Digital Repositories in Private Clouds

Mushashu Lumpa and Hussein Suleman^(✉) 

University of Cape Town, Cape Town, South Africa
mushashu@gmail.com, hussein@cs.uct.ac.za
<http://dl.cs.uct.ac.za/>

Abstract. Installing and configuring a digital repository toolkit for an organisation is a non-trivial task for which many organisations now seek external third-party service providers. Some of these service providers offer a cloud-hosted environment. However, universities increasingly have such cloud infrastructure in-house to support internal systems, and to maintain control and custody of data and systems. This study investigated the feasibility of using a private cloud internal to an organisation for the management of digital repositories. The results show that private cloud environments can run institutional repositories with negligible performance degradation as the number of virtual machine instances in the cloud are increased. A usability study of the prototype tool received positive feedback. Participants in the study were able to install and customise their own DSpace repositories.

Keywords: Private clouds · Digital repositories · Institutional repositories

1 Introduction

Cloud computing can be described it as a model of enabling on-demand network access to a shared pool of computing resources, which can be provisioned and released with minimal management effort [26]. Cloud computing has promised cost-effectiveness, flexibility and scalability for software and hardware requirements of organisations [19].

Digital repositories are software tools that are used to manage digital content, share it and provide the means for potential long-term preservation of that content. Digital repository tools are used extensively by libraries and institutions like universities that continually generate original content through scholarly publications and teaching materials.

If institutions can deploy digital repositories in cloud environments, they can leverage the benefits that cloud computing provides. Institutions could then devote more time to managing their digital content than their compute infrastructure and the software make-up of the digital repositories. And on the infrastructure end, administrative functions would benefit a lot from the internal management features of a cloud environment coupled with the efficient usage of their

computing resources. Institutions' digital repository content is always growing and, as such, infrastructures that scale and provide inherent elastic features are ideal to contain this content growth. Also, digital repositories are required to contribute to the preservation of their content, and running them in cloud infrastructures have a benefit of having their data potentially replicated in the variety of storage options that cloud environments provide.

For institutions to take advantage of the benefits of the cloud, it is important that there exist tools that simplify the deployment, management and monitoring of the digital repositories. One option is to adopt a service such as that provided by Bepress or Duraspace, where third party and public cloud infrastructure is used as the basis for digital repositories. Private clouds, based on equipment hosted completely within an organisation, are an alternative as institutions increasingly adopt this technology for local infrastructure; arguably, this could serve as the platform for the institution's own digital repositories as well.

This study therefore explores the feasibility of using private clouds for hosting of digital repositories. The key questions asked in this study focus on: the efficiency of the current generation of private cloud tools coupled with the current digital repository tools; and the feasibility of managing repository management at a high level on top of low-level private cloud tools.

2 Literature Review

2.1 Digital Repository Toolkits

There are different types of digital repository software toolkits available, and some of the commonly [6] used ones include DSpace [12,28], EPrints [15] and Fedora [23]. These repository tools share similar features, and to some extent are developed using the same technologies. A number of studies [1,6,21,31] have been done to compare the different features that each of these provide.

DSpace is an open source product, which is widely used and has an active developer community. It is developed in the Java programming language and runs off a PostgreSQL or Oracle database backend. Execution of Java requires the use of a Java Container and the commonly used one is Tomcat. DSpace installation [13] requires running commands that may require users to have technical skills [22]. Installation of DSpace is not trivial and often requires multiples tries and some amount of time to get it right.

Installation, configuration and customisation of EPrints and Fedora are not any less complicated compared to DSpace. They do not come with a guided installation graphical user interface [16,17]. They both require one to have some technical skills to run the commands that are shared in their installation documentation.

Like installation and configuration, open source repository tools require someone with some degree of familiarity with software development to be able to customise them. Familiarity with HTML and XLST are some of the skills required for a successful basic customisation of DSpace, for instance. Customisation of repositories is desirable as it allows institutions to brand their repositories to

adhere to their branding policies, while also improving the aesthetics of the repository to enhance its appeal. Verno [32] documented Boston Biomedical Consultants, Inc's experience in implementing a DSpace instance. As much as it was successful, the difficulties with the installation, configuration and customisation of DSpace were highlighted. These complexities need to be addressed in any cloud implementation.

2.2 IaaS and Eucalyptus

Infrastructure as a Service forms the basic layer of service delivery in cloud computing. Consumers/end-users are aware of their interactions with the lower level features of the computing infrastructure, they can decide which operating system to use, what and how applications can be installed, request to use more RAM, persistent storage, etc. All these features are provided to end users in a manner very different from traditional computing - end users have no need to directly interact with the physical machines, and can quickly change their preferences and effect them in a matter of seconds or minutes. However, to provide this abstraction, elasticity, machine orchestration, and on-demand features that cloud computing promises/provides, cloud computing software is used. They could be thought of as cloud computing operating systems. These software can be used to run either public clouds or private clouds. Examples of some of these cloud computing software platforms include: Amazon AWS [3], OpenStack [27], Eucalyptus [25], and OpenNebula [29].

Sharing of compute resources on servers to provide IaaS is achieved using virtualisation technology - virtual machines are created on the servers with user determined specifications. Cloud systems manage and monitor the creation of virtual machines using hypervisors [18,33]. Hypervisors are software programs that enable operating systems to host one or more virtual machines. Examples of hypervisors include Xen [7], KVM [8], Nitro [4], and vSphere [34].

For this study, Eucalyptus was used as the IaaS platform. Eucalyptus is an open source cloud computing operating system that can be utilised for building Infrastructure as a Service environments both for private and public cloud deployments [25]. Eucalyptus was developed to have interfaces that use similar commands as Amazon's very popular EC2 system. Eucalyptus comprises of 5 components that deliver a complete implementation of a cloud computing system. The components are: Cloud Controller (CLC), Cluster Controller (CC), Walrus Storage Controller (WS3), Storage Controller (SC) and Node Controllers (NC) [20].

2.3 Repositories in Clouds

Wu et al. described their work in migrating CiteSeerX into a private cloud environment [36]. Long terms costs, compared to migrating to a public cloud, were one of the reasons for their motivation to set up a private cloud infrastructure to host their digital repository. Their custom setup utilises proprietary software

- VMware ESXi - for the hypervisor and VMware VSphere for instance provisioning and general cloud orchestration and monitoring. Aljenaa et al. have explored the possibility of hosting e-learning systems in cloud based environments [2]. Their evaluation leads them to recommend hosting their systems in a private cloud environment. They discuss the on-demand and elasticity features of cloud computing environments as a major reason to recommend the use of cloud computing.

The Texas Digital Library described their efforts in first replicating their in-house computing infrastructure onto EC2, then completely migrating all their digital library services to Amazon's EC2 [24]. In the cloud, their services are provided on 48 virtual machine instances. Overall, they describe their move to the public cloud environment as a positive one. Their work demonstrates the successful implementation of a repository on virtualised, albeit public, infrastructure. Thakar et al. [30] described their experience migrating the Sloan Digital Sky Survey science archive, which has a 5TB database. They describe their experience as frustrating based on two reasons: degraded performance of the queries run off the database compared to their in-house infrastructure; and their inability to transfer the entire 5TB to the cloud. Their experience suggests that performance is a factor in assessing feasibility.

Others have looked into content preservation across cloud environments. DuraCloud [14], for instance, utilises different public cloud storage options to replicate content, providing the needed redundancy and potential data preservation. Digital repository tools can store content to DuraSpace through the different interfaces that DuraCloud provides. Kindura [35] is another project that is encouraged by the possibilities of content preservation using cloud storage systems. Both DuraCloud and Kindura are driven by long term content preservation needs and public cloud infrastructure, but suggest that repository management within a cloud can be feasible.

3 Methodology

3.1 System Development and Deployment

This study involved hosting digital repositories in a private cloud computing environment. The process involved simplifying the installation/configuration process of DSpace and automating its installation in a virtual machine instance.

A browser based application was developed to aid the installation process, customisation of DSpace and virtual machine management for ordinary repository end users. Installation was performed with a high degree of automation, and provided the necessary information required to identify and log on to the DSpace repository installed. The application also enabled end users to perform customisation of the DSpace, branding it with colors and logos according to an institution's branding policies.

A private cloud computing environment was setup using Eucalyptus, an Infrastructure as a Service (IaaS) software tool. Eucalyptus came bundled with the Ubuntu operating system and went by the alias of Ubuntu Enterprise Cloud

(UEC). Version 10.10 of UEC was used for this study. The institutional digital repository used was DSpace v1.8. Installation and configuration of DSpace was achieved through use of an orchestration and configuration management Python API called Fabric. The repository management tool was developed using the Django Web framework [11]. Django Celery [10] formed a core part of the overall solution: it enabled execution of tasks (e.g. installation and customisation of repositories) in an asynchronous manner.

3.2 User Front End

The process flow begins with a user logging onto the prototype application that was developed called *Lilu*, using a username and password. Figure 1 shows what is presented to a user or system administrator once they have logged in, as well as the functions that can be carried out on existing installations. At this point, the user/system administrator are able to carry out a DSpace installation or manage already existing repositories.

Repository Name	Admin - Email	View Repository	Activity	Activity - Status	Actions
Experiments Institute	dicloud@gmail.com	pending...	installing...	<div style="width: 10%; height: 10px; background-color: blue;"></div>	Action
Experiments Institute	dicloud@gmail.com	pending...	installing...	<div style="width: 20%; height: 10px; background-color: blue;"></div>	Customise
Experiments Institute	dicloud@gmail.com	pending...	installing...	<div style="width: 30%; height: 10px; background-color: blue;"></div>	Delete
Experiments Institute	dicloud@gmail.com	pending...	installing...	<div style="width: 40%; height: 10px; background-color: blue;"></div>	Shutdown
Experiments Institute	dicloud@gmail.com	pending...	installing...	<div style="width: 50%; height: 10px; background-color: blue;"></div>	Restart
Experiments Institute	dicloud@gmail.com	pending...	installing...	<div style="width: 60%; height: 10px; background-color: blue;"></div>	Start

Fig. 1. Features on a repository once it is installing or it has fully installed

Installation of a new DSpace instance starts with providing information that should be associated with a given repository - this is repository-identifying information and system administrator credentials. The needed information is provided via a Web form. Installation of the repository takes over 10 min to complete. However, the browser does not block until the installation has completed. Control is returned to the user, who can continue to perform other tasks provided by the prototype. Progress status of the installation is given - a successfully completed installation will have a green tick and its progress status information is updated accordingly.

While the installation is progressing or has completed successfully, basic customisation can be performed on the given repository or any repository associated with the currently logged in user. Figure 1 shows how to access the customisation function. The customisation feature allows for making minor modifications to the repository's overall look and feel. Positioning of the repository's logo can be switched between left and right, custom images for logos can be used, colors on the site can be changed, and the name of the repository and text on the body of the repository can be adjusted. Figure 2 shows the customisation page, showing the different parts of the repository that can be customised.

Other features available include shutting down the repository, restarting the repository and completely deleting the repository.

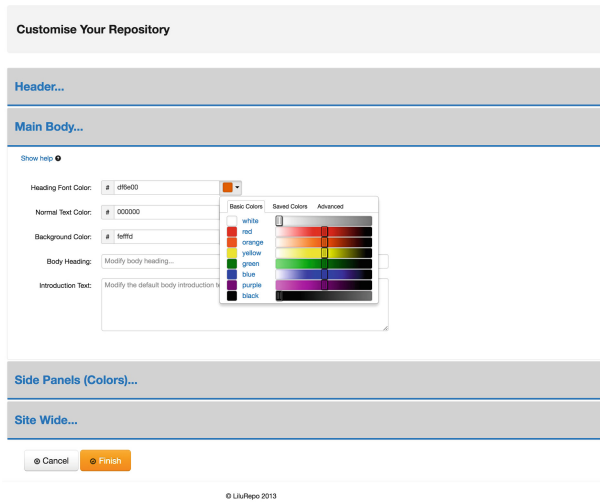


Fig. 2. Repository customisation page

3.3 Backend

The application on the backend manages the workflows that are needed to complete the requests made from the frontend. It does so by interacting with Eucalyptus cloud's APIs via *euca2ools*¹ [20]. *euca2ools* provides an abstraction and a set of programmable functions that enable administrative management of Eucalyptus cloud services. Some of the *euca2ools* functions utilised in this experimental solution are described below:

1. **Start (run) virtual machine instances** - this is achieved by executing the *euca-run-instance* from the commandline or *run_instance* python function of the *euca2ools* API. This function essentially boots the instance, bringing to life the virtual machine instance. Depending on the image used to run the instance, the virtual machine may or may not have an operating system in it. For this study, the image that was used had the Ubuntu operating system. Each instance that is run will have a system generated ID by Eucalyptus.
2. **Terminate virtual machine instances** - the commandline function associated with this function is *euca-terminate-instances*, with the equivalent python API call being *terminate_instances*. This call destructs the running virtual machine that is being terminated. It is the equivalent of shutting down

¹ *euca2ools*, <https://wiki.debian.org/euca2ools>.

and powering off a computer. All computing resources i.e. RAM, Volumes, CPU etc., will be freed up and ready for use by another virtual machine instance.

3. **Reboot virtual machine instances** - rebooting a virtual machine maintains all the instance's information and its connected peripheral devices. The command for this function is *euca-reboot-instances*. This call is used when the virtual instance needs to maintain its state of its connected peripheral devices and all other data saved on its ephemeral volume.
4. **Create and delete block storage volumes** - block storage volumes are what are used to persist data for virtual machine instances. The call to create volumes is *euca-create-volume* while the one used to delete the volume is *euca-delete-volume*. A given volume can only be deleted when it is not attached to a running virtual machine instance.
5. **Attach block storage volumes to instances** - this will attach the created block storage volume to a running instance. The *euca2ools* function called to attach volume(s) is *euca-attach-volume*. This function only ends at attaching the volume to the running instance. For the volume to be usable, it would need to be mounted by the operating system of the virtual machine and also formatted - formatting of the volume is only done once. Subsequent attachment of the same volume to instances requires no formatting unless it is the user's preference to do so.
6. **Check the status of virtual machines and volumes** - *euca-describe-volumes* and *euca-describe-instances* will check the status of virtual machine instances and available volumes in the cloud, respectively. Virtual machines will be in either of 2 states: running or terminating. Once terminated, the virtual machine ceases to exist. Volumes too will be in one of 3 states: deleting, available, or attached. The available state indicates that the volume is ready for use by an instance.
7. **Monitoring of the clouds' resource utilisation** - *euca-describe-availability-zones* is one of the important functions that helps ascertain the amount of resources that are in use against the capacity of the cloud environment. The function can be used to determine whether the cloud still has enough resources to run another virtual machine instance.

The application builds on these functions to provide the features available on the user frontend. The backend application will receive the requests, namely (1) *install DSpace*, (2) *customise DSpace*, (3) *shutdown DSpace*, and (4) *start DSpace*.

Install DSpace (install new repository) is the primary function in the developed application prototype. When the request to install DSpace is received, the backend application goes through the following phases:

1. Via the cloud's API, boot a new virtual machine instance.
2. If the instance booted successfully, the application will check if there is any available free block storage volume. If not, a new volume will be created. The successfully booted instance will have a private IP address.

3. The volume in step 2 will then be attached to the booted instance. The application will generate a unique identifier for this volume. This identifier will also be associated with the user requesting this installation.
4. The attached volume will be formatted and prepared for use in the virtual machine instance.
5. Using Fabric, DSpace and its dependency libraries will be installed on the virtual machine instance.
6. Configuration will be done for the DSpace PostgreSQL database and the location where the DSpace bitstreams should be saved. All data that should be persisted will be saved on the attached volume.
7. Restart Tomcat server on the virtual machine instance.
8. Assign this instance a URL and register it on the primary front-facing Apache Web server. This URL will be used to access the DSpace instance in the cloud. The URL will be mapped to the instance's private IP address in the cloud environment. Note that the DSpace instance in the cloud is using Apache Tomcat as its Web server and Java Container. The rerouting of external calls via the front-facing Apache server to the Tomcat server is achieved by using an Apache module called *mod_alias*² - this is installed on the front-facing Apache server, where the publicly accessible URL is mapped to the internal private IP address for the virtual machine instance.

Customise DSpace. When the backend receives this request to customise a given repository, it keeps a copy of the customisations to be made before publishing them to the DSpace instance. The customisation process entails overwriting files on the target DSpace instance. Overwriting of files is achieved through *rsync*³ calls. Using Django Celery, the user can perform the customisation while the DSpace instance installation is ongoing.

Shutdown DSpace. When this request is received on the server, the application will detach the volume from the instance before terminating the virtual machine instance. Detaching of the volume is only carried out once the PostgreSQL database and Web servers (i.e. Tomcat and Apache) have been stopped successfully. Termination of the virtual machine instance is via calls to the Eucalyptus API *terminate-instance*. This API call results in the equivalent process of shutting down the machine's operating system and powering off the virtual machine. Termination of an instance frees up the compute resources that can be used for other purposes. Note that this is transparent to the front-end user, who will have the perception of a traditional computer shutdown.

Start DSpace (or Restart DSpace). This action can only be executed in the event that the DSpace instance was previously shutdown. Starting a DSpace instance follows the same process as installing a new DSpace instance. The difference is that there is no DSpace installation and configuration required. A start DSpace task reattaches the volume that was detached during the shutdown. This

² *mod_alias*, https://httpd.apache.org/docs/2.4/mod/mod_alias.html.

³ *rsync*, <https://linux.die.net/man/1/rsync>.

volume will still have all the information about the DSpace instance before it was shutdown.

3.4 Evaluation

Two types of evaluations were carried out: (1) performance measurement to assess impact of virtual environment on typical activities; and (2) usability study of the developed browser-based application for installation and customisation of DSpace.

Performance Experiment. The objective of the performance experiment was to measure ingestion and viewing/downloading of items in DSpace and ascertain if there is any effect on performance when the number of instances in the cloud are increased or when concurrent requests are made to different instances.

The hardware and specifications used for this test are listed in Table 1.

Table 1. Hardware specifications used in performance experiment

	Main server	Nodes	Virtual machines	Client laptop
CPU	3.2 GHz Intel Core i5 (4 cores)		3.2 GHz QEMU Virtual CPU (2 cores)	2.5 GHz Intel Core i7 (4 cores)
RAM	8 G		1 GB	16 GB
Hard Disk	300 GB		5 GB	500 GB

Apache JMeter⁴ was used to simulate repository user actions and also take response time measurements.

The experiment began by running a number of virtual machine instances in the cloud environment. Once the virtual machine instances had fully booted, DSpace was installed in each one of the instances, following the steps outlined previously. Using JMeter, 15 items were ingested into DSpace installed in each of the running virtual machine instances. Item ingestions in a single DSpace instance were carried out sequentially, while ingestions in all the other running instances in the cloud were run in parallel. The length of time to ingest an item was measured, which in this study is called ingestion time.

Once the 15 items in each DSpace instance were successfully ingested, JMeter was used to view (load) the item. This was to mimic the process of viewing and downloading items with their associated attachments/documents. This step was used to measure the performance of accessing items in a DSpace instance running in a cloud environment. The process of booting virtual machine instances, installing DSpace in the instances, ingesting and viewing all 15 items in each of the running DSpace instances constituted a single run in the experiment. Each run consisted of one or more instances running in parallel.

⁴ <https://jmeter.apache.org/>.

Each run had a different number of instances running in the cloud - 1, 2, 5, 8 and 11 - and every run had 5 replications to establish a stable average measurement.

Figure 3 has a detailed breakdown of the average ingestion times by instance and also ingestion order of the item.

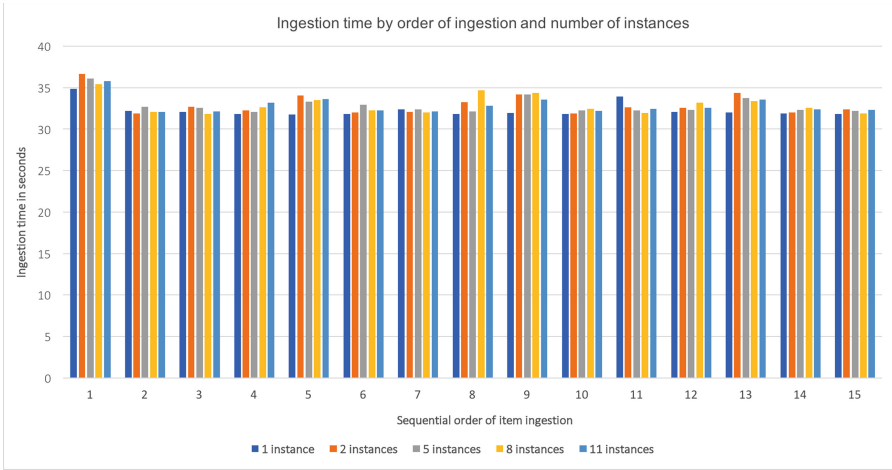


Fig. 3. Overall average ingestion time by order of ingestion and number of instances

It can be deduced from the data that an increase in the number of instances does not have a considerable effect on the performance of the ingestion of an item in DSpace. The overall average ingestion time remains between 30 and 35 s. The average ingestion time by 1, 2, 5, 8 and 11 instances are 32.27, 32.98, 32.86, 32.93 and 32.86 s respectively.

The initial ingestion time into DSpace is marginally higher than the rest of the ingestions. This can be attributed to Apache server on the central cloud server making its first connection with the Tomcat server on the DSpace instance in the cloud. In addition, at first run, Tomcat on the DSpace instance will be loading the necessary resources, which may contribute to this slowness. Once connections have been established and other configurations cached by Apache on the main server and the Tomcat server has been loaded fully, subsequent requests are noticeably faster.

The average view and download response times are shown in Fig. 4.

Overall, the average response times for the two tasks - item view and item download - are all under one eighth of a second. However, there is a noticeable linear decrease in performance as the number of instances running in the cloud are incrementally increased to full capacity. This difference would be hardly felt by users of the system. Using the ANOVA test to compare the differences between the view times of 1, 2, 5, 8 and 11 instances, an F-statistic of 72.57

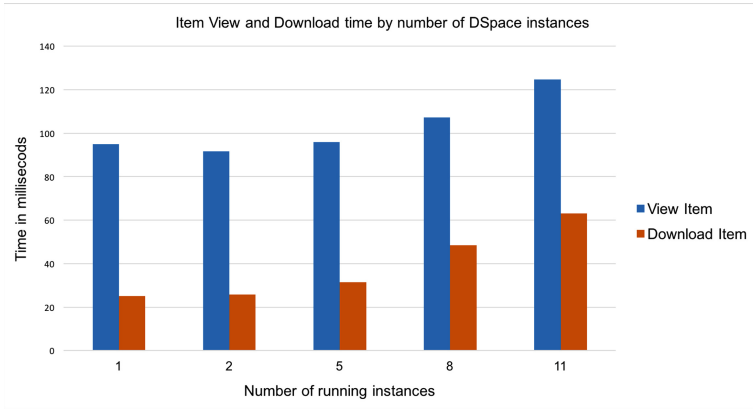


Fig. 4. Overall average item view and download time by number of instances

and p-value of 2.23 is obtained. Since $p > 0.05$, it cannot be said that there is a significant difference in the view times experienced.

Usability Evaluation. The System Usability Scale (SUS) [9] was adopted for this study. SUS is widely used⁵, has respectable reliability [5] and is available for free use without a licence. Users were also asked to comment on satisfaction and provide any additional comments.

The evaluation exercise involved completion of 2 main tasks that were the core features of the developed prototype: installation and customisation of a digital repository.

Twenty two (22) postgraduate student participants were recruited for this study. The minimum requirements for participants were that they should be familiar with Web technologies and be everyday users of the Internet. Based on prior experience with installation of complex software applications, participants were identified to fall into 3 categories: non-expert, intermediate and expert.

The overall average SUS score from the evaluation was 74. SUS is scored out of 100, and the global average from the SUS project is stated as 68. There was an observed difference in the average scores by each of the categories that were devised: experts scored an average of 81, intermediates scored 65 and non-experts scored 70.

Responses to the question, *What did you like about the application?* and *What did you NOT like about the application?* were grouped and analysed by common themes. There was a total of 39 unique positive comments and 27 unique negative comments. “Ease of use”, “ease of learning” and “a clean interface” were the most mentioned positive comments. “Installation time”, “responsiveness”, “system help” and “minor bugs” were the most mentioned negative comments.

⁵ Measuring Usability with the System Usability Scale, <https://www.measuringusability.com/sus.php>.

The time taken to complete the installation is a constraint of the cloud environment's response times. A number of techniques can be applied to ensure that users do not have to wait for over 8 min to start administering their repositories. One option would be to create instances in advance - once they are allocated to users, their only task would be to configure the system to their preferences.

The results from this evaluation indicate that users are able to complete tasks of installation and customisation of the system, and there is an above average satisfaction in interacting with repositories in the cloud through use of the developed system.

4 Conclusions and Future Work

This study set out to investigate the feasibility of hosting a digital repository system in a managed way in a private cloud environment. The system that was developed was functional, usable and had an acceptable level of system performance. Thus, a private cloud is indeed a feasible option for such software systems. Current interest in containerisation of software tools, through mechanisms such as Docker, are highly compatible with cloud management systems, as they each address a different aspect of the ease of management equation.

During the course of this research, no substantive changes were made to the DSpace repository toolkit. Thus, the move to a managed repository ecosystem, whether in a private cloud or elsewhere, can likely treat existing tools as black boxes to be encapsulated and managed at a higher level.

This form of turnkey management will ease the burden on administrators, many of whom are repository managers rather than systems administrators, and enable the adoption of repositories more easily than is the case at present.

Future work can investigate hybrid models that include replication and scalability, as well as managed containerisation of future back-end systems.

Acknowledgements. This research was partially funded by the National Research Foundation of South Africa (Grant numbers: 85470 and 105862) and University of Cape Town. The authors acknowledge that opinions, findings and conclusions or recommendations expressed in this publication are that of the authors, and that the NRF accepts no liability whatsoever in this regard.

References

1. Adewumi, A.O., Omoregbe, N.A.: Institutional repositories: features, architecture, design and implementation technologies. *J. Comput.* **2**(8) (2011)
2. Aljena, E., Al-Anzi, F.S., Alshayji, M.: Towards an efficient e-learning system based on cloud computing. In: *Proceedings of the Second Kuwait Conference on e-Services and e-Systems, KCESS 2011*, pp. 13:1–13:7. ACM, New York (2011). <https://doi.org/10.1145/2107556.2107569>
3. Amazon: Overview of Amazon Web Services (2018). <https://docs.aws.amazon.com/aws-technical-content/latest/aws-overview/introduction.html>



4. Amazon Web Services: Amazon EC2 FAQs - nitro hypervisor (2018). <https://aws.amazon.com/ec2/faqs/#compute-optimized>
5. Bangor, A., Kortum, P.T., Miller, J.T.: An empirical evaluation of the system usability scale. *Int. J. Hum.-Comput. Interact.* **24**(6), 574–594 (2008)
6. Bankier, J., Gleason, G.: Institutional Repository software comparison, vol. 33. United Nations Educational, Scientific and Cultural Organization (2014). http://works.bepress.com/jean_gabriel_bankier/22/
7. Barham, P., et al.: Xen and the art of virtualization. In: *ACM SIGOPS Operating Systems Review*, vol. 37, pp. 164–177. ACM (2003)
8. Bellard, F.: QEMU, a fast and portable dynamic translator. In: *USENIX Annual Technical Conference, FREENIX Track*, vol. 41, p. 46 (2005)
9. Brooke, J.: SUS-A quick and dirty usability scale. *Usability Eval. Ind.* **189**, 194 (1996)
10. Celery Project: Celery - distributed task queue (2019). <http://docs.celeryproject.org/en/latest/index.html>
11. Django Foundation: Django web framework (2019). <https://www.djangoproject.com/start/overview/>
12. DSpace: DSpace - A Turnkey institutional repository application (2018). <https://duraspace.org/dspace/>
13. DSpace: Installing DSpace - DSpace 6.x documentation - DuraSpace wiki (2018). <https://wiki.duraspace.org/display/DSDOC6x/Installing+DSpace>
14. DuraSpace: DuraCloud Guide (2019). <https://wiki.duraspace.org/display/DURACLOUDDOC/DuraCloud+Guide#DuraCloudGuide-WhatisDuraCloud?>
15. EPrints: EPrints Services (2018). <http://www.eprints.org/uk/>
16. EPrints: Installing EPrints on Debian/Ubuntu - EPrints Documentation (2018). https://wiki.eprints.org/w/Installing_EPrints_on_Debian/Ubuntu
17. Fedora: Installation and Configuration - Fedora 3.8 Documentation - DuraSpace Wiki (2016). <https://wiki.duraspace.org/display/FEDORA38/Installation+and+Configuration>
18. Freet, D., Agrawal, R., Walker, J.J., Badr, Y.: Open source cloud management platforms and hypervisor technologies: a review and comparison. In: *SoutheastCon 2016*, pp. 1–8. IEEE, March 2016. <https://doi.org/10.1109/SECON.2016.7506698>. <http://ieeexplore.ieee.org/document/7506698/>
19. Han, Y.: On the clouds: a new way of computing. *Inf. Technol. Librar.* **29**(2), 87–92 (2010). <https://search.proquest.com/docview/325033464/fulltextPDF/A52FC66FB8DE4D9BPQ/1?accountid=14500>
20. Johnson, D., Kiran, M., Murthy, R., Suseendran, R., Yogesh, G.: Eucalyptus beginner 's guide UEC edition eucalyptus beginner 's guide - UEC edition. Eucalyptus, v2.0 edn. (2010). <http://cssoss.files.wordpress.com/2010/12/eucabookv2-0.pdf>
21. Kőkörčený, M., Bodnárová, A.: Comparison of digital libraries systems. In: *Proceedings of the 9th WSEAS International Conference on Data Networks, Communications, Computers, DNCOCO 2010*, pp. 97–100. World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, USA (2010). <http://dl.acm.org/citation.cfm?id=1948805.1948823>
22. Körber, N., Suleman, H.: Usability of digital repository software: a study of dspace installation and configuration. In: *Buchanan, G., Masoodian, M., Cunningham, S.J. (eds.) ICADL 2008. LNCS*, vol. 5362, pp. 31–40. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-89533-6_4

23. Lagoze, C., Payette, S., Shin, E., Wilper, C.: Fedora: an architecture for complex objects and their relationships. *Int. J. Digit. Libr.* **6**(2), 124–138 (2006). <https://doi.org/10.1007/s00799-005-0130-3>. <http://link.springer.com/10.1007/s00799-005-0130-3>
24. Nuernberg, P., Leggett, J., McFarland, M.: Cloud as infrastructure at the Texas digital library. *J. Digit. Inf.* **13**(1) (2012). <http://journals.tdl.org/jodi/index.php/jodi/article/view/5881>
25. Nurmi, D., et al.: The eucalyptus open-source cloud-computing system. In: 9th IEEE/ACM International Symposium on Cluster Computing and the Grid, CCGRID 2009, pp. 124–131. IEEE (2009). <http://ieeexplore.ieee.org/document/5071863/>
26. Peter, M., Timothy, G.: The NIST definition of cloud computing (2011). <http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf>
27. Sefraoui, O., Aissaoui, M., Eleuldj, M.: OpenStack: toward an open-source solution for cloud computing. *Int. J. Comput. Appl.* **55**(3), 38–42 (2012)
28. Smith, M., et al.: DSpace - an open source dynamic digital repository. *D-Lib Mag.* **9**(1) (2003). <https://doi.org/10.1045/january2003-smith>. <http://www.dlib.org/dlib/january03/smith/01smith.html>
29. Sotomayor, B., Montero, R.S., Llorente, I.M., Foster, I.: Virtual infrastructure management in private and hybrid clouds. *IEEE Internet Comput.* **13**(5), 14–22 (2009). <https://doi.org/10.1109/MIC.2009.119>. <http://ieeexplore.ieee.org/document/5233608/>
30. Thakar, A., Szalay, A.: Migrating a (large) science database to the cloud. In: Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing - HPDC 2010, p. 430. ACM Press, New York, June 2010. <https://doi.org/10.1145/1851476.1851539>. <http://dl.acm.org/citation.cfm?id=1851476.1851539>
31. Tramboos, S., Humma, Shafi, S.M., Gul, S.: A study on the open source digital library software's: special reference to DSpace, EPrints and Greenstone. *CoRR abs/1212.4* (2012). <http://arxiv.org/abs/1212.4935>
32. Verno, A.: IVDB... for free! implementing an open-source digital repository in a corporate library. *J. Electron. Resour. Librarianship* **25**(2), 89–99 (2013). <https://doi.org/10.1080/1941126X.2013.785286>. <http://dx.doi.org/10.1080/1941126X.2013.785286>
33. VMware: What is a hypervisor? (2018). <https://www.vmware.com/topics/glossary/content/hypervisor>
34. VMware: vSphere Hypervisor (2019). <https://www.vmware.com/products/vsphere-hypervisor.html>
35. Waddington, S., Zhang, J., Knight, G., Hedges, M., Jensen, J., Downing, R.: Kindura: Repository services for researchers based on hybrid clouds. *J. Digit. Inf.* **13**(1) (2012)
36. Wu, J., et al.: Migrating a digital library to a private cloud. In: 2014 IEEE International Conference on Cloud Engineering, pp. 97–106, March 2014. <https://doi.org/10.1109/IC2E.2014.77>. <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6903462>

Multimedia Processing



Towards Automatic Cataloging of Image and Textual Collections with Wikipedia

Tokinori Suzuki¹ , Daisuke Ikeda¹, Petra Galuščáková² ,
and Douglas Oard²

¹ Kyushu University, Fukuoka 8190395, Japan
suzuki.tokinori.070@s.kyushu-u.ac.jp,
daisuke@inf.kyushu-u.ac.jp

² University of Maryland, College Park, MD 20742, USA
{petra, oard}@umd.edu

Abstract. In recent years, a large amount of multimedia data consisting of images and text have been generated in libraries through the digitization of physical materials into data for their preservation. When they are archived, appropriate cataloging metadata are assigned to them by librarians. Automatic annotations are helpful for reducing the cost of manual annotations. To this end, we propose a mapping system that links images and the associated text to entries on Wikipedia as a replacement for annotation by targeting images and associated text from photo-sharing sites. The uploaded images are accompanied by descriptive labels of contents of the sites that can be indexed for the catalogue. However, because users freely tag images with labels, these user-assigned labels are often ambiguous. The label “albatross”, for example, may refer to a type of bird or aircraft. If the ambiguities are resolved, we can use Wikipedia entries for cataloging as an alternative to ontologies. To formalize this, we propose a task called *image label disambiguation* where, given an image and assigned target labels to be disambiguated, an appropriate Wikipedia page is selected for the given labels. We propose a hybrid approach for this task that makes use of both user tags as textual information and features of images generated through image recognition. To evaluate the proposed task, we develop a freely available test collection containing 450 images and 2,280 ambiguous labels. The proposed method outperformed prevalent text-based approaches in terms of the mean reciprocal rank, attaining a value of over 0.6 on both our collection and the ImageCLEF collection.

Keywords: Entity linking · Multimedia · Wikipedia · Test collection

1 Introduction

In recent years, large amounts of analog materials have been digitized by libraries and museums to preserve their contents [1]. This has led to a large amount of multimedia collections consisting of image and text data. When these data are archived, librarians assign appropriate metadata to them to catalogue them for reference. Manual annotations for these ever-growing digital archives are expensive. To solve this problem, we propose a mapping system that links image and textual multimedia data to appropriate

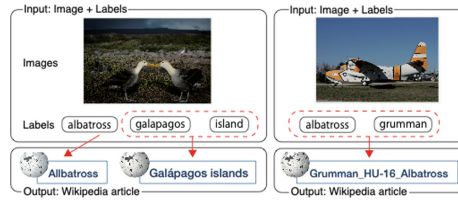


Fig. 1. “albatross” label is linked to albatross (bird) Wikipedia page on the left. By contrast, “albatross” label along with “grumman” label are linked to an aircraft model page on the right.

entries on Wikipedia as a replacement for cataloging. The use of the web data, especially Linked Data, to supplement ontologies for cataloging has been attempted for music archives [2] and has been discussed in the context of libraries [3, 4]. Our mapping utilizes Wikipedia for an ontology of collections and processes of annotating metadata to these collections. It can thus be beneficial in various ways, not only for automatic annotations but also for accommodating emergent metadata quickly updated by users on Wikipedia as an ontology and for implementing Q&A systems. This is an example of the application of such a system that can answer visitors’ questions about the contents of exhibited images using knowledge from Wikipedia.

We first target images and the associated labels linked from the photo-sharing site Flickr¹ to articles on Wikipedia. When users upload photos to the site, they tag them with labels describing the photos [5]. However, users tend to freely assign labels and ambiguities naturally occur in the designation. If these ambiguities are solved for, we can link Wikipedia entries with records of image collections as catalogue metadata.

To formalize the task, we introduce the task *image label disambiguation (ILD)*, in which an image and its associated user-assigned labels are provided as a query, the system in response identifies an entry represented by labels in Wikipedia. Figure 1 illustrates the overview of the task. The two pictures shown are examples from Flickr in which the same label, “albatross”, is used to refer to different types of entities. The image on the left is that of the bird called albatross whereas the one on the right is that of an aircraft of the same name. In the ILD task, the system links the label “albatross” either to the bird or the aircraft model.

There are two general ways of performing the above task. One involves using other labels (e.g., the label “Galápagos islands” provides a hint that the image on the left may be that of the bird native to the Galápagos Islands) or image features (e.g., the image on the right has a shape more similar to that of an aircraft than that of a bird).

Labels are used as text data in traditional entity linking. The entity linking task has been extensively studied in the context of selecting a Wikipedia page to which to link a given mention of an entity in text (the so-called “wikification” task) [6–9]. In general, this line of work has focused on linking entity mentions found in narrative text, e.g., newspaper articles, whereas image descriptions, which is our focus in the ILD, often contain a few words.

¹ <https://www.flickr.com/>.

The other ways of disambiguating labels of images involves image recognition. Image recognition using deep learning has been actively studied in the last decade. Although the performance of recent classifiers has been impressive, the applicability of image classifiers to ILD is restricted because of the scarcity of training data. Image classification requires large amounts of training data. Preparing enormous amounts of data covering all possible entities, such as the “Grumman HU-16 Albatross” aircraft model, is unrealistic.

There are shortcomings in both the above approaches when applied to ILD. We propose solving ILD by combining textual and image features to remedy the defects of each. Classification labels, such as “albatross (bird)” for the image on the left and “aircraft” for that on the right in Fig. 1, can be used as clues for disambiguation. Based on this idea, we propose a hybrid approach using user labels and image features generated by CNN classifiers.

Since ILD is a new task that requires a testbed, we develop a test collection by combining both of the aspects of wikification with image classification. The collection consisted of 450 images and a total of 2,280 ambiguous target labels. Compared with the test collection for image classification, this collection was small because it was manually annotated. However, our initial focus is on establishing a highly accurate linking approach for this collection as a first step in research on ILD.

The contributions of this paper are threefold:

1. We introduce a task called image label disambiguation in which, given an image and associated user labels as input, the system links the labels to Wikipedia pages for the corresponding entity.
2. We propose a hybrid approach that uses labels of images and objects generated by image classifiers using a CNN. The results of experiments show the superior retrieval performance of the proposed approach to prevalent approaches, with an improvement of approximately 0.1 MRR on two test collections.
3. We develop a freely available test collection² containing 450 images and 2,280 ambiguous target labels for evaluating ILD systems.

The remainder of this paper is structured as follows: Sect. 2 summarizes related work and Sect. 3 introduces test collections used in the evaluation and details of the development of our ILD test collection. Section 4 describes our proposed approach, Sect. 5 describes experiments used to evaluate it and Sect. 6 concludes this paper.

2 Related Work

In this section, we summarize research in three areas: image recognition, entity linking, and test collection.

Image Recognition. Image recognition using neural network models has been actively studied in the last decade [10–13]. Neural network architectures of image recognizers now use neural network models with deep layers of convolutional networks, such as in

² <https://zenodo.org/record/3353813>.

AlexNet [10] and VGGNet [11]. These structures have exhibited impressive performance on image recognition tasks. Recent models [12, 13] have explored deeper layers in these architectures. However, applying image recognition to ILD may not be suitable due to the difficulty of preparing training data, which is crucial for adequate classification performance in image recognition. For example, the concept of the Galápagos Islands in Fig. 1 has too many visual representations for the classification. Furthermore, image recognition using neural network models requires large amounts of training resources. Preparing massive amounts of training data with labels over specific entities is unrealistic.

Entity Linking in Text or Image. Entity linking is the task of identifying entity mentions in text passages and linking them to corresponding entities in a knowledge base, especially called wikification in the case of Wikipedia. Wikification systems have been extensively studied. Cucerzan’s approach, for example, uses similarity based on Wikipedia anchor text (words that label a web link) [6], whereas some studies make use of graphs. Ratinov *et al.* leveraged the link structure of Wikipedia as a basis for disambiguation [7]. Moro proposed a graph-based approach for the semantic interpretation of text. Cheng and Roth, by contrast, leveraged linguistic features, coreference relations and named entity recognition [8]. Ganea used contexts with a probabilistic model [9]. In general, this line of work has focused on linking entity mentions in narrative text (e.g., newspapers articles), whereas image labels, which are our focus in ILD, often contain few words.

Entity linking has also been studied for images and is called *visual entity linking* [14, 15]. It involves detecting regions of objects in images and linking them with entities in the associated description. In one example in a past paper, a motorcyclist and bike were identified from the target description in Flickr8k [16] “A motorcycle racer leans his bike”. But we want more granular entities, such as the model of the motorcycle that users specify for the image. Moreover, in contrast to visual entity linking, all user-assigned labels do not have corresponding regions in images, e.g., Galápagos islands for the left of Fig. 1.

Test Collection. Test collections for images have been developed for image recognition tasks such as the PASCAL VOC [17], MS COCO [18] and ILSVRC [19]. However, the classification systems of these collections are too broad for the assessment of ILD. The PASCAL VOC contains 20 classes of notional taxonomy and MS COCO contains 92 common object categories. The ILSVRC 1,000 synset task collection uses ImageNet [20] ontology for classification. However, these image recognition collections have no user-assigned labels. For image annotation tasks, NUS-WIDE [21] contains Flickr images with user-assigned labels. The correspondence of the labels and their entities have not been solved for, however.

3 Test Collection

In this section, we describe two test collections that can be used for ILD evaluation. First, we briefly explain the ImageCLEF collection [22] for the scalable image annotation task. Second, we describe the development of a new test collection (Animal Name collection).

3.1 ImageCLEF Collection

The ImageCLEF collection was developed for the scalable image annotation task where systems automatically annotate input pictures collected from the web. The annotations were based on popular user queries for image searches.

This collection consists of 7,291 images with 207 manually annotated concepts. The concepts are defined as in WordNet [23] synsets and have links to Wikipedia articles in most cases. This correspondence between the concepts of images and Wikipedia entries allowed us to evaluate ILD systems on the ImageCLEF collection, for which we selected 53 of the concepts that had associated Wikipedia pages and had been defined in the 1,000 classes of ILSVRC [19] as target entities for disambiguation. The latter condition was used to evaluate approaches relying on the 1,000-class classifier (explained in Sect. 4). The 53 entities featured a variety of types, such as guitars, mushrooms and lakes (the list of the selected entities is available in our uploaded test collection). Using these entities, we selected images assigned to them in the concepts. As a result, 1,197 images and the assigned concepts were available for evaluation.

3.2 Building a Test Collection for ILD

ImageCLEF can be used for ILD evaluation. However, the concepts of the collection are general words reflecting searchers' queries for images. We are interested in more granular target entity types, such as the Grumman model HU-16 Albatross shown in Fig. 1. This motivates us to develop a test collection designed for ILD evaluation. To formulate the gold standard of correspondences between labels of images and granular entity types, we annotated the entries in the new collection.

Data Collection. We first defined categories of the test collection. We used the policy whereby labels of the test collection of ILD can map to several entities leading to ambiguity. We chose animal names because they are often used as brand names of products or nicknames of sports teams, and this satisfies the ambiguity requirement. We selected 15 animal names listed on a Wikipedia page³ of 512 general animal names.

We then collected images and tagged labels assigned by users on Flickr by searching the chosen animal names as queries. We randomly selected 30 images for each animal name as labels. The selected images were not limited to pictures of animals, but included any genre, e.g., automobile and airplane. The label "jaguar", for example, refers not only to a wild cat but also a luxury car brand. Figure 2 shows the distribution of the main subjects of the pictures. It represents the ambiguity of the

³ https://en.wikipedia.org/wiki/List_of_animal_names.

selected names, which is desirable for the evaluation. The average number of labels for each category was 180 after camera specification labels had been eliminated, e.g., “nikond5” and “500mmf4”.

Table 1. Examples of entities in two test collections

Test collection	Examples of entities
Animal Name collection	Albatross (bird), Wandering albatross, Grumman HU-16 Albatross, Cricket (sports), Cricket (insect), Jaguar Cars, Jaguar
ImageCLEF 2014 collection	Banana, Bicycle, Cauliflower, Cheetah, Drum, Lion, Potato

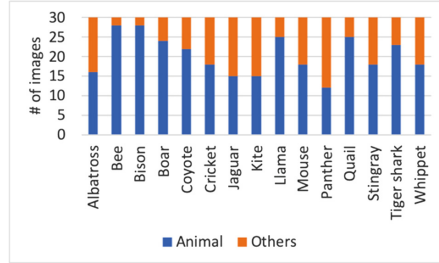


Fig. 2. Distribution of subjects of pictures

Annotation. We recruited five annotators who were graduate students and had studied English as a second language. Each of them was assigned six categories to annotate that were balanced to nearly the same volume across annotators. Two annotators were assigned to each category.

The annotators subjectively determined entities to link to labels of an image. An image and its labels were given to the annotators on a spreadsheet. They identified a label (or labels) which should refer to a given entity in the English version of Wikipedia regardless of whether it was depicted in the picture. Using the labels “Galapagos” and “island” in Fig. 1 as an example, the annotators were asked to judge whether the labels were linked to the Galápagos Islands from the image and the other labels. They then searched Wikipedia pages for entities of the labels by querying their single or multiple labels. In this step, they could modify their queried labels within their interactions with the search results of Wikipedia to obtain candidates. Finally, they judged whether the retrieved pages were appropriate for the label and, if so, assigned the Wikipedia pages as the correct entity on their spreadsheets. If they determined that the labels did not correspond to any Wikipedia page, they assigned a NIL entity to the relevant labels. After three trials of the above procedures, they began annotation. We examined the quality of the outcomes of annotation by investigating inter-annotator agreement and Cohen’s κ coefficient between them [24]. Their agreement was indicated with a value of 0.8, and when κ was over 0.9, this implied almost perfect agreement.

Finally, we created a test collection of 2,280 ambiguous target labels of 450 images, which showed high inter-annotator agreements. The number of images was not as large as that of image test collections intended for supervised learning. The number of labels, however, was large for the testbed, three times that of test collections with a maximum of 700 mentions used for the wikification of text [8]. Table 1 compares examples of

entities in the collection of Animal Name and the ImageCLEF collection. Our collection covered ambiguities among entities, which is desirable for ILD evaluations.

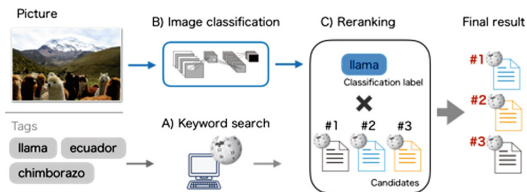


Fig. 3. Overview of proposed approach

4 An Approach with Use of Textual Labels and Object Recognition

This section describes our proposed approach for disambiguating user labels. The approach is generally a reranking method based on the results of a keyword search system. An overview is provided in Fig. 3. It consists of three steps: (A) keyword search system, (B) image classification and (C) reranking.

- (A) The purpose of the keyword search step is to obtain candidate Wikipedia pages for image labels. To generate the pages, we employed a keyword search system with BM25 similarity scoring implemented in Apache Solr (ver. 7.3.0)⁴. All pages for each image label in the search results were used as candidate pages while (C) reranking.
- (B) In the image classification step, we extract the objective information of an input image by using deep learning image classifiers. We implemented two types of classifiers with different policies to generate objective information. (B-1) The first classifier was trained on the primal collection categories described in Sect. 3. It classified an input image to one of these categories. Because collection categories can be a subject in a picture, our aim in using this classifier is to obtain the collection category as an objective label with high accuracy. (B-2) The second classifier aimed to obtain as much objective information as possible. The classified labels were not limited to primal collection categories and included other objects in a picture. For example, Fig. 3 shows the picture of the collection category “llama”. If another label “alp” is obtained by the classifiers, it can be useful to infer the mountain-related contents of the image and disambiguate it from the label “chimborazo” (an inactive volcano in the Andes). Because images may contain several objects, we used the top five classification labels per image.
- (B-1) The aim of classifying categories of collection (Category classifier) is to obtain highly accurate primal categories of the input pictures. To this end, we utilize a

⁴ <http://lucene.apache.org/solr/>.

recent convolutional architecture called the *deep residual network* (*ResNet*) [12]. In this network, each layer is connected through a direct connection and an additional connection of the input layer that formulate the layer as a residual learning function. Using this framework, residual networks can train substantially deeper models that deliver better performances.

The training data were collected by downloading images from ImageNet⁵, where these images were assigned to classes of ImageNet corresponding to the collection categories. We had 400 and 650 images, determined by the minimum number of collected images in each category, in the Animal Name and ImageCLEF collections, respectively. We splitted up the collected images into 70% for training and 30% for validation. We used a 152-layer model of ResNet⁶ trained on 1,000 object classes of ImageNet with the replacement of the last fully connected layer with the output layer for the collection categories, and trained the model with a mini-batch size of 32 and an SGD optimizer. The initial learning rate, weight decay and the momentum were 10^{-4} , 10^{-6} and 0.9, respectively. The accuracy of the trained classifiers was 0.65 on the Animal Name collection and 0.66 on ImageCLEF. The output labels and the probability (confidence score) of the *softmax* layer were used in the (C) reranking step.

- (B-2) We used a multi-label classifier trained on 1,000 object classes of the ILSVRC classification task [19] (1k classifier). A total of 1,000 synsets were selected from ImageNet so that there was no overlap between them; any given synset was not an ancestor of the others in the ImageNet hierarchy. This is an advantage of using the category set. The dataset category contained a variety of classes, such as animals, locations, sports activities, stationery items and gadgets, which are beneficial for inferring the contexts of the pictures. To implement this classifier, we used a pre-trained classifier using Keras's implementation of the ResNet [12] model trained on over 1.2 million images.
- (C) In this step, we used the similarity between candidate Wikipedia pages from (A) and a pair of image labels originally tagged by users and objective labels from (B). We used Wikipedia pages from only (A) and discarded the ranking and score of the search results to measure the relation between the Wikipedia pages and the labels.

Because the vocabulary of the classification labels was limited to the trained class system, term-matching retrievals could not distinguish among the relatedness of the labels to entities if they were in different vocabularies. To handle the relatedness of words in the ranking, we employed word embeddings using *word2vec* [25] to this similarity calculation, motivated by its recent success in such ranking tasks as web search [26] and document retrieval in software engineering [27]. Word2vec learns word embeddings by maximizing the log-conditional probability of a word given words within a fixed-sized window, so that words with similar meanings are associated with similar vectors. Word vectors, which contain useful knowledge about the distributional properties of words, allow the ranking to recognize relations. For example,

⁵ <https://www.image-net.org/>.

⁶ <https://keras.io/applications/#resnet>.

“alp” was a classification labels of B-2 for the image in Fig. 3, which indicates mountain-relatedness and closeness to “chimborazo”, an inactive volcano in South America, although term-matching retrievals overlooked similarity.

There are two models for learning word embeddings in word2vec: *skip-gram* and *continuous bag-of-words*. Both models produce similar embeddings in terms of quality and quantity. We used the skip-gram model with a window size of five to learn word vectors of 300 dimensions using English Wikipedia dump data on 1st October 2017.

The overall similarity was computed by the similarity between the Wikipedia pages and pairs of user labels, w_q and objective labels, w_l defined as follows:

$$\text{sim}(T, Q) = \alpha \sum_{w_q \in Q} \text{sim}(T, w_q) + (1 - \alpha) \sum_{w_l \in L} \text{sim}(T, w_l), \quad (1)$$

where T represents the titles of candidate Wikipedia pages, Q represents the set of user labels of images and L represents a label or labels automatically generated by image classifiers. In preliminary experiments, using words in the titles of articles to compute similarity among them delivered the best retrieval performance after several trials of document modeling. α in the above represents a weighting parameter of similarities on labels and objective labels. In computing the similarity of automatic labels, similarities to Wikipedia pages were weighted by using the confidence scores of the classifiers:

$$\text{sim}(T, w_l) = \frac{1}{|T|} \sum_{w_t \in T} \text{sim}(w_t, w_l) \times \text{score}(w_l), \quad (2)$$

where w_t is a word in a Wikipedia document, w_l is a word in classification labels and $\text{score}(w_l)$ is a confidence score of the prediction of the given label by the classifiers. Similarity between words was computed by the cosine distance of their vector representations:

$$\text{sim}(w_1, w_2) = \cos(\mathbf{w}_1, \mathbf{w}_2) = \frac{\mathbf{w}_1^T \mathbf{w}_2}{\|\mathbf{w}_1\| \|\mathbf{w}_2\|}. \quad (3)$$

This is the inner product of two vectors \mathbf{w} of words w learned by word embedding. Although *out-of-vocabulary (OOV)* words also pose challenges to similarity computation, as mentioned by Mitra [26], we ignored OOV words here.

5 Experiment

The experiment aimed to answer following two questions: First, to what extent can the proposed approach solve ILD tasks compared with other approaches in the literature? For this, we evaluated several approaches on image label disambiguation targeting all given labels of images. Following this, when labels that represented objects in images were only used, this was considered image classification. The question was that of whether the proposed approach, which uses both text labels and image features, could outperform image classifiers that use only image features. For this, we evaluated several approaches on image classification tasks.

Two test collections, as explained in Sect. 3, were employed in this evaluation. One is our Animal Name collection containing 2,280 ambiguous labels of 450 images. The other was the ImageCLEF collection. We used 1,197 images using 53 concepts. The disambiguation targets were 10,573 conceptual labels in total. For the entity set, English Wikipedia dump data on 1st October 2017, which contained 5,486,204 pages, were used.

Table 2. Image label disambiguation results. ** means statistical significance compared to results of BM25 at 0.01 level.

Method	Animal Name collection			Image CLEF 2014 collection		
	MRR	R@1	R@10	MRR	R@1	R@10
TF-IDF	0.480	0.436	0.532	0.668	0.632	0.706
BM25	0.509	0.471	0.577	0.627	0.595	0.706
Wikification	0.523	0.481	0.601	0.628	0.595	0.708
Method 1	**0.609	0.558	0.684	**0.715	0.711	0.718
Method 2	**0.583	0.518	0.679	**0.719	0.711	0.730

5.1 Image Label Disambiguation Setting

In this setting, given image labels as a query, the system returned a ranked list of Wikipedia pages for the relevant entities. In this task, all image labels were targets to be disambiguated and used as queries. For example, five labels—“albatross”, “galapagos” and “island” on the left in Fig. 1, and “albatross” and “grumman” on the right—were used for image label disambiguation. The number of labels in each query, i.e., single or multiple labels, was determined in the annotations according to Sect. 3.

We evaluated five approaches: TF-IDF, BM25, Wikification, Method 1 and 2. Method 1 and 2 are our proposed approaches. Method 1 used a collection category classifier and Method 2 the 1k classifier (B-1 and B-2 in Sect. 4, respectively). TF-IDF and BM25 are text retrieval systems. Retrieving English Wikipedia by querying using user labels does not much differ from traditional retrieval, so that we used $k_1 = 1.2$, $b = 0.75$ in Okapi at TREC-3 as free parameters settings for BM25 [28].

The wikification system (Wikification) was proposed by Cucerzan [6]. Considering that user labels tagged to a picture consisted of a few keywords, where the linguistic structure was rarely maintained, this method is a better choice than other wikification systems that rely on the linguistic features mentioned in Sect. 2. Wikification retrieves correct entities for a query by computing cosine similarity between word occurrence vectors of an input document and the candidate Wikipedia pages with same surface forms as that of the use of the label. We prepared all pages listed in a Wikipedia disambiguation page for a queried label as candidate Wikipedia pages.

Table 2 shows the results in terms of MRR [29] and Recall@1, 10. The proposed approaches achieved the best performance in both collections. The MRR of Method 1 was 0.609 on the Animal Name collection and that of Method 2 was 0.715 on

ImageCLEF. These results were superior to those of BM25 used in reranking steps in Method 1 and 2. We also assessed statistical significance between the results of BM25, and Method 1 and 2 with a two-tailed paired t-test. The difference was significance at a 0.01 level.

Some successful cases of Method 1 and 2 are shown in Fig. 4. The figure displays pairs of an image and the correct entity, where Method 1 or 2 output the correct entity but the other approaches could not. Our approach, for example, generated both a coyote and a Coyote Buttes for the label “coyote”. On this category, Method 1 achieved a higher improvement in MRR than BM25. The major errors, by contrast, appeared in users’ free-written labels, which are different from the entity in their surfaces. Neither method could generate the correct entity on its list for 20% of the labels in the Animal Name collection.

Wikification and TF-IDF yielded the second-highest values of MRR. The two tended to output a particular page with high rank in their results for same labels regardless of the visuals of the image. For example, when the query label was the “jaguar” label of an animal or pictures of cars, they persistently ranked a wild cat jaguar page at the first and the page for the carmaker Jaguar around the 50th in their ranked lists.

Analysis of Results. We investigated the influence of the weighting parameter α , representing similarities between user labels and classification labels in Eq. (1), on MRR. Figure 5 shows the MRR of Method 1 and 2 for a value of α of 0.1. The similarity in user labels highly influenced the MRR in both collections. In particular, the MRR of ImageCLEF was flat at any given α , possibly because ImageCLEF labels of general search queries were not ambiguous. On the animal collection, by contrast, MRR increased with α , which also indicates the dominance of user labels in the similarity computation. However, because we fixed α for all categories of the collection to evaluate its influence, the maximum MRR in Fig. 5 was 0.52, smaller than that at the best setting of α in Table 2. Therefore, image features can improve the disambiguation of labels.



Fig. 4. (Left) Successful pairs of images and entities by Method 1 and 2.

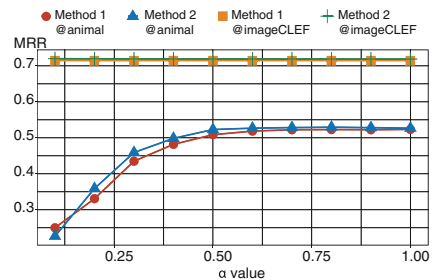


Fig. 5. (Right) MRR at each value of a parameter.

5.2 Image Classification Setting

We picked images and the labels of objects. For the ImageCLEF collection, we used all 1,197 images and our 53 categories. For the Animal Name test collection, we selected images that had been assigned object labels referring to animals. Taking Fig. 1 as an example, we used the label “albatross” for the left image but did not use it for the image on the right because the correct referent of “albatross” was linked to an aircraft entity that is not in animal. In total, 277 images with 15 animal categories were used.

We compared Method 1 and 2 with the two image classifiers B-1 and B-2 in Sect. 4. The first classifier was trained on 1,000 classes (1k classes) of ILSVRC and the second on categories of the two collections (Category).

We evaluated the accuracy of classification as $\text{Accuracy} = \# \text{ of correctly classified images} / \# \text{ of images in category}$, where we considered an image to have been correctly classified when the correct object label was listed in the top five predictions of the 1k classes and Category classifiers because the pictures are tagged with several labels—fewer than five labels in most cases. For Method 1 and 2, the results were counted as correct when they generated the correct Wikipedia page in the first rank in the retrieval results for a queried label.

Table 3 shows the average accuracy of the examined approaches. The proposed approaches achieved the best classification at 0.958 on the Animal Name and one on the ImageCLEF collections. Method 1 and 2 delivered better performance than the 1k classifier and the category classifier, which indicates that they can make use of user labels as text information for classification. A notable case of this success is a spray-painted image of a bee in the bee category, where Method 1 succeeded while the others failed. Although the proposed method generally showed high accuracy, that of Method 2 was low at 0.789 on the Animal Name collection. This was because of failure in the reranking step, even though the correct classification labels were generated by the 1k classifier.

While category had high accuracy above 0.7, 1k classes exhibited low accuracy, primarily because of the differences in the collection of images between Flickr and ImageCLEF, and ImageNet, on which the 1k classes’ model was trained. Compared with ImageNet, which arranged one class per image, image in Flickr and ImageCLEF were from the web, and usually included multiple categorial images, often three or more. The 1k classifier failed on these images.

Table 3. Image classification results (average accuracy)

Test collection	1k classes	Category	Method 1	Method 2
Animal Name collection	0.803	0.889	0.958	0.789
ImageCLEF 2014 collection	0.680	0.741	0.995	1

6 Conclusion

In this paper, we proposed a task called image label disambiguation (ILD), in which the system retrieves correct entities in response to queried ambiguous labels associated with an image, with the aim of automatically cataloging multimedia collections. To evaluate ILD systems, an ILD test collection was developed that is freely available to users and developers. We developed an approach to disambiguate labels by employing user labels of images and a CNN classifier for image classification.

In future work, more sophisticated document modeling, such as using the average of word vectors in a document, will be considered. With regard to ILD approaches, we plan to test a combination of two types of classifiers to further improve the results. We also plan to increase the size of the test collection for more robust evaluation.

References

1. Corrado, E.M.: *Digital Preservation for Libraries, Archives, and Museums*, 2nd edn. Rowman & Littlefield Publishers, Cambridge (2017)
2. Page, K.R., Bechhofer, S., Fazekas, G., Weigl, D.M., Wilmering, T.: Realizing a layered digital library: exploration and analysis of the live music archive through linked data. In: *Proceedings of the 17th ACM/IEEE Joint Conference on Digital Libraries*, pp. 89–98 (2017)
3. Kroegen, A.: The road to BBIBFRAME: the evolution of the idea of bibliographic transition into a post-MARC Future. *Cataloging Classif. Q.* **51**(8), 873–890 (2013)
4. Nurmikko-Fuller, T., Jett, J., Cole, T., Maden, C., Page, K.R., Downie, J.S.: A comparative analysis of bibliographic ontologies: implications for digital humanities. In: *Digital Humanities 2016: Conference Abstracts*, pp. 639–642 (2016)
5. Sawant, N., Li, J., Wang, J.Z.: Automatic image semantic interpretation using social action and tagging data. *Multimedia Tools Appl.* **51**(1), 213–246 (2011)
6. Cucerzan, S.: Large-scale named entity disambiguation based on Wikipedia data. In: *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 708–716 (2007)
7. Ratnov, L., Roth, D., Downey, D., Anderson, M.: Local and global algorithms for disambiguation to Wikipedia. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 1375–1384 (2011)
8. Cheng, X., Roth, D.: Relational Inference for wikification. In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1787–1796 (2013)
9. Ganea, O.E., Ganea, M., Lucchi, A., Eickhoff, C., Hofmann, T.: Probabilistic bag-of-hyperlinks model for entity linking. In: *Proceedings of the 25th International Conference on World Wide Web*, pp. 927–938 (2016)
10. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: *Proceedings of the 25th International Conference on Neural Information Processing Systems*, pp. 1097–1105 (2012)
11. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv preprint [arXiv:1409.1556](https://arxiv.org/abs/1409.1556)* (2014)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the 29th IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778 (2016)

13. Szegedy, C., Loffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, Inception-ResNet and the impact of residual connections on learning. In: Proceedings of the 31st AAAI Conference on Artificial Intelligence, pp. 4278–4284 (2015)
14. Weegar, R., Hammarlund, L., Tegen, A., Oskarsson, M., Åström, K., Nugues, P.: Visual entity linking: a preliminary study. In: Proceedings of Cognitive Computing for Augmented Human Intelligence Workshop at the 28th AAAI Conference on Artificial Intelligence, pp. 46–49 (2014)
15. Tilak, N., Gandhi, S., Oates, T.: Visual entity linking. In: Proceedings of the 2017 International Joint Conference on Neural Networks, pp. 665–672 (2017)
16. Hodosh, M., Young, P., Hockenmaier, J.: Framing image description as a ranking task: data, models and evaluation metrics. *J. Artif. Intell. Res.* **47**(1), 853–899 (2013)
17. Everingham, M., Eslami, S.M.A., Gool, L.V., Williams, C.K.I., Winn, J., Zisserman, A.: The Pascal visual object classes challenge: a retrospective. *Int. J. Comput. Vis.* **111**(1), 98–136 (2014)
18. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
19. Russakovsky, O., et al.: Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **115**(3), 211–252 (2015)
20. Deng, J., Dong, W., Socher, R., Li, L., Li, K., Fei-Fei L.: ImageNet: a large-scale hierarchical image database. In: Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255 (2009)
21. Chua, T.S., Tang, J., Hong, R., Li, H., Luo, Z., Zheng, Y.: NUS-WIDE: a real-world web image database from National University of Singapore. In: Proceedings of the 8th ACM International Conference on Image and Video Retrieval (2009)
22. Gilbert, A., et al.: Overview of the ImageCLEF 2016 scalable web image annotation task. In: Proceedings of the Seventh International Conference of the CLEF Association (2016)
23. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press, Cambridge (1998)
24. Carletta, J.: Assessing agreement on classification tasks: the kappa statistic. *Comput. Linguist.* **22**(2), 249–254 (1996)
25. Mikolov, T., Sutskever, I., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Proceedings of the 26th International Conference on Neural Information Processing Systems, pp. 3111–3119 (2013)
26. Mitra, B., Nalisnick, E., Craswell, N., Caruana, R.: A Dual Embedding Space Model for Document Ranking. arXiv preprint [arXiv:1602.01137](https://arxiv.org/abs/1602.01137) (2016)
27. Ye, X., Shen, H., Ma, X., Bunescu, R., Liu, C.: From word embeddings to document similarities for improved information retrieval in software engineering. In: Proceedings of the 38th IEEE International Conference on Software Engineering, pp. 404–415 (2016)
28. Robertson, S.E., Walker, S., Jones S., Hancock-Beaulieu, M.M., Gatford, M.: Okapi at TREC-3. In: Proceedings of the third Text REtrieval Conference, pp. 109–126 (1996)
29. Craswell, N.: Mean reciprocal rank. In: Liu, L., Özsu, M.T. (eds.) Encyclopedia of Database Systems. Springer, Boston (2009). https://doi.org/10.1007/978-0-387-39940-9_488



Automatic Semantic Segmentation and Annotation of MOOC Lecture Videos

Ananda Das^(✉) and Partha Pratim Das^(✉)

Indian Institute of Technology, Kharagpur, India
anandadas2005@gmail.com, ppd@cse.iitkgp.ac.in

Abstract. Locating and searching a desired topic inside a long MOOC lecture video is a rigorous and time consuming task unless video meta data provides annotation information of different topics taught in it. In this work we propose a system that performs topic wise semantic segmentation and annotation of MOOC lecture videos to mitigate user effort. As input, our system takes a lecture video, its associated speech transcript, and list of topic names (available in the form of course syllabus) and produce an annotated information of each semantically coherent topics taught in that video. We have two major contribution in this work. First, we have applied state-of-the art neural network based text segmentation technique on textual information to obtain topic boundaries. Secondly, for annotation we have utilized a combination of visual and textual information, that assigns accurate topic name for each segment. We have tested our method on 100 lecture videos hosted in NPTEL [1] and on an average get $\sim 72\%$ segmentation accuracy, better than the existing approaches available in literature.

Keywords: e-learning · Lecture video annotation · Lecture video content retrieval · Multimedia application · Semantic segmentation

1 Introduction

In last few years lots of lecture videos are available on internet and it's growing day by day. Usually content of a large (~ 1 h) lecture video constitutes of several smaller topics taught sequentially. To facilitate students to minimize topic searching effort, easily locating and browsing a topic inside the video, lecture video metadata needs to have topic wise segmentation information in it. Manual segmentation process is the most accurate one, but it is very costly and time consuming task. Several research effort have been put on to perform automatic segmentation of lecture videos. This is mainly done by analyzing textual information available in subtitle and transcript file, and visual contents like video file synchronized with lecture slides or the slide images. In [17], a modified version of TextTiling [14] is proposed to obtain segment boundaries from audio transcripts. To predict topic transition in textual content, cue phrases are used which are represented by N-gram model [18] and word embedding [12]. As an additional

semantic information [19] used Wikipedia text. In [18] an SVM is trained to learn temporal transition cues in the lecture video. In their work [5, 8, 22] propose a system that uses OCR to find segment boundaries by analyzing slide images extracted from synchronized lecture video. In [6] an SVM is trained to detect transition of events (like instructor is writing on board, slide presentation) and segment boundaries are derived from these event transition.

In our work we have considered three different semantic resources as follows. (1) MOOC lecture videos publicly available in *nptel.ac.in* [1] website. These videos have duration ~ 1 h and synchronized with lecture slides. (2) Machine generated speech transcripts associated with each video file. and (3) List of topic names which are available in the form of course syllabus (NPTEL lecture video comes into series form). For semantic segmentation task, we have adopted neural network based NLP technique. Annotation of each segmented topic involves utilization of both video file and topic name list. Annotation process relies on image processing methods. Finally for each lecture video we create a list of semantically coherent topic names along with their starting and ending time in the lecture video. A glimpse of our final annotated output is shown in Table 1.

The paper is organized as follows. In Sect. 2 we discuss detail methodology and different experiments. Section 3 focuses on performance evaluation, comparative analysis and discussion. Finally we conclude our work in Sect. 4.

2 Methodology

We can divide our solution into two parts. First, we create semantically coherent text segments from the speech transcript and obtain topic boundaries in text data. Now these boundaries are inversely mapped to the original video and store the corresponding time stamps, which act as the starting time and ending time of each topics. Next task is segment annotation, that assigns topic name of each segmented part. For annotation purpose we use lecture video file and topic name list. In the following subsections we discuss these steps in more detail.

2.1 Segment Creation

A transcript file is considered as a sequence of sentences with fixed number of words. Semantic segmentation of text is formulated as a binary classification problem. Given a sentence S_K with K sentences S_0, S_1, \dots, S_{K-1} preceding it and K sentences $S_{K+1}, S_{K+2}, \dots, S_{2K}$ following it, we need to classify whether sentence S_K is the beginning of a segment or not. To represent sentence vector we use InferSent [9], which tries to build a universal sentence encoding by learning sentence semantics of Stanford Natural Language Inference (SNLI) [7] corpus.

Table 1. Sample output

Topic Name	Start time (mm : ss)	End time (mm : ss)
Lecture Outline	01:24	13:02
Iterative Refinement Search	13:06	24:50
Hill Climbing/Gradient Descent	24:53	40:10
Iterative Depening A^*	40:12	47:05
Multi Objective A^*	47:11	55:07

Next, each encoded sentence along with its K -sized left and right contexts are fed to a sentence classifier, whose output is used to determine topic boundaries.

Sentence Classification. Our classification approach is similar in spirit of the method proposed by Badjatiya et al. [4] with some modifications. Relationship of a sentence with its left and right contexts matter most to becoming it as the segment boundary. Hence, first we compute context vectors of left and right context using stacked bidirectional LSTM [15] with an attention layer. Also two sentences located at equal distance, one in the left context and other on the right, may have different impact on the middle sentence to be it segment boundary. To handle this we have applied a soft attention layer on top of the bi-LSTM layer. Attention allows to assign different weights to the different sentences resulting better model accuracy. We define attention vector, z_s as

$$e_i^{K \times 1} = H_i^{K \times d} \times W^{d \times 1} + b_i^{K \times 1}, a_i = \exp(\tanh(e_i^T z_s))$$

$$\alpha_i = \frac{a_i}{\sum_p a_p}, v_i = \sum_{j=1}^K \alpha_j h_j$$

where, d is dimension of the output vector of last bi-LSTM layer, H is the output of the bi-LSTM layer, W and b are the weight matrix and bias vector and v_i is the resultant context embedding. After computing both the context vector in similar manner, we compute semantic relatedness between left context (let it be C_L), right context (C_R) and the middle sentence (M) by the following way.

1. Concatenation of three representation (C_L, M, C_R)
2. Pair wise multiplication of the vectors ($C_L * M, M * C_R, C_R * C_L$)

The resulting vector contains relationship among context vectors and middle sentence. The resultant vector is fed to a binary classifier consists of fully connected layers culminating in a soft-max layer. Architecture diagram of sentence classifier is shown in Fig. 1.

Video Segment Creation. Sentences classified as segment boundaries are inversely mapped to the video file to find corresponding time stamps in the video. Let us assume, length of a video file is T seconds and corresponding transcript file consists of N sentences (S_1, S_2, \dots, S_N). Now, if we find a topic starts at S_p th sentence and ends at S_q th sentence then corresponding starting and ending time in the video will be $T \frac{(p-1)}{N}$ th and $T \frac{(q-1)}{N}$ th second respectively.

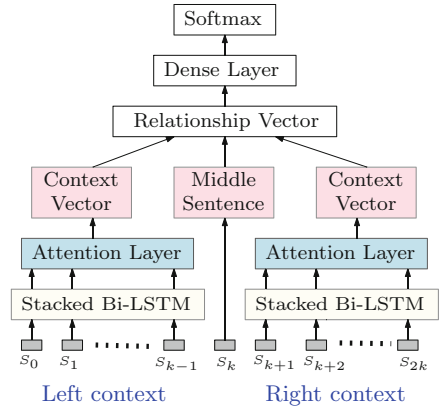


Fig. 1. Sentence classifier

Model Training. For training we randomly select ~ 300 Wikipedia [3] pages having $\sim 55K$ sentences, with $\sim 8\%$ sentences being the segment boundaries. Section boundaries in the Wikipedia pages are considered as segment boundaries. We handle this highly imbalanced dataset by selecting weighted binary cross entropy loss function [13], which gives higher weight to the minority class and lower weight to the majority class. Loss function is defined as, $\ell = -\frac{1}{N} \sum_{n=1}^N w_n (t_n \log(o_n) + (1-t_n) \log(1-o_n))$. We neglect the introduction section of each page, as it briefs the overall idea of the whole page. We have used AdaDelta [23] optimizer with dropout 0.3, epoch size as 35 and mini batch size as 40.

Context size K plays an important role in segmentation. Higher value of K captures some unnecessary context while very low value may not properly capture the original context. We optimize the value of K as 10 by measuring the segmentation accuracy on our dataset. Details of dataset, performance evaluation and accuracy measurements are discussed in Sect. 3 and Table 2. Figure 2 shows the experimental result for different values of K .

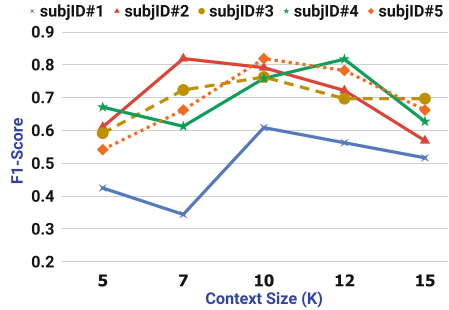


Fig. 2. Context size vs F1-score

2.2 Segment Annotation

In this stage we assign topic name to each video segment obtained from previous stage. Annotation process involves extracting titles from synchronized lecture slides and finding proper topic name from topic name list.

Slide Title Extraction. First we sample video frames at 3s interval. Video file displays presentation slides, instructor face and board works. Here we are only interested in slide frames as other two types do not contain any semantic information. We classify the video frames to select the slide frames only. We leverage transfer learning using inception-V3 [20] pre-trained model to classify these frames. We re-train inception-V3 using 4921 slide-frames, 4702 board-work frames and 4345 instructor-frames and obtain frame classification accuracy (F1-score) more than 97%. To extract text lines from each slide frame, we apply Stroke Width Transform (SWT) [11] on each slide frame. SWT creates a bounding box surrounding each text line. Next, cropped text line images are passed to an OCR engine for text recognition. Open source OCR library *tesseract* [2] is used for this task. To find slide titles from detected text lines we take top 23% lines [21] of the slide images. These slide titles are used later to assign topic name for each coherent topic segment.

Finding Topic Name. For a video segment, this step finds proper topic name using a list of topic names and identified slide titles. First we remove duplicate

titles, punctuation marks and apply lemmatization on each slide title. Let us assume, within time stamp TS_p and TS_q we get N unique titles denoted as $t_1, t_2 \dots t_N$. Now for each topic name, we measure cumulative similarity score between that topic name and all N titles. Topic name that have highest similarity score is assigned to this video segment. Once a topic name is assigned, we remove that name from the topic name list so that same name is not assigned to multiple video segment. Word Mover's Distance [16] is used for similarity measure. Using these methods we can successfully annotate each video segment obtained previously.

3 Dataset, Result, Comparison and Discussion

Dataset and Result. To evaluate our proposed system we take 100 lecture videos of 10 different subjects, publicly available in *nptel.ac.in*. [1], duration of each video is ~ 1 h. Following are some characteristics of our dataset.¹

1. Videos have varying resolution. We use 30 videos with resolution 320×256 , 30 videos of 640×480 and 40 videos having resolution 720×576 .
2. All the video files are synchronized with lecture slides, hence slide frames are always present inside each lecture video. Additionally videos may or may not have instructor faces and board works.
3. We consider videos those have single instructor through out the whole video. Also all the videos are associated with speech transcripts.

Ground truth is prepared by topic wise segmenting and annotating each video by corresponding domain experts. Each annotated topic information is a triplet of *topic-name*, *starting-time* and *ending-time* as shown in Table 1. We evaluate the result of our approach against this ground truth. The evaluation process is as follows: consider a triplet (*tpName*, *sTime*, *eTime*) is generated from our system, we match *tpName* with our ground truth. If both *sTime* and *eTime* approximately match with the *starting-time*, *ending-time* of corresponding *topic-name* in ground truth then we mark the generated triplet as a successful one. Here approximate match means we allow 10s deviation of time stamp on either side. Accuracy measure of each subject is shown in Table 2.

Comparison and Discussion. We compare our method with other methods in literature by applying them on our dataset. Following are some brief overview of the compared methods and Table 2 shows the comparison results.

1. **Modified TextTiling (MTT)**. [17] This approach divides the whole speech transcripts into multiple blocks of 120 words with a overlapping of 20 words. Cosine similarity is measured between two neighbouring blocks using noun, verb and cue phrases as feature vectors. Topic boundaries are identified based on a threshold value of the similarity score.
2. **D&V** [12] makes a modification of the above approach by using word embedding to represent cue-phrases.

¹ Available at <https://github.com/ananda2016/SemanticSegmentation>.

Table 2. Topic segmentation accuracy measure and comparison

Subj ID	Subject Name (No. of Topics)	(F1 score of the segmentation)				
		OURS	Comparison with other methods			
			MTT	D&V	ATLAS	Lecture Khoj
1	Foundation of Scientific Computing (85)	0.61	0.54	0.57	0.64	0.62
2	Advanced Control System Design (69)	0.79	0.71	0.76	0.72	0.69
3	Multiphase Flow (73)	0.76	0.68	0.74	0.70	0.66
4	Computational Techniques (68)	0.76	0.67	0.72	0.66	0.64
5	Data Structures and Algorithms (81)	0.82	0.75	0.80	0.77	0.76
6	Design of Digital VLSI Circuits (59)	0.61	0.57	0.57	0.64	0.68
7	Intr. to Prob Solving and Prog. (60)	0.67	0.58	0.62	0.62	0.56
8	Cryptography and Network Security (48)	0.82	0.72	0.74	0.72	0.70
9	Low Power VLSI Ckts. and Systems (71)	0.69	0.61	0.67	0.64	0.65
10	Mathematics II (50)	0.71	0.62	0.69	0.65	0.67
	ALL (694)	0.72	0.65	0.69	0.68	0.66

- ATLAS** [18] uses a combined approach of predicting transition cues both from visual and textual contents. SVM [10] and N-gram language models are used for finding visual and textual transitions respectively.
- LectureKhoj**. [5] Determines topic transitions by analyzing textual contents of lecture slides shown in the lecture video.

In Table 2 we find that our approach obtains better accuracy for most of the subjects except *subjID#1* and *subjID#6*. For these subjects accuracy drops to 60%, while, on the other hand we get more than 80% accuracy for *subjID#5* and *subjID#8*. After analyzing we find two main reasons behind this uneven result.

- Quality of Transcript File.** Our segmentation method only uses transcript file, hence segmentation accuracy is heavily dependent on its quality. Our method performs well for subjects with high quality speech transcript, achieving ~80% segmentation accuracy. While for transcripts with poor speech recognition, our method is not able to compute proper semantic relationship between sentences, resulting in a drastic performance drop. As ATLAS and LectureKhoj consider visual information during segmentation, hence performing better than other methods for poor quality transcript file.

2. **Comprehensiveness of Topic Name List.** Our annotation process is highly dependent on topic name list. For comprehensive topic name list we can assign topic name to the video segments more accurately while there occurs some error for incomplete topic name list. Subjects associated with accurate transcript file and comprehensive topic name list, our approach achieves more than 75% segmentation accuracy, while with any one of them is not up to the mark, accuracy lies in between 65% to 75%. For subjects whose accuracy is less than 65% come up with poor quality transcription file and incomplete topic name list.

4 Conclusion and Future Work

In this work we have segmented and annotated MOOC lecture videos for content searching inside the video. For segmentation we have applied modern neural network model on speech transcripts which improves segmentation accuracy over the existing approaches. Also for annotating a segment we have used both slide title and syllabus information that significantly improves the overall accuracy of our system. We conducted a comparative evaluation of our approach against three state of the art methods. The result empirically proves the strength of our approach. In future we have a plan to modify the annotation methods and evaluate our system on more challenging dataset.


References

1. National program on technology enhanced learning (2019). <https://nptel.ac.in/>
2. Tesseract open source ocr, engine (2019). <https://github.com/tesseract-ocr/>
3. Wikipedia (2019). <https://en.wikipedia.org/>
4. Badjatiya, P., Kurisinkel, L.J., Gupta, M., Varma, V.: Attention-based neural text segmentation. In: Pasi, G., Piwowarski, B., Azzopardi, L., Hanbury, A. (eds.) ECIR 2018. LNCS, vol. 10772, pp. 180–193. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-76941-7_14
5. Baidya, E., Goel, S.: Lecturekhoj: automatic tagging and semantic segmentation of online lecture videos. In: 2014 Seventh International Conference on Contemporary Computing (IC3), pp. 37–43. IEEE (2014)
6. Bhatt, C.A., et al.: Multi-factor segmentation for topic visualization and recommendation: the must-vis system. In: Proceedings of the 21st ACM international conference on Multimedia, pp. 365–368. ACM (2013)
7. Bowman, S.R., Angeli, G., Potts, C., Manning, C.D.: A large annotated corpus for learning natural language inference. arXiv preprint [arXiv:1508.05326](https://arxiv.org/abs/1508.05326) (2015)
8. Che, X., Yang, H., Meinel, C.: Lecture video segmentation by automatically analyzing the synchronized slides. In: Proceedings of the 21st ACM International Conference on Multimedia, pp. 345–348. ACM (2013)
9. Conneau, A., Kiela, D., Schwenk, H., Barrault, L., Bordes, A.: Supervised learning of universal sentence representations from natural language inference data. arXiv preprint [arXiv:1705.02364](https://arxiv.org/abs/1705.02364) (2017)
10. Cortes, C., Vapnik, V.: Support-vector networks. *Mach. Learn.* **20**(3), 273–297 (1995)

11. Epshtein, B., Ofek, E., Wexler, Y.: Detecting text in natural scenes with stroke width transform. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2963–2970. IEEE (2010)
12. Galanopoulos, D., Mezaris, V.: Temporal lecture video fragmentation using word embeddings. In: Kompatsiaris, I., Huet, B., Mezaris, V., Gurrin, C., Cheng, W.-H., Vrochidis, S. (eds.) MMM 2019. LNCS, vol. 11296, pp. 254–265. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-05716-9_21
13. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press, Cambridge (2016)
14. Hearst, M.A.: Texttiling: segmenting text into multi-paragraph subtopic passages. *Comput. Linguist.* **23**(1), 33–64 (1997)
15. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
16. Kusner, M., Sun, Y., Kolkin, N., Weinberger, K.: From word embeddings to document distances. In: International Conference on Machine Learning, pp. 957–966 (2015)
17. Lin, M., Nunamaker Jr, J.F., Chau, M., Chen, H.: Segmentation of lecture videos based on text: a method combining multiple linguistic features. In: Null, p. 10003c. IEEE (2004)
18. Shah, R.R., Yu, Y., Shaikh, A.D., Tang, S., Zimmermann, R.: Atlas: automatic temporal segmentation and annotation of lecture videos based on modelling transition time. In: Proceedings of the 22nd ACM International Conference on Multimedia, pp. 209–212. ACM (2014)
19. Shah, R.R., Yu, Y., Shaikh, A.D., Zimmermann, R.: Trace: linguistic-based approach for automatic lecture video segmentation leveraging wikipedia texts. In: 2015 IEEE International Symposium on Multimedia (ISM), pp. 217–220. IEEE (2015)
20. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826 (2016)
21. Yang, H., Meinel, C.: Content based lecture video retrieval using speech and video text information. *IEEE Trans. Learn. Technol.* **1**(2), 142–154 (2014)
22. Yang, H., Siebert, M., Luhne, P., Sack, H., Meinel, C.: Automatic lecture video indexing using video OCR technology. In: 2011 IEEE International Symposium on Multimedia (ISM), pp. 111–116. IEEE (2011)
23. Zeiler, M.D.: Adadelata: an adaptive learning rate method. arXiv preprint [arXiv:1212.5701](https://arxiv.org/abs/1212.5701) (2012)



Play, Pause, and Skip: Touch Screen Media Player Interaction on the Move

Nicholas Vanderschantz^(✉)  and Zhiqin Yang

Computer Science Department, University of Waikato, Hamilton, New Zealand
vtwoz@waikato.ac.nz, zy71@students.waikato.ac.nz

Abstract. Our active lifestyles see us playing, pausing, and skipping through life all the while our phones are in our hands. For many, completing the daily grind requires regular audio and visual media accompaniment and for this we interact with our phones as we skip, run, and jump. In this respect, a unique form of digital library is our mobile media player of choice. These media players serve as both the interface for listening to and watching these audio and visual media as well as the media library and storage. We argue therefore that the interface design considerations of the media library as well as the media interaction require user centered investigation. We tested button placement variations and analyzed the user preferences as well as user interaction with these mobile media player prototypes while on the move. Early insights suggest users prefer what they are most accustomed to, yet issues of accuracy with interface designs that are unfamiliar require further investigation.

Keywords: User interaction · Touch screen · Media player · Mobile design

1 Introduction

There are few parts of society that the smartphone is not found, and our daily commute and regular exercise are likely no exception. When on the move we are often found tapping, scrolling, texting, checking email, and listening to our favorite tune. This common use of mobile phones for music playback is undeniable, with evidence of substantial decrease of dedicated portable media player sales for some time now [7]. Punchoojit and Hongwarittorn [12] recently completed a broad survey of usability studies of mobile user interface design and have concluded that further empirical evidence is still required to guide designers and developers in creating efficient and effective interactive touch screen interfaces for use while users are moving. We present one of the few studies into mobile use of media player interfaces on touchscreen devices and show that there is need for further investigation into button placement for users in motion.

2 Related Work

Investigations of touch screen interaction has explored a range of devices and screen sizes as well as sizing, spacing, and placement of GUI features. Few investigations have considered the impact of interface design on device use during movement [i.e., 3].

Of the recent digital library interface design research for mobile touch screen use, the majority of this work has focused on user preferences regarding the display of information [i.e., 16] rather than user interaction with particular media. We focus the discussion of the related work on research relating to design and implementation of interactive touch screen features of smartphone applications.

2.1 Button Size and Spacing

Generally accepted target size recommendations for single click buttons range from 8 mm to 14 mm. Parhi et al. [10] found that touch target sizes larger than 9.6 mm showed the lowest error rates. Park and Han [11] studied one-handed thumb interaction concluding that a 10 mm wide button provided the best usability. Tao et al. [14] found that users preferred and were more efficient using 17.5 mm² buttons during text and numeric input.

Of the few investigations of mobile interface use during movement, Conradi et al. [3] investigated optimal button sizes for use while users are walking. Users were able to use the 8 mm sized buttons, but they made more mistakes than with 14 mm buttons. This relates to the findings of Chourasia et al. [5] who found that users made more mistakes during number entry using smaller buttons when standing than when sitting.

Unsurprisingly when the spacing of buttons is larger user interaction times are increased. Jin et al. [6] investigated button size and padding on tablet sized touch screens with consideration of reaction time and accuracy and recommend button spacing of between 3.17 mm and 12.7 mm. Similarly, Sun et al. [13] found that their users preferred a spacing of 6.35 mm to 12.7 mm.

2.2 Button Location

Much of the button location research focusses on supporting users holding and using a device with the same hand. Park and Han [11] proposed a way to analyse and describe location of elements on a mobile screen by dividing the device screen into twenty-five areas. When a user held the device with their right hand and interacted with their thumb, Park & Han recommended placing interactive items in the centre of a touch screen because these locations were faster to interact with and preferred by users of their study. Buschek [2] also investigated thumb interaction, and tested prototype UI interaction methods for common mobile GUIs. From their UI experiments they recommend that designers reconsider default locations and presentations of typical GUI features. Karlson et al. [9] recommended placing points of interaction centrally to accommodate both left- and right-handed users.

2.3 Holding and Using a Device

Hooper [8] showed that only 15.8% of the users used their left hand to touch the screen. Cornelia [4] divided her results into when participants were sitting, standing or walking and confirmed that most of the participants used their right hand to touch the screen irrespective of what they were doing at the time. Users are known to change their hold for certain interactions and research shows some hand holds are more successful than

others for text entry on mobile touchscreen devices [1]. Reinforcing the importance of button layout and location on a touchscreen device Trudeau et al. [15] showed that different button positions require users to adjust the way that they hold their device.

3 Case Study

We conducted a case study audit of the typical design and placement of play/pause, last track and next track buttons in iOS and Android music player apps. To identify appropriate music player apps to study a Google search (“most popular music player applications”) was performed. Apps common to both device platform as featured in the results on the first page of this Google search were selected. For this case study we reviewed Spotify, SoundCloud, YouTube Music, and Google Play Music, and excluded apps solely for music discovery, karaoke, or available only on a single platform.

All apps were audited on an iOS iPhone 6s and an Android Nexus E960 which have similar physical dimensions.

3.1 Case Study Results

Given the limits of this paper and the nature of this explorative case study we report a summary of our findings which were used to guide the development of our user study.

Button Design: Button iconography for media players has roots in physical media player buttons and therefore follows expected traditions of the highly recognised icons of a single right facing triangle for play buttons, two vertical bars for pause buttons, and either two interlocked triangles or a triangle and vertical bar for last track and next track buttons. For all of the apps reviewed, only play/pause icons were encapsulated in a shape—always a circle.

Button Size: Only the identifiable touch area or containment shape, or the icon itself was measured. The most typical size of play and pause buttons with a circle were 8.8 mm to 13 mm and 3.9 mm to 6.8 mm buttons without. Last and next track buttons were never contained within a shape and the icon sizes measured between 2 mm to 4 mm.

Buttons Layout and Spacing: In all instances the play/pause button was placed in the middle of the three buttons with the last track button on the left and the next track button on the right—a horizontal layout. Button spacing was always consistent between these three buttons and ranged from 4.5 mm to 22.6 mm between the visible edges.

Button Location: To ascertain commonalities of button location we took guidance from Park and Hans [11] method of dividing the device screen into twenty-five areas. The findings indicate that interface areas were most likely to be placed at approximately the vertical-centre of the screen or below.

A typical example of the button design, size, layout, and spacing that we observed in this case study is portrayed in Fig. 1.

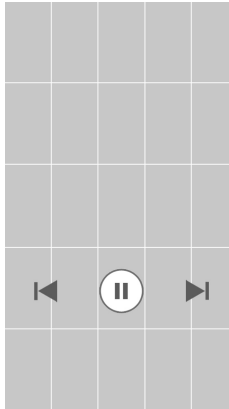


Fig. 1. Typical button design, sizing, and layout for the four interfaces reviewed across the two operating systems (eight interfaces total reviewed)

A representation of the button locations of the eight interfaces reviewed is provided in Fig. 2. We use the Park and Han [11] method of representing a screen as 25 divisions to assist with this visualisation.

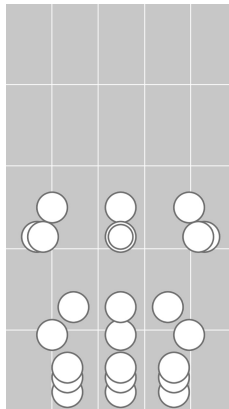


Fig. 2. Representation of observed button locations for the four interfaces reviewed across the two operating systems (eight interfaces total reviewed)

4 User Observation Study

We followed our case study with a user observation study of 30 participants interacting with six media player interface prototypes while walking. We conducted an observed interaction study with in-situ interview questions after the use of each test condition.

4.1 User Study Method

We designed and implemented six media-player interface prototypes using Proto.io (Proto.io, 2018). The prototypes were explored by 15 female and 15 male participants using an iPhone 6s. Ethical approval was received by the university to conduct this research and all participants consented to participation before the study was conducted. Participants followed verbal instructions for the use of the six prototype test conditions. Field notes and audio recordings were taken during the observation and interview.

User Study Test Materials

We used our findings from the related work (see Sect. 2) and our own case study (see Sect. 3) to assist with the design and implementation of six media-player interface prototypes. This section describes the design considerations used when developing the six test conditions tested in this study.

The location and orientation of buttons are the two variables that were of interest for this study. For this reason, we controlled other interface variables with all test conditions using the same button designs, sizes, and spacings. We tested three horizontal conditions and three vertical conditions (see Table 1). This allowed for testing preferences and successful use of elements in horizontal and vertical layout of interactive elements.

Table 1. Button locations in six test conditions

Label	Test condition	Screen location, button orientation
TC1	Test condition 1	Centre location, horizontal orientation
TC2	Test condition 2	Middle bottom location, horizontal orientation
TC3	Test condition 3	Left bottom location, horizontal orientation
TC4	Test condition 4	Centre location, vertical orientation
TC5	Test condition 5	Middle bottom location, vertical orientation
TC6	Test condition 6	Left bottom location, vertical orientation

In line with recommendations and observations we designed and implemented play, pause, last track and next track buttons using typical iconography to ensure visibility and recognition. We implemented buttons that had an 11 mm diameter circle denoting the interactive tap space for each of the buttons. Due to the use of a containment shape for our buttons and in-line with recommendations we have spaced the buttons 3.5 mm from edge to edge of the adjacent button. This resulted in buttons with a center-to-center spacing of 14.5 mm.

Our interface test materials can be found in Appendix A.

User Study Procedure

Observations: sessions begun with the participant and researcher sat at a table. Following discussion of the procedure and participant consent, the researcher asked the participant to pick up a phone placed in front of them. The participants were then asked to stand and walk along a foyer and follow verbal instructions for use of the prototype given by the researcher. Each participant used all six interface conditions (see Table 1

and Appendix A). To avoid effects of order-bias, we divided the 30 users across six intervention orders.

The participants were observed by the researcher as they interacted with the prototype on the phone. The researcher sat with the user during the initial phase of the observation, and stood and walked with the user during the movement phase of the experiment. If the participants successfully tapped a button, the interface presented a success page, while inaccuracy resulted in the user needing to tap again.

Interviews: after interacting with an interface, the user was asked an open-ended perception question regarding ease of use. Before being presented with the next test condition, each user was also asked to rank the interface on a 5-point Likert Scale from very easy to very hard. Following the use of all six test conditions the users were also asked which condition was most and least comfortable to use.

4.2 User Observation Study Results

Here we report the results of our study with 30 participants aged 20 to 60 years old. While we had four participants who were over the age of 30, the remaining participants were aged between 20 and 29 years old and therefore we do not attempt to infer age related statistical insights.

Device Pickup and Hand Switching

Of the 30 participants, 20 picked up the device with their right hand, two picked up the device with their left hand. The remaining eight participants (5 female, 3 male) picked up the device with both hands.

Six participants switched their hands when they were instructed to click the first button in the first test condition they were given. Three participants switched hands in order to tap the first requested button on TC1, (center location, horizontal orientation). For each of TC6 (left bottom location, vertical orientation), TC5 (middle bottom location, vertical orientation) and TC2 (middle bottom location, horizontal orientation), one participant switched hands in order to tap the first button.

Interaction

When interacting with the prototypes the 30 participants, tapped three buttons in each test condition, therefore each test condition received a possible 90 interactions, and thus we observed a total of 540 interactions.

414 taps were made by a right thumb or right finger compared to 126 made by a left thumb or finger. 372 single-handed interactions were made compared to 168 two-handed interactions. Overall use of two-handed operation was higher for horizontal orientations (91 taps) than vertical orientations (77 taps). Similarly, total right-handed operation was higher for horizontal orientations (214 taps using a right thumb or right finger) compared to vertical orientations (200 taps).

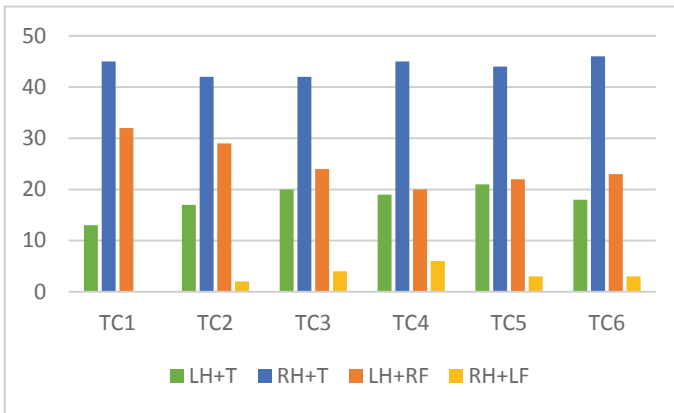
Hand Holds

Four methods of holding the device while interacting were identified (see Table 2).

Table 2. Four methods of interacting with the device

Label	Hand hold	Description of hold
LH+T	Single-handed	Device in left hand tap using left-hand thumb
RH+T	Single-handed	Device in right hand tap using right-hand thumb
LH+RF	Two-handed	Device in left hand tap using right-hand finger
RH+LF	Two-handed	Device in right hand tap using left-hand finger

The most used method of interacting with the prototypes was by holding the device in the right hand and using the right-hand thumb (RH+T) a single-handed interaction. The second most used method was the two-handed interaction where the user held the device with the left hand and tapped using a finger on the right hand (LH+RF). The third most used method was the single-handed LH+T with very few participants using RH+LF. Figure 3 shows aggregated results of the method of interaction for each tap made by the users across each of the test conditions.

**Fig. 3.** Aggregated results of interaction method made by the users across each test condition

Adjustment During Interaction

We report here further detail regarding users who adjusted their interaction method while using a particular test condition. Figure 4 shows the changes that occurred; for example, P4 used her left hand to tap both the play button and the next button on the horizontal TC1 but used her index finger on her right hand to touch the last track button which would have required her to bend her thumb uncomfortably. P4 also used her left hand and thumb to interact with the vertical TC6, swapped hands to use her right hand and right thumb for the next track button and then swapped back to her left hand and left thumb for the last track button.

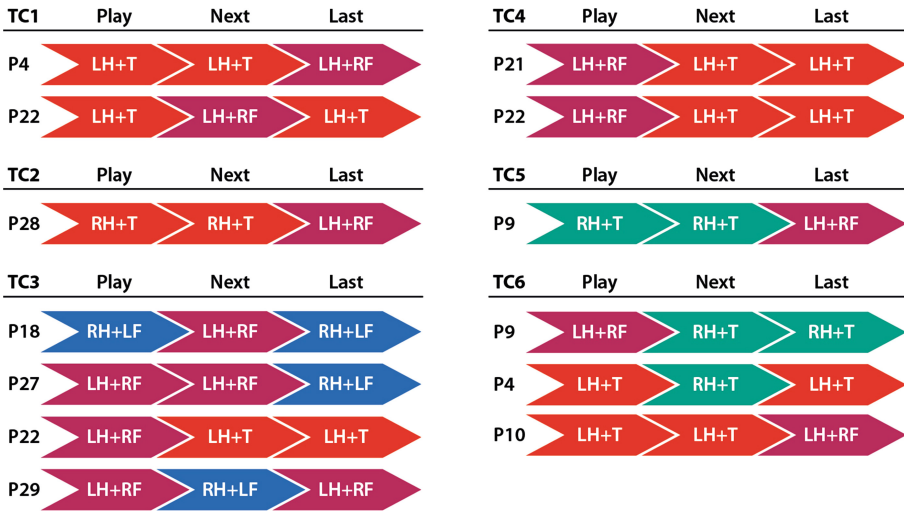


Fig. 4. Representation of the ways participants adjusted their hand holds during interaction

In total nine of 30 participants needed to adjust their hold during their interactions with one of the test conditions. Two of those nine participants adjusted their hand hold twice and one participant adjusted hand hold three times. Seven hand hold adjustments occurred when using a horizontal condition (see left-hand column of Fig. 4), while six adjustments were needed while using a vertical condition (right-hand column of Fig. 4).

Accuracy

In our study, none of the participants tapped the wrong button and only seven of the participants did not successfully click their intended button the first time they attempted to. That is to say, that of the 540 instructions to tap, only seven inaccurate or unsuccessful attempts to tap were recorded. Inaccuracies were self-identified by participants who directly after a miss-tap, corrected their aim and tapped a second time on the target button they had missed with their first attempt.

We recorded three inaccuracies in horizontal test conditions and four inaccuracies in vertical test conditions. The buttons that caused issues for users were TC1 play and last track (1 inaccuracy each), TC3 play (1 inaccuracy) TC4 play (2 inaccuracies) and next track (1 inaccuracy), and TC6 next track (1 inaccuracy).

In the horizontal conditions the inaccuracies were for users holding with their left-hand. Two of these inaccuracies were for two handed operation, while one was for left-hand and left-thumb interaction. In the vertical conditions the inaccuracies were for single-handed operation—three right-hand and right-thumb interactions, and a single left-hand and left-thumb interaction.

Perceptions of Ease

After a participant finished interacting with a test condition we asked our two questions regarding preference. Our first question asked “Was the interface easy to use?” “Why?” More participants reported that the horizontal conditions were easy to use than reported the vertical conditions were easy to use (see Fig. 5). However, in all cases less than half of the participants stated that a condition was not easy to use.

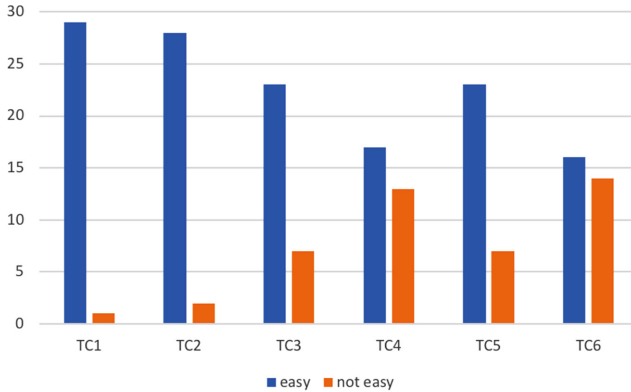


Fig. 5. Perception of ease for each interface

Our second question was a request for participants to rank the interface they had just worked with on a 5-point Likert scale from “very easy” to “very hard”. Figure 6 shows that very few participants labelled any of the six test conditions as “very hard” or “hard”. The test conditions that were thought to be hard included horizontal and vertical conditions, while the vertical conditions were the only conditions to be labeled as very hard when scored on the Likert scale. The numbers are very low in both instances.

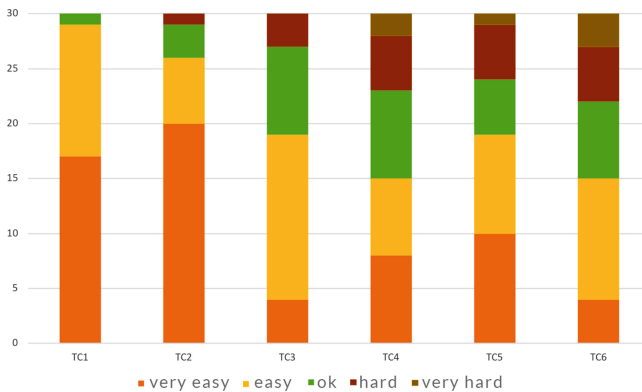


Fig. 6. Likert Scale perception of ease

When we asked participants to qualify their answers, it was common for users to report that a TC was “easy to use”, a button was “easy to reach”, or they were familiar with a TC and this impacted their perception of ease. Similarly, common responses regarding why a TC was considered hard to use was because a button was “too far away to reach”, or they were unfamiliar with a TC, or felt that aesthetically the interface looked “unusual”, or “odd”.

Reasons for both positive and negative responses were similar across the test conditions. For example, TC1 and TC4 both contain buttons located in horizontally and vertically central positions, and each received feedback regarding the central location of buttons being a positive feature.

Very few participants ranked vertical layouts as hard or very hard and this is reinforced in the qualitative feedback received. Vertical conditions often received feedback that suggested the participants were unsure about this layout. Participants reported that vertical conditions were unusual, and felt that they would require some getting used to. Participants used the word “odd” or “unusual” to describe this condition, and also noted for example “I’m not familiar with vertical placement but it is ok for using”. One participant reported that they felt the vertical layout caused fewer mistakes for them when they were interacting with these interfaces.

5 Discussion

It is commonly held that interactive features on touch screen devices should fall in the lower portion and be centred horizontally in the screen [9, 11]. Our analysis of popular music player interfaces, reveals that the most commonly used location for play/pause, last track, and next track buttons follows this belief.

Unique to our study was the inclusion of the non-typical vertical orientation of interface buttons. Vertical orientation resulted in higher left-handed operation and higher single-handed operation suggesting that further investigation is warranted. Confirming the findings of [4, 8], our user study showed the most common interaction in both a horizontal and vertical layout was right handed holding of the phone and interaction with the right thumb. The second most common interaction method in our study was a right finger interaction while the user held the phone with their left hand. This suggests that interfaces that consider right thumb or right index finger interaction will support the most users when they are moving.

Tao et al. [14] recommend considering both subjective and objective measures when developing touchscreen design guidelines. The subjective feedback regarding these vertical orientations was less positive than for the horizontal orientations. When questioned participants more often reported a preference for a horizontal button placement in touch screen mobile media player interfaces. This is almost certainly because the existing media player interfaces on mobile devices are typically arranged horizontally. It is therefore not unexpected that numerous participants reported that the vertical presentations were something they had little experience with and would require

some getting used to. Given that very few participants ranked these vertical interfaces as very hard while other participants reported that the unusual interface would require further time to become comfortable may suggest that a study which allowed users more time to interact with the interfaces may alter the results of our study presented here.

5.1 Limitations and Future Work

Our exploratory first study reported here sought to control a number of variables in order to identify potential interacting factors as well as opportunities for further investigation. With this in mind, this study controlled for location with all participants using the mobile device in the same physical location and beginning from the same seated position and phone placement. Future work would do well to consider varied environments such as indoor and outdoor use, navigation of entrances or exits to buildings, and inclining and declining terrain. Equally, it may prove interesting to have participants also place and remove the phone into their pocket, bag, or typical transport location as a feature of the study which may be an interaction similar to that of picking up the phone from the table for the first time. Placing the device back on the table may also prove insightful in understanding the ergonomics of common phone use.

A shortcoming of our method was that we did not ask users what their dominant hand was. This is an oversight when we find that total right-handed operation and two-handed interaction was higher in the horizontal orientations than the vertical orientations. These two results lead us to suggest that investigation of how orientation effects the need for a user to use their dominant hand is warranted.

Our exploratory study did not consider measures of accuracy related to interaction time, button size, or button design. The low number of inaccuracies in the present study could well be a feature of the design decision to implement button sizing, button iconography, and spacing based within the guidelines identified by the previous work of [i.e. 3, 7, 11, 14] and our own case study observations (see Sect. 3). Our future work will incorporate tools that measure performance across these interfaces to provide deeper insights into if and how location, layout, design, and size of buttons in media players might contribute to user success while moving.

Presently we have been unable to identify digital library research that addresses interface design for users in motion. The majority of the related work we present sits within wider fields of investigation and does not specifically address interface design considerations which might be prevalent for digital libraries, especially those used by people on the move. We therefore believe that this area that we begin to address in this paper requires continued and broad work given the prevalence of personal digital libraries available today on the now ubiquitous touch screen mobile device.

6 Conclusion

Today's users are interacting with personal digital libraries on a diverse range of devices in increasingly more nuanced environments and daily situations. The varied media that digital libraries contain and the ways in which users expect to interact with

their media requires constant investigation to ensure that users are afforded the best interactive experience possible with their chosen digital media library. We report here a study of user interface interaction with media player interfaces on mobile touch screen devices. Our study provides one of the few insights into use of touchscreen media player interfaces for users in motion, be that the daily commute or a visit to the gym.

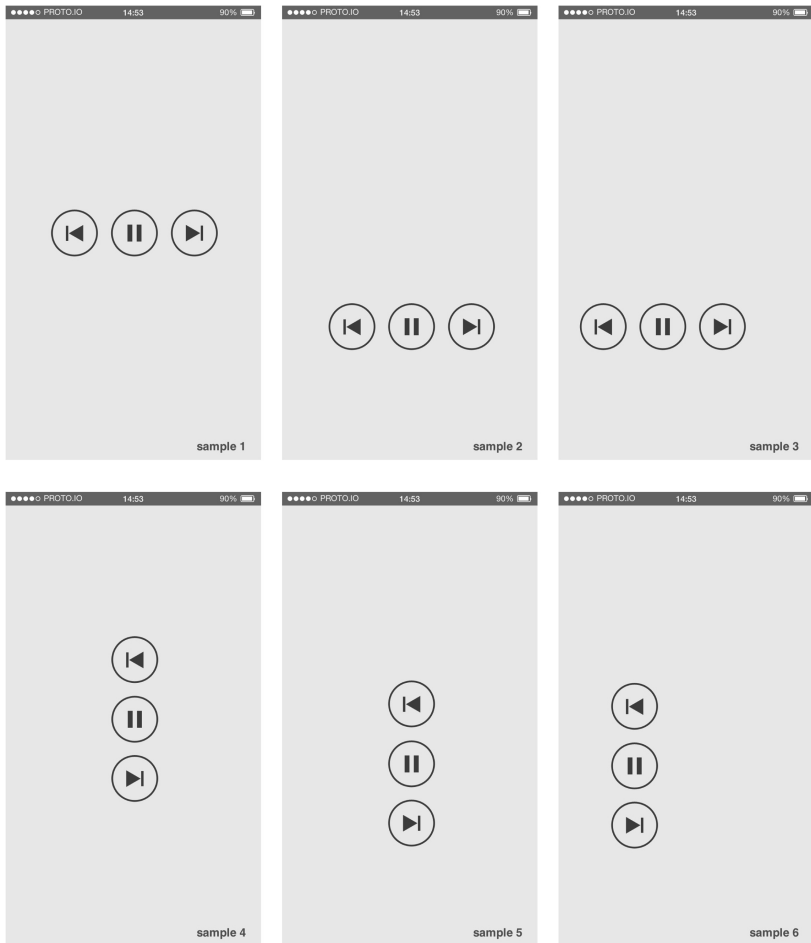
Most participants in our study picked up the device with their right hand. While interacting with a test condition, the participants were more likely to use their right hand and right thumb to interact. Considering total interactions, most participants tended to use a single hand when interacting. Given these results we recommend that designers develop mobile digital library interfaces that afford single handed interaction.

Further, vertical layout of media player buttons provided the greatest opportunity for users to use only one hand, while horizontal layout of buttons saw the greatest number of participants needing to switch hands while interacting. When asked users suggested horizontal layout of media player buttons was their preference. Our case study of the current design conventions for media players showed typical media player button placements to be horizontal and therefore we hypothesise the preferences reported are likely due to familiarity. Given our results that suggest improved efficiency with these uncommon button layouts we would recommend further investigation into media player interfaces for users in motion.

Our results and recommendations here pertain to the specific use-case of digital media player interfaces on touch screen mobile devices for users in motion. We argue that these recommendations and the requirement for future work in this area also exists more broadly in the digital library field. Personal digital libraries today encompass a variety of media, be that audio, video, still image, or multi-page text-based documents. Users are likely to interact with these various media on their personal mobile device in a range of individual and shared situations, whether they are sitting, standing, or indeed moving. We, therefore, recommend investigation of the use of digital library interfaces on mobile devices as well as design interventions that support single-handed interaction in ways that may not be the current standard for a given use-case.

Appendix A

The six test conditions developed and tested with 30 participants while they walked. TC1–TC3 (the three tested horizontal test conditions) can be seen in the top row, with the vertical test conditions, TC4–TC6 in the bottom row.



References

1. Azenkot, S., Zhai, S.: Touch behavior with different postures on soft smartphone keyboards. In: Proceedings of the 14th International Conference on Human-Computer Interaction with Mobile Devices and Services, pp. 251–260. ACM, September 2012
2. Buschek, D., Hackenschmied, M., Alt, F.: Dynamic UI adaptations for one-handed use of large mobile touchscreen devices. In: Bernhaupt, R., Dalvi, G., Joshi, A., Balkrishan, D.K., O’Neill, J., Winckler, M. (eds.) INTERACT 2017. LNCS, vol. 10515, pp. 184–201. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-67687-6_13

3. Conradi, J., Busch, O., Alexander, T.: Optimal touch button size for the use of mobile devices while walking. *Procedia Manuf.* **3**, 387–394 (2015)
4. Cornelia, L.: Do users interact with their mobile devices with their dominant hand? (2014). <https://realites-paralleles.com/>
5. Chourasia, A.O., Wiegmann, D.A., Chen, K.B., Irwin, C.B., Sesto, M.E.: Effect of sitting or standing on touch screen performance and touch characteristics. *Hum. Factors* **55**(4), 789–802 (2013)
6. Jin, Z.X., Plocher, T., Kiff, L.: Touch screen user interfaces for older adults: button size and spacing. In: Stephanidis, C. (ed.) *UAHCI 2007*. LNCS, vol. 4554, pp. 933–941. Springer, Cham (2007). https://doi.org/10.1007/978-3-540-73279-2_104
7. Hall, J.: MP3 players are dead, 26 December 2012. <https://www.businessinsider.com>
8. Hooper, S.: Design for Fingers, Touch, and People, Part 2, 8 May 2017. <http://www.uxmatters.com>
9. Karlson, A.K., Bederson, B.B., Contreras-Vidal, J.: Understanding single-handed mobile device interaction. In: *Handbook of Research on User Interface Design and Evaluation for Mobile Technology*, vol. 1, pp. 86–101 (2006)
10. Parhi, P., Karlson, A.K., Bederson, B.B.: Target size study for one-handed thumb use on small touchscreen devices. In: *Proceedings of the 8th Conference on Human-Computer Interaction with Mobile Devices and Services*, pp. 203–210. ACM, September 2006
11. Park, Y.S., Han, S.H.: Touch key design for one-handed thumb interaction with a mobile phone: effects of touch key size and touch key location. *Int. J. Ind. Ergon.* **40**(1), 68–76 (2010)
12. Punchoojit, L., Hongwarittorn, N.: Usability studies on mobile user interface design patterns: a systematic literature review. *Adv. Hum.-Comput. Interact.* (2017)
13. Sun, X., Plocher, T., Qu, W.: An empirical study on the smallest comfortable button/icon size on touch screen. In: Aykin, N. (ed.) *UI-HCII 2007*. LNCS, vol. 4559, pp. 615–621. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73287-7_71
14. Tao, D., Chen, Q., Yuan, J., Liu, S., Zhang, X., Qu, X.: Effects of key size, gap and the location of key characters on the usability of touchscreen devices in input tasks. In: Harris, D. (ed.) *EPCE 2017*. LNCS, vol. 10276, pp. 133–144. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-58475-1_10
15. Trudeau, M.B., Young, J.G., Jindrich, D.L., Dennerlein, J.T.: Thumb motor performance varies with thumb and wrist posture during single-handed mobile phone use. *J. Biomech.* **45**(14), 2349–2354 (2012)
16. Vanderschantz, N., Timpany, C., Feng, C.: Tertiary students’ preferences for library search results pages on a mobile device. In: Dobрева, M., Hinze, A., Žumer, M. (eds.) *ICADL 2018*. LNCS, vol. 11279, pp. 213–226. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-04257-8_23

Search Engines



Towards a Story-Based Search Engine for Arabic-Speaking Children

Rawan Al-Matham¹(✉), Wed Ateeq³, Reem Alotaibi¹,
Sarah Alabdulkarim², and Hend Al-Khalifa¹

¹ Information Technology Department, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia
r.almatham@gmail.com

² Department of Educational Policies, King Saud University, Riyadh, Saudi Arabia

³ Information Technology Department, College of Computer and Information Sciences, Al Imam Mohammad Ibn Saud Islamic University, Riyadh, Saudi Arabia

Abstract. This paper presents بُكُور Bukoor, an experimental Arabic children stories search engine built to help children formulate their queries. It utilizes two main query reformulation techniques: query expansion and query suggestions. Both are developed using test collection bi-gram collocation terms and corrections' query terms generated from Bukoor test collection. It also uses two indexing techniques: Okapi BM25 and Latent Semantic Indexing. Our proposed approach was evaluated using Bukoor children's stories test collection which showed promising results when using our proposed query expansion with Okapi BM25 indexing.

Keywords: Bi-gram collocations · Query boosting · Query expansion · Search engine · Arabic language · Information retrieval · Query corrections · Children

1 Introduction

Web search engines are information retrieval systems used to search for information on the World Wide Web. Most well-known web search engines are oriented to serve the information needs of general audience that are usually adults [1]. However, every specific type of user has different search approaches, cognitive skills, and information needs [2]. For example, children need specific content, have limited vocabulary, and face difficulties in choosing the right keywords to write an expressive and an effective query [3]. In addition, children commit much more misspelled queries than adults [4]. Those characteristics for children's queries lead to mismatch between terminologies used in the query and document contents [5]. Hence, there are some safe search engines developed to provide suitable content for children such as kidRex by Google [6]. Also, some recent studies emerged [1, 7] that utilized query suggestion technique to help children in formulating their information needs.

However, to the best of our knowledge, we have noticed that most of the previous research efforts in enhancing children search engines were implemented for English speakers [1, 7] and none for Arabic. Even research for English speakers is still limited. Therefore, the problem we are trying to solve is to help Arabic-speaking children in formulating proper search queries in order to retrieve the related and satisfactory results for their information needs.

This research aims to answer the following questions: (1) How does the use of bi-gram collocations and corrections for query expansion affect the retrieval of Arabic children's stories? (2) How is the performance of the proposed method affected by using BM25 compared to LSI? To answer these questions, we built a test collection called "Bukoor" that contains children stories and queries. Additionally, we used the Lucene and Semanticvectors Java open-source library to develop and evaluate the Bukoor search engine using Bukoor test collection.

The rest of the paper is structured as follows: Sect. 2 discusses the proposed solution, including Bukoor's proposed architecture, and Bukoor's test collection; Sect. 3 presents the experimental evaluation including the obtained results and the analysis and discussion; and finally, Sect. 4 presents the conclusion and future directions.

2 Proposed System

Bukoor search engine is a children story-based search engine that is developed in order to improve children's search tasks. It targets children aged 8 to 12 years old because they start to read and write independently in this age range [8]. It also helps children to formulate their queries by providing them with auto completion and query suggestions. In addition, it has a query expansion feature that uses the query correction terms and bi-gram collocations to enhance documents retrieval. The idea of query expansion using bi-gram collocations was inspired from Moawad et al. study [9], since they proved the effectiveness of using bi-gram collocations for query expansion to retrieve Holy Quran documents. In this section, we present Bukoor test collection, Bukoor system architecture, and its mechanisms.

2.1 Test Collection

Bukoor collection contains 218 documents consisting of 75,946 words, 21,555 vocabularies and 112 queries. The stories are classified into 14 topics and each topic has 13 to 20 stories. The topics represent moral subjects. They were collected from the Web using some related keywords that were searched on Google search engine. Table 1 presents the number of stories for each topic.

Table 1. Test collection topics

Code	Topic	Total Stories	
T1	القناعة	Contentment	15
T2	الصدقة	Charity	20
T3	التعاون	Cooperate	16
T4	الصدق	Honesty	15
T5	المحبة	Love	15
T6	التواضع	Modesty	15
T7	الذكاء	Intelligence	15
T8	الشجاعة	Courage	16
T9	الأمانة	Honesty	15
T10	العدل	Justice	13
T11	النظافة	Cleanliness	15
T12	بر الوالدين	Honoring parents	17
T13	الرحمة	Mercy	13
T14	الصداقة	Friendship	18

Table 2. Sample of error patterns

#	Pattern	Description
1	ة < ت	Writing (ة) as (ت) and via versa.
2	ض < ظ	Writing (ض) as (ظ) and via versa.
3	و < ُ	Writing (و) as (ُ) and via versa.
4	الش < اش	Dropping the non-pronounced (ل) in (ال).

In order to collect children's queries, we followed Kalsbeek et al. [10] approach. They constructed a list of children's possible writing errors with the help of primary school teachers. Therefore, we collected the children's misspelling patterns from two primary school teachers, and we removed the duplication. Table 2 presents a sample of the collected errors' patterns.

Additionally, we asked a specialist in childhood writing characteristics to write a list of five queries for each topic and three misspelled variations for each query using our list of children writing errors. Then the relevance judgments were conducted by the authors and four queries with their variations were selected for each topic by agreement. The final queries list contains 112 queries, 56 queries with correct spelling and 56 misspelled queries with different errors types. Table 3 shows examples of the final queries. The collected queries were divided into two subsets as follows:

1. Training set: contains 56 queries, 28 correct queries and 28 misspelled queries.
2. Testing set: contains 56 queries, 28 correct queries and 28 misspelled queries.

Table 3. Collected queries sample

Topic	Query	Misspelled Query
الصدقة Charity	المساعدة والصدقة " Assistance and charity "	المساعدة والصدقت
التعاون Cooperate	أساعد الآخرين " I help others"	اساعيد الآخرين
الأمانة Honesty	خلق الأمانة عظيم The moral of Honesty eminent	خلق الأمانة عظيم

2.2 Bukoor's Architecture

The architecture of Bukoor proposed system and the overall search mechanism are illustrated in Fig. 1. Our implementation is composed of two phases as follows:

Offline Phase. In this phase, two thesauri and three indexes are generated as follows:

1. Suggestions thesaurus generation by performing word tokenization on the documents.
2. Bi-gram collocation thesaurus generation from Bukoor test collection that will be used for expansion and suggestions.
3. Documents preprocessing by performing word tokenization, normalization, stop words removal and light stemming.
4. Okapi BM25 and VSM indexes creation.
5. LSI index creation by using SVD algorithm.

Online Phase. Children's queries are processed to retrieve most relevant documents during this phase. It will start when the child starts to write his/her query as follows:

1. Providing an auto completion feature while the query is being written.
2. Checking the query and suggestion thesaurus to provide similar terms list in case the query is misspelled.
3. Generating query suggestions from bi-gram collocations thesaurus for the original query and its similar terms.
4. Expanding the original query by its correction terms, then by its collocations from bi-gram collocations thesaurus.
5. Performing the preprocessing steps for the submitted query as what have been done previously to the test collection during the offline phase.
6. Retrieving relevant documents for the expanded query and ranking them based on the used model either Okapi BM25 or LSI.

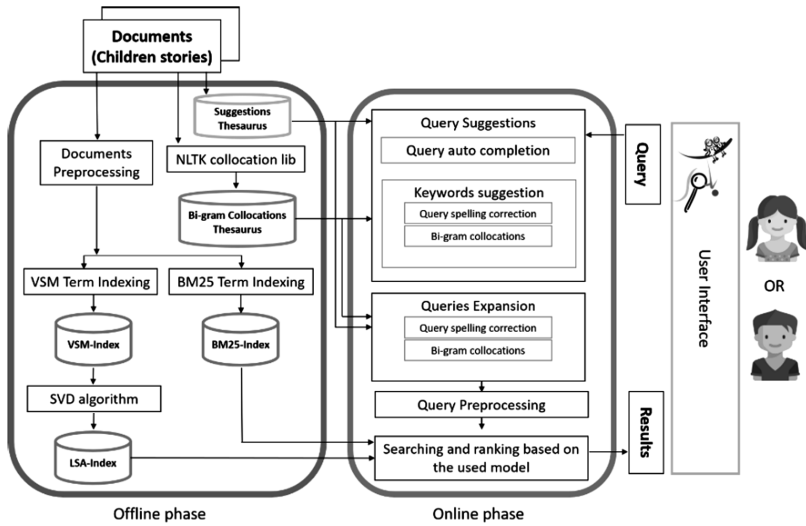


Fig. 1. Bukoor system architecture

3 Experiments and Results

To evaluate the effectiveness of the proposed approach for query expansion, we designed three different experiments using different variations of expansion terms: (1) query expansion using correction terms, (2) query expansion using bi-gram collocations terms, and (3) query expansion using correction terms and bi-gram collocations terms. We applied two runs for each experiment. The first run was without using query boosting; consequently, all expanded query terms had their normal weight based on the used weighting model. In the second run, we applied query boosting for the expanded query terms. Table 4 presents each conducted experiment with its aim.

Table 4. Experiments' aims and used expansion terms.

	Aim	Expansion terms
Experiment (1)	Examine the effect of using only the <i>corrections</i> for query Expansion	Corrections
Experiment (2)	Examine the effect of using only the <i>bi-gram collocation</i> for query Expansion	Bi-gram collocation
Experiment (3)	Examine the effect of using the <i>corrections</i> terms and the <i>bi-gram collocation</i> together for query Expansion	Corrections terms Bi-gram collocation

All experiments were performed on Bukoor test collection. In the following sections, we describe the used baseline, the selected measurements for evaluation, and the obtained results from each experiment.

3.1 Baseline

To evaluate the effectiveness of Bukoor query expansion feature, we obtained our baseline using Lucene's Okapi BM25 and Semanticvectors' LSI model. Some pre-processing steps were applied by using Lucene's Arabic analyzer to the dataset and queries.

We searched about children's behavior to choose suitable evaluation measures and found that children tend to click on higher ranked results [11]. Therefore, we care about the ranking of the retrieved results; hence we chose two evaluation measures. First: Precision, which is calculated at rank 1 to check the first ranked result for each system. Second: Mean Average Precision (MAP), which was calculated twice, the first one was for overall queries to measure the overall performance, second it is calculated @ rank 3, since we want to examine the high ranked results that tend to be chosen by the child. The calculations are done using the following formulas:

$$Precision = \frac{\text{relevant documents} \cap \text{retrieved documents}}{\text{retrieved documents}} \quad (1)$$

$$MAP = \frac{\sum_{i=1}^{|Q|} Avg(q_i)}{|Q|} \quad (2)$$

Where Q represents the set of queries and q_i is a query of Q . The best used indexing model in the baseline run was LSI in terms of P@1 and MAP measurements; while BM25 was the best in terms of MAP@3.

3.2 Experiments' Results

We conducted many experiments on Bukoor collection using different types of expansion terms: the correction terms only, the bi-gram collocation terms only, and both. Each experiment with and without query boosting. Table 5 shows a comparison between the percentage of improvements and decay for each experiment compared to the baseline. Overall, the previous experiments have showed that Bukoor's proposed approach has proved its effectiveness in retrieving children's Arabic stories which answers our first research question. To answer our second research question, it appears that BM25 implementation showed an improvement equal to 59.15%. This result has outperformed the performance of LSI implementation. LSI indexing with the query expansion showed little improvement when using corrections terms only or bi-gram collocations only. However, it was the worst when using them together since it decreases the retrieval by -10.63%. We think that happened because LSI creates a semantic layer for different topics and the expanded query that uses the corrections set and bigram collocation sets has many terms. Those terms came from different topics; therefore, it cannot retrieve matching results for all of them. Whereas when using corrections only or bi-gram collocations only, the number of terms became smaller than when they have used them together. Therefore, our query expansion proposed approach was not suitable to use with LSI model.

Table 5. Comparison between the percentage of improvement and decay in terms of MAP for each used model.

The used expansion terms	BM25	LSI
Corrected terms and bi-gram collocations terms	59.15%	-10.36%
Bi-gram collocation terms	36.79%	2.86%
Corrections terms	30.72%	5%

4 Conclusion and Future Work

This paper presented بُكُور (Bukoor), an experimental Arabic children story-based search engine that is developed to improve children's search tasks. It is built based on Lucence and Semantic vectors which are open source libraries. We used query boosting technique to choose the right weight for each expansion term. In addition, two indexing techniques were used to build and evaluate Bukoor: BM25 and Latent Semantic Indexing.

As a future work, we plan to evaluate the query suggestion feature. In addition, we plan to find more accurate error detection techniques to detect the misspelled queries. We will also try using other association measures for the bi-gram collocations such as logDice to enhance their quality. Moreover, we are considering applying our methodology to serve children in general search engine purpose. Lastly, we want to try our methodology with LSI indexing over BM25 model.

References

1. Madrazo Azpiazu, I., Dragovic, N., Anuyah, O., Pera, M.S.: Looking for the movie Seven or Sven from the movie frozen?: a multi-perspective strategy for recommending queries for children. In: Proceedings of the 2018 Conference on Human Information Interaction & Retrieval, pp. 92–101. ACM, New York (2018). <https://doi.org/10.1145/3176349.3176379>
2. Foss, E., et al.: Children's search roles at home: implications for designers, researchers, educators, and parents. *J. Am. Soc. Inf. Sci. Technol.* **63**, 558–573 (2012)
3. Duarte Torres, S., Weber, I., Hiemstra, D.: Analysis of search and browsing behavior of young users on the web. *ACM Trans. Web (TWEB)* **8**, 7 (2014)
4. Duarte Torres, S., Hiemstra, D., Serdyukov, P.: An analysis of queries intended to search information for children. In: Proceedings of the Third Symposium on Information Interaction in Context, pp. 235–244. ACM (2010)
5. Billerbeck, B., Zobel, J.: Questioning query expansion: an examination of behaviour and parameters. Presented at the 1 January 2004
6. KidRex - Kid Safe Search Engine. <http://www.kidrex.org/>
7. Wood, A., Ng, Y.-K.: Orthogonal query recommendations for children. In: Proceedings of the 18th International Conference on Information Integration and Web-based Applications and Services, pp. 298–302. ACM (2016)
8. School-Age Readers. <https://kidshealth.org/en/parents/reading-schoolage.html>

9. Moawad, I., Alromima, W., Elgohary, R.: Bi-gram term collocations-based query expansion approach for improving Arabic information retrieval. *Arab. J. Sci. Eng.* **43**, 1–14 (2018)
10. van Kalsbeek, M., de Wit, J., Trieschnigg, D., van der Vet, P., Huibers, T., Hiemstra, D.: Automatic reformulation of children’s search queries. University of Twente (Internal Report) (2010)
11. Kuhlthau, C.C.: Inside the search process: information seeking from the user’s perspective. *J. Am. Soc. Inf. Sci.* **42**, 361–371 (1991)



Metadata-Based Automatic Query Suggestion in Digital Library Using Pattern Mining

Susmita Sadhu^(✉) and Plaban Kumar Bhowmick^(✉)

Indian Institute of Technology, Kharagpur 721302, India
susmita90iitkgp@gmail.com, plaban@gmail.com

Abstract. This paper presents a Query Auto-Completion (QAC) framework that aims at assisting users in a digital library to specify their search intent with reduced effort. The proposed system suggests metadata-based facets to users as they specify their queries into the system. In this work, we model the facet-based QAC problem as frequent pattern mining problem where the system aims at leveraging association among different facet combinations. Among several frequent pattern mining algorithms, the present work make use of FP-Growth to discover facet patterns at large-scale. These facet patterns represented in form of association rules are used for online query auto-completion or suggestion. A prototype QAC augmented digital library search system is implemented by considering a limited bibliographic dataset (35K resources) of the National Digital Library of India (NDLI: <https://ndl.iitkgp.ac.in>) portal. We perform extensive experiments to measure the quality of query suggestions and QAC augmented retrieval performance. Significant improvement over baseline search system is observed in both the aspects mentioned above.

Keywords: Query suggestion · Frequent pattern mining · Query auto completion

1 Introduction

Facet-based access to the resources has been a pre-dominant feature of any digital library. In contrast to the traditional approach in many search engines, where document retrieval primarily depends on user-provided query, a digital library can assist users with a set of facets (e.g., author, subject domain, language, date of publication, etc.) to express their respective search intent. However, usage of facets have been reported to be rather sparse [1]. Furthermore, incorporation of facets by users is biased towards lower count (around 60% requests involve only one facet) [1]. Cognitive overload (in form of choice overload, visual complexity, and information overload) in using the facets, specifically for the novice users, is one of the primary factors behind this low usage [1]. As an alternative to facets, users tend to present *verbose queries* to the systems. Verbose queries tend to be

longer and resemble more to natural language sentences or phrases than simple bag-of-words. Verbose queries, though are easy to formulate, pose challenges in the form of processing difficulty and degrade retrieval performance [2]. The sources of difficulty include ambiguous or misspelled words, abbreviations etc. Due to degraded retrieval, users, in many situations, need to reformulate the base query more than once. Consequently, the need for repeated reformulation sometime brings about user frustration and poor user experience.

One way to circumvent the mentioned problem is to segment the input verbose query semantically, followed by faceted search derived out of the semantic analysis [3]. This strategy requires the entire query to be placed by the user. Interactive query formulation could be another alternative where at each step of query formation, the user is presented with suggestions to be selected as part of query. This idea is realized through query suggestion and query auto-completion that are two leading features of the modern search engines. While query suggestion presents the probable queries in the form of suggestions below the search box, the query auto-completion refines the list dynamically when users provide prefix continuously. Both the features aim to predict user intention with only a few keystrokes and help users to formulate search query with reduced effort and time.

One advantage that digital libraries have over keyword-based search engines is the extensive use of metadata to organize and index the resources. These metadata fields, often used as filters, can be appropriate candidates for query auto-completion. In this work, we envisage designing an auto-suggestive search interface, which provides facets as suggestions over the base query in a hierarchical and dynamic manner. At each level of the hierarchy, a user is presented with a list of (metadata field, value) pairs as suggestions. The suggestions at every level are generated dynamically by considering the selections made by the user in the preceding levels.

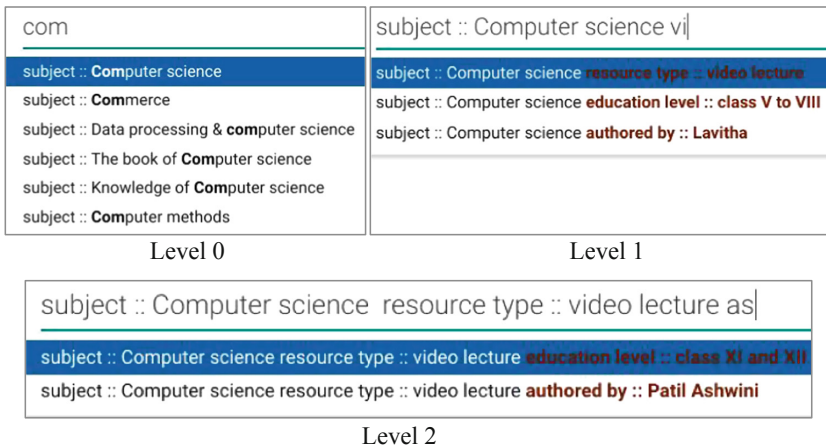


Fig. 1. Description of levels in QAC

Figure 1 depicts an example of hierarchical suggestion for a verbose query “computer science video lecture by ashwini patil”. The query session of a hypothetical user starts with a prefix “com” (level 0). Based on selection “Subject :: Computer Science” by the user, the candidates in level 1 are generated and further refined by prefix “vi” typed by the user. Finally, the list is further reduced in level 2 as the user types in the prefix “as” with an intention to specify ‘ashwini patil’. One point is worthwhile to note here is that query suggestions are drawn from the domain of metadata values. This leads to the queries that conform to the controlled vocabulary of metadata values rather than free text provided by users. This enables the system to transform the query accurately into a faceted query which ensures better retrieval.

In the past decade, many works on query suggestion have analyzed click-through information [4–6] from query logs and leveraged co-clicked URLs to identify related queries. In other work, Baeza-Yates et al. [7] has acknowledged the importance of text contained in user clicked URLs to cluster similar queries. Cao et al. [8] proposed an approach to cluster similar queries based on concepts and then by mapping user query sessions to concept sessions. According to these approaches, possible suggestions for a given query can be generated from its corresponding cluster. Some other recently proposed QAC models [9–13] rank a list of suggested queries for each prefix based on different features like words present in the query logs, user behavior during search with QAC-system and popularity, time and others. However, the metadata structure of the host repository has not been leveraged and explored in the earlier works.

In this work, we intend to generate and present query suggestions using metadata description of resources in digital libraries. The entire repository is analyzed to discover interesting associations among metadata structure of the resources. The derived associations are used to generate suggestions as ranked list which is consulted by the users to specify their search intent interactively. Towards solving the QAC problem in focus, the present work contributes in the following way:

- The metadata facet-based QAC problem is modelled as Frequent Item-set Mining problem.
- To address the scale of operation, the proposed work make use of state-of-art algorithm like FP-Growth and parallel processing framework like Apache Spark.
- An extensive human-based experiment is performed to judge the quality of suggestions and effect of suggestion in retrieval.

2 Proposed Query Auto-Completion Model

The present work follows a two-stage ‘Discover-and-Rank’ approach towards addressing metadata-based QAC problem:

- **Offline Association Generation (Discover):** Instead of using query logs, we focus to find associations among metadata information of the resources

in digital library. A popular Frequent Item-set Mining (FIM) method, FP-Growth, is used to uncover the associations among metadata fields (association rules) based on their presence in documents. Pruning of the all possible association rules are performed using two scores, namely, *minimum confidence* and *minimum support*. The derived association rules are indexed for efficient retrieval required in later stages.

- **Online Query Suggestion (Rank):** The online query generation process takes inputs from user in form of partial query (viz. prefix) and provides *ranked* suggestions in a hierarchical manner by consulting the association rules indexed in offline stage. Each suggestion is a unique ⟨metadata field, value⟩ pair. The candidate suggestions in the n^{th} level are generated based on sequence of selections made in the previous $(n - 1)$ levels. The generated candidates are then ‘ranked’ based on *confidence* scores of relevant association rules. After specification of the entire query, the system poses a filter query considering selected suggestions and presents a result list to the user.

3 Offline Association Generation

This section describes the approach to generate an index of possible association patterns of ⟨metadata field, value⟩ pairs by studying their presence in the dataset. Figure 2 presents schematics of association generation workflow. Discovering associated ⟨metadata field, value⟩ for a given large set of related metadata items (referred as a transaction) is the key problem. We formulate it as a frequent item-set mining (FIM) problem.

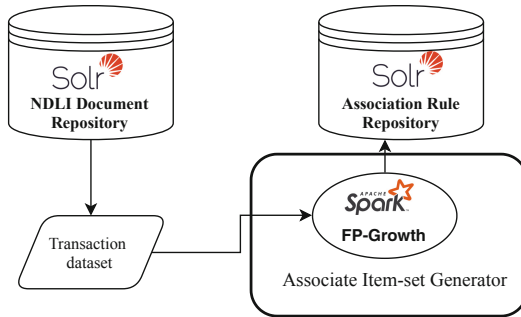


Fig. 2. Architecture of offline association rule generation

3.1 Frequent Item-Set Mining Problem

Let T be a list of m transactions $\{t_1, t_2, \dots, t_m\}$. In our case, transaction t_i represents metadata description of the i^{th} resource. Each transaction t_i contains a list of r items $\{a_1^i, a_2^i, \dots, a_r^i\}$. Each a_j^i represents a ⟨metadata field, value⟩

pair. The frequent item-set P for T contains the item pairs that appear at least δ (minimum support) times together in transactions data. Mining a set of associative item pairs for a given transaction set, by considering a certain threshold (minimum confidence) is the focal point of this problem.

3.2 FP-Growth Approach

FP Growth algorithm [14] used in this work operates in two phases: at first, it calculates item frequencies and prunes rare items according to a minimum support threshold. A highly compact tree structure called FP-Tree is developed to store the selected frequent items. The construction of FP-Tree requires two scans on the transaction database. This is in contrast to traditional Apriori [15] algorithm which is very expensive as it yields a huge number of candidates explicitly. The information stored in the FP-tree is used to mine the set of frequent patterns.

FP-Growth makes use of two scores in deciding the interesting association rules.

Support: Support implies how frequently an item appears in transaction dataset. For example, if an item appears 10 out of 25 transactions, it has a support of $10/25 = 0.4$.

Confidence: Confidence measures the conditional co-occurrence of the items. It is defined as:

$$confidence(x \rightarrow y) = \frac{support(x \rightarrow y)}{support(x)}, \quad \text{range: } [0, 1]$$

In the second phase, the algorithm determines itemsets which co-occur in the transaction matrix above a certain *confidence*. These co-occurring items are presented as association rules in the form of $x \rightarrow y$. x and y are called as *antecedent* and *consequent* respectively.

FP-Growth algorithm generates a set of association rules:

$$\mathcal{A} = \{A_1, A_2, \dots, A_p\}$$

where each A_i is an association rule of the form given below:

$$A_i = \langle X_i \rightarrow Y_i, \alpha_i, \beta_i \rangle$$

Here, X_i is the antecedent, Y_i is the consequent, α_i is the support and β_i is the confidence ($\text{ConfScore}(A_i)$). X_i may involve multiple $\langle \text{metadata field, value} \rangle$ pairs; whereas Y_i involves only single $\langle \text{metadata field, value} \rangle$ pair.

After studying user query logs of NDLI portal, we notice that few metadata like author name, resource type, educational level, subject, language, publisher name of the resource, publishing year, origin of the document, etc. are used extensively in the most query string. Therefore, we prefer to exhibit those metadata values as suggestions in our system. Thus, a dataset is made by selecting those particular metadata details from each document of the NDLI repository.

We consider 35K bibliographic resources from NDLI repository for our prototype QAC system. The metadata description of each resource is used to create a row in the dataset. The resulted dataset, a transaction matrix of 35K rows and eight columns of frequently used metadata fields, is then fed to FP-Growth algorithm.

The dataset is very sparse in nature with respect to the metadata values. As an example, only five percent of resources are there which are in language “Bengali”, whereas most of the resources are in language “English”. Eventually, FP growth algorithm removes the outlier associations while generating the association rules using *minimum support* and *minimum confidence* thresholds. The algorithm yielded around 95 millions of associations out of $35K \times 8$ transaction matrix and those are then indexed in Apache SOLR¹ to be used in the online process.

4 Online Query Suggestion

In this section, we present our hierarchical QAC system, which primarily shows suggestions by fetching association rules from the repository generated in the previous offline phase. The query suggestion module generates suggestion list at each n^{th} level based on the selections up to $(n - 1)^{th}$ level. Whereas, the suggestions at level 0 depends entirely on user provided prefix, selections at the preceding levels are used for suggestions at other levels (i.e., levels greater than 0). For example, Fig. 3(a) shows the suggestions presented at 2nd level, which is entirely based on the selected sequence “subject::Mathematics, education level::class IX and X”.

Let us consider that $\mathcal{S}^{(n-1)}$ be the sequence of $\langle \text{metadata field, value} \rangle$ pairs selected by the user upto $(n - 1)^{th}$ level. Top- k suggestions at n^{th} level are picked up from association rules ($\mathcal{A}[\mathcal{S}^{(n)}]$) based on their corresponding confidence values.

$$\mathcal{A}[\mathcal{S}^{(n)}] = \{A_1, A_2, \dots, A_k\}$$

with constraints on each A_i : $X_i = \mathcal{S}^{(n-1)}$ and $\text{ConfScore}(A_i) \leq \text{ConfScore}(A_{i-1})$. The candidates for suggestion list at n^{th} level ($\mathcal{G}^{(n)}$) is formed with consequent parts of each A_i , $i = 1, 2, \dots k$.

$$\mathcal{G}^{(n)} = \{Y_1, Y_2, \dots, Y_k\}$$

At any level, when a user places a prefix, the model focuses on re-arranging the ranked list which is initially returned at that level by the suggestion generator. The initial ranked list is then refined by exact matching of the given prefix, ranked on confidence score. Figure 3(b) depicts the refinement in suggestions at 2nd level, on providing prefix “bo” by a user.

In some situation, when a user enters a space for more suggestion at n^{th} level, the system may fail to fetch further suggestion due to the absence of association rules in the repository for the particular selected sequence up to $(n - 1)^{th}$ level.

¹ <http://lucene.apache.org/solr/>.

subject :: Mathematics education level :: class IX and X
subject :: Mathematics education level :: class IX and X in language :: english
subject :: Mathematics education level :: class IX and X resource type :: book
subject :: Mathematics education level :: class IX and X resource type :: solution
subject :: Mathematics education level :: class IX and X resource type :: question set
subject :: Mathematics education level :: class IX and X resource type :: question paper
subject :: Mathematics education level :: class IX and X source :: StemEZ
subject :: Mathematics education level :: class IX and X published by :: StemEZ.com
subject :: Mathematics education level :: class IX and X education level :: class XI and XII
subject :: Mathematics education level :: class IX and X published by :: Government of Karnataka
subject :: Mathematics education level :: class IX and X source :: Karnataka Secondary Education Examination Board

(a)

subject :: Mathematics education level :: class IX and X bo
subject :: Mathematics education level :: class IX and X resource type :: book
subject :: Mathematics education level :: class IX and X source :: Karnataka Secondary Education Examination Board
subject :: Mathematics education level :: class IX and X published by :: Gujarat State Board of School Textbooks
subject :: Mathematics education level :: class IX and X source :: Gujarat Secondary and Higher Secondary Education Board
subject :: Mathematics education level :: class IX and X published by :: Central Board of Secondary Education
subject :: Mathematics education level :: class IX and X source :: Board of Secondary Education, Madhya Pradesh

(b)

Fig. 3. Online hierarchical QAC system

For example, a user forms a query “subject :: life science resource type :: book in Language :: urdu” using QAC, and then enters a space for more hints, but fails to get any suggestion. Entering a space after “in Language :: urdu” implies receiving suggestion which can retrieve a subset of resources that can be retrieved with “life science book in urdu language”. As there are no association rules matching the present antecedent, the model backs off to use aggregated facets of the resources retrieved by posing a query to the repository index considering the selections at previous levels as filters. These facets are then presented as suggestions. Figure 4 narrates workflow of the online suggestion system concisely.

5 Quality Assessment

5.1 Experimental Setting

In our interactive QAC system, metadata suggestions are presented hierarchically. As suggestions in each level claim equal importance to form the whole query string, we evaluate up to 3rd level auto-completion including level 0 for all test data-set. A subset of the resources (35K) from NDLI repository is selected for the present study. These resources are annotated and indexed with the earlier mentioned metadata fields like author name, resource type, educational level, subject, language, publisher name of the resource, publishing year, origin of the document, etc.

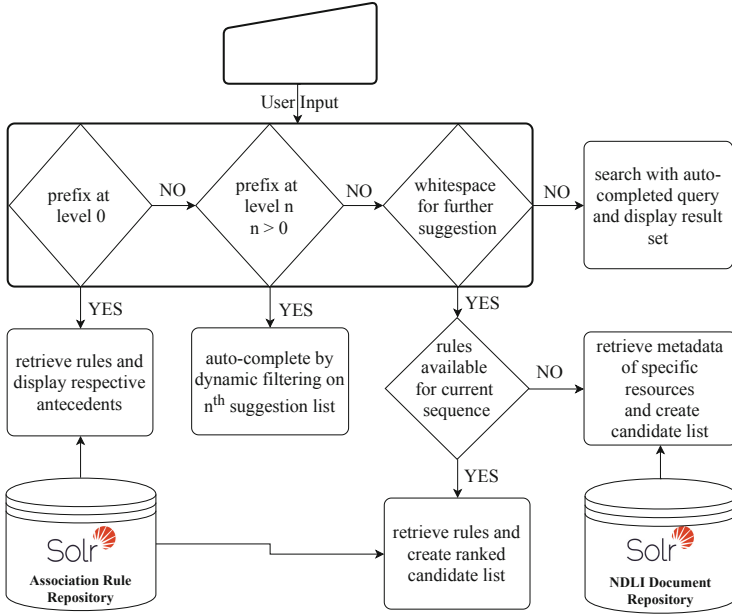


Fig. 4. Workflow of online query suggestion

We design ten search tasks (see Appendix Table 1 for example) to evaluate the proposed system. The tasks are assigned to twenty-five candidates who are both graduates as well as post-graduates in different subjects (e.g. computer science, mathematics, chemistry, biology, history etc.). They have moderate experience (at least 5 years) in querying search engines. The candidates have used both (i) QAC system and (ii) Keyword-Based search in NDLI to specify search intents as described in search tasks assigned to them. The keyword-based system does not provide any suggestion during search and retrieves search results entirely based on standard keyword-search algorithm (e.g., BM25) implemented in SOLR indexing engine. This keyword-based search system is used as a baseline system.

5.2 Evaluation Metrics

We make use of three different metrics to evaluate three aspects of our proposed QAC system.

(1) Assessing query suggestion: To assess the effectiveness of the QAC ranking **Mean Reciprocal Rank (MRR)** has become a standard measure [16–19]. When a user types in a prefix to express query q , a list of matching candidates (C) is shown. In response, the user may choose c_i as the final selection. Then, the Reciprocal Rank (RR) for the prefix is computed as follows:

$$RR = \begin{cases} \frac{1}{rank\ of\ c_i\ in\ C}, & \text{if } c_i \in C \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

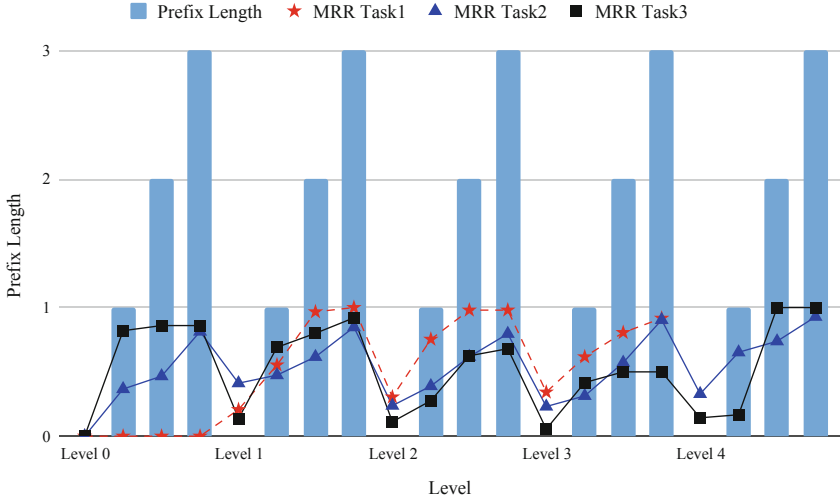


Fig. 5. Performance study by MRR (based on Top-10 suggestions)

In our experiment, a single task is executed by N number of users by producing N different queries. Thus, we calculate MRR by averaging RR values on the number of users (N).

(2) Effort in Specifying Query: We evaluate our system by estimating user effort in specifying query using **Minimal Keystrokes (MKS)** metric [20]. MKS indicates the minimal number of key presses that a user needs to perform to specify a query. For example, if a user types ‘ques’ and ‘question paper’ appears in auto-completion list at 4th position, the user may select the suggestion by pressing Down Arrow key 3 times (by default first option is selected) and one hit on ‘Enter’ key to submit the option. In total, $(4 + 3 + 1) = 8$ keystrokes are required to perform the task.

(3) Quality of Search Results: We report the effectiveness of our model based on the comparison of the result set retrieved by two different search systems mentioned above. Average **Precision@K (P@K)** metric is used in this estimation.

5.3 Performance Analysis

(1) Quality of Query Suggestions: We measure MRR for each prefix length (up to 3), on each level of suggestion for each task. For the sake of readability, we have plotted three out of ten tasks (selected at random) in Fig. 5. We observe the followings from the Fig. 5:

- At any level, MRR score is increasing with prefix length, as lengthy prefix makes search intention more specific.

- MRR score is comparatively higher in higher level (e.g., level 4) as the intent of the whole query becomes more evident after writing a few words.
- For some tasks (e.g., Task 1), the users have not selected any suggestions up to the third prefix of Level 0. In those cases, users decide to put only keywords in Level 0 without selecting any candidate suggestion.
- Some specific MRR plot like Task 1 is discontinued after the third level. It implies that to get expected results for the specific task user does not use any suggestions after the third level.

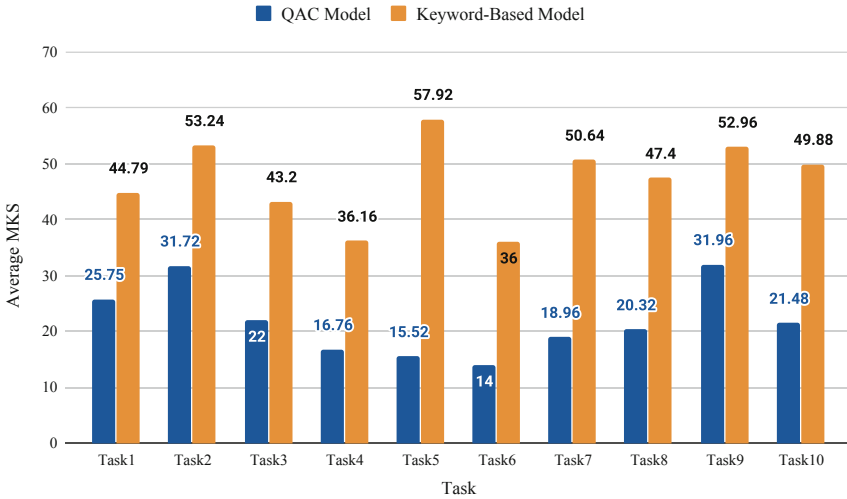


Fig. 6. Performance comparison with baseline system based on MKS

(2) Effort in Specifying Query: Computed MKS values for both the models (proposed QAC and baseline) are presented in Fig. 6. Since the baseline model (keyword-based search) does not provide any suggestion during the search, users have to provide the entire query string by using keystrokes. Therefore, the length of the user-given query string is computed as minimum keystrokes in the keyword-based search. On the other hand, to form a query using our QAC system, user has to do some specific types of activities at each level. For a given level, a user has to (i) enter a prefix, (ii) scroll through the suggestions using arrow-key, (iii) select a suggestion by pressing enter-key and (iv) space-key to proceed in next level. The minimal keystrokes for a single task in QAC model is estimated by counting the keystrokes of the mentioned four activities at all levels. We measure MKS for each task by averaging over the number of users (N), separately for two search systems.

Figure 6 depicts the average keystrokes for two individual search systems, and it may be concluded that our QAC system is more convenient than the baseline system. The QAC system saved 53.73% keystrokes across the tasks.

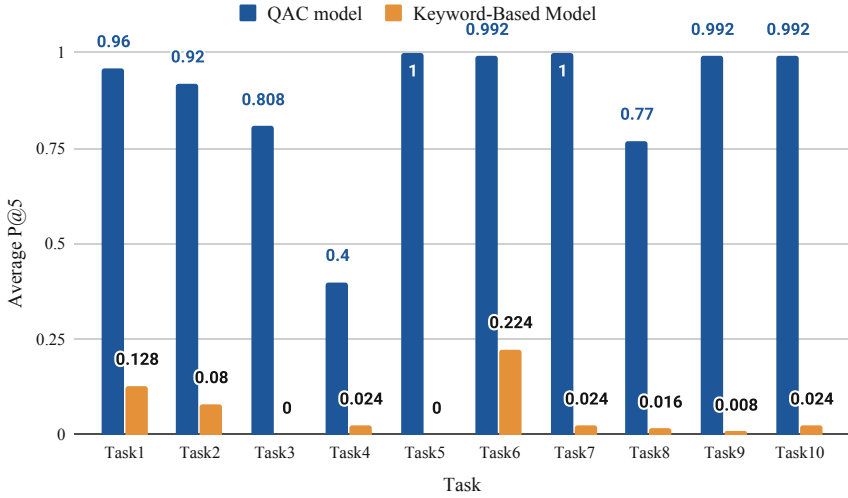


Fig. 7. Comparison of retrieval performance of the proposed system against baseline

(3) Quality of Search Results: Relevance feedback from the same set of users on the top 5 results is obtained from both models (QAC and Normal Keyword Search). Average precision values on the top 5 results (average P@5) for each task, in both the systems, are plotted in Fig. 7. It asserts that our system outperforms the baseline system in retrieving relevant results, and it is estimated that for the assigned ten tasks, the QAC system is 83% more effective than the keyword-based model on average.

6 Conclusion

In this work, we presented a two-step approach for automatic query suggestion based on metadata for searching bibliographic resources in a digital library. The first stage of this pipeline aims to gather associated metadata pairs based on their occurrence in bibliographic resources. It is modeled as a frequent itemset mining problem, and FP-growth algorithm is used in this implementation. The online QAC model suggests metadata values based on the stored association in the previous stage. The model yields a consistent performance while tested with state-of-art QAC evaluation metrics. However, we notice that the proposed system occasionally fails to fetch relevant documents in the top list. As per our observation, improper annotation of metadata values in resources induced this situation. The authors intend to perform a more comprehensive evaluation by considering other mechanisms of query suggestions, e.g., simple keyword suggestions generated through information retrieval measures.

Acknowledgement. This work is supported by IBM Research through Shared University Research grant and Ministry of Human Resource Development, Government of India the National Digital Library of India project.

Appendix: List of Tasks for Assessment

Table 1. List of ten search tasks used in evaluation

Task ID	Description	Sample query in baseline model	Sample query in QAC model
1	Imagine you are a student of class 9, your class test is knocking the door, and still, you have some doubts in a topic of biology called “algae and fungi”. You have read about that but not sure about the question pattern. So, you need some sample question on the topic	Questions on Algae and fungi from Biology for class 9 student	keyword :: Algae and fungi education level :: class IX and X resource type :: question set subject :: Life sciences; biology
2	Imagine you are a student of Urdu medium school of class 6. Today your history teacher advised a reference book. You forgot the author name but remembered only the publishing year of the book i.e. 2006. Now you have to search the book	History book published in 2006 for class 6 urdu medium	subject :: History & geography resource type :: book published at :: 2006 in language :: urdu
3	Imagine you are a student of class 10 and preparing lessons on geometry. Besides, you need to check what kind of question may appear in the examination	Geometry practice question paper for class 10	subject :: Geometry resource type :: question set education level :: class IX and X
4	Imagine you are a class 6 student, studying in a Telugu medium government school in Andhra Pradesh. You are searching for life science books which are published by SCERT Andhra Pradesh	SCERT life science book for class 6 in telegu	source :: SCERT Andhra Pradesh education level :: class V to VIII subject :: Life sciences; biology resource type :: book in language :: telugu
5	A student of class 10, studying in an English medium school, have completed reading about “Pythagorean Theorem” and need to check the question papers on it	Questions on Pythagoras Theorem for class 10 student in English	keyword :: pythagorean theorem education level :: class IX and X resource type :: question set in language :: english
6	Imagine you are a student of class 8, and you need to search some book of life science published by NCERT for getting further knowledge	NCERT books on life science for class 8 student	source :: NCERT resource type :: book subject :: Life sciences; biology education level :: class V to VIII

(continued)

Table 1. (*continued*)

Task ID	Description	Sample query in baseline model	Sample query in QAC model
7	Imagine you are a student of class 11 and reading some topics on computer science. You heard about the excellent teaching pattern of a professor named Ashwini Patil. You want to search video lectures taught by him	Video lecture on computer science by ashwini patil for class XI	resource type::video lecture subject::Computer science authored by::Patil Ashwini education level::class XI and XII
8	Imagine you have to teach some Bengali medium students of class 5 to 8. You are searching of some books on history and geography to check thoroughly	History geography text book for class 5 to 8 bengali medium	subject::History & geography resource type::book education level::class V to VIII in language::bengali
9	Imagine you are studying at class 9 in an English medium school under CBSE board. You have taken computer science as an elective subject. You need to search some book on it in English only	CBSE computer science book for class 9 in English	source::Central Board of Secondary Education subject::Computer science resource type::book education level::class IX and X in language::english
10	Imagine you are studying in class 11 of an English medium school and preparing the topic “Maxwell’s equations”. You want to search what kind of questions may appear in the examination	Questions on maxwell’s equations for class 11 in English	keyword::maxwell’s equation resource type::question set education level::class XI and XII in language::english

References

1. Niu, X., Hemminger, B.: Analyzing the interaction patterns in a faceted search interface. *J. Assoc. Inf. Sci. Technol.* **66**(5), 1030–1047 (2015)
2. Huston, S., Croft, W.B.: Evaluating verbose query processing techniques. In: *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010*, pp. 291–298. ACM, New York (2010)
3. Sadhu, S., Bhowmick, P.K.: Automatic segmentation and semantic annotation of verbose queries in digital library. In: Méndez, E., Crestani, F., Ribeiro, C., David, G., Lopes, J.C. (eds.) *TPDL 2018. LNCS*, vol. 11057, pp. 270–276. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00066-0_23
4. Ma, H., Yang, H., King, I., Lyu, M.R.: Learning latent semantic relations from clickthrough data for query suggestion. In: *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, pp. 709–718. ACM (2008)
5. Beeferman, D., Berger, A.: Agglomerative clustering of a search engine query log. In: *KDD*, vol. 2000, pp. 407–416 (2000)

6. Wen, J.-R., Nie, J.-Y., Zhang, H.-J.: Clustering user queries of a search engine. In: Proceedings of the 10th International Conference on World Wide Web, pp. 162–168. Citeseer (2001)
7. Baeza-Yates, R., Hurtado, C., Mendoza, M.: Query recommendation using query logs in search engines. In: Lindner, W., Mesiti, M., Türker, C., Tzitzikas, Y., Vakali, A.I. (eds.) EDBT 2004. LNCS, vol. 3268, pp. 588–596. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-30192-9_58
8. Cao, H., et al.: Context-aware query suggestion by mining click-through and session data. In: Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 875–883. ACM (2008)
9. Shokouhi, M., Radinsky, K.: Time-sensitive query auto-completion. In: Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 601–610. ACM (2012)
10. Shokouhi, M.: Learning to personalize query auto-completion. In: Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 103–112. ACM (2013)
11. Bar-Yossef, Z., Kraus, N.: Context-sensitive query auto-completion. In: Proceedings of the 20th International Conference on World Wide Web, pp. 107–116. ACM (2011)
12. Cai, F., Liang, S., De Rijke, M.: Time-sensitive personalized query auto-completion. In: Proceedings of the 23rd ACM International Conference on Information and Knowledge Management, pp. 1599–1608. ACM (2014)
13. Zhang, A., et al.: adaQAC: adaptive query auto-completion via implicit negative feedback. In: Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 143–152. ACM (2015)
14. Han, J., Pei, J., Yin, Y.: Mining frequent patterns without candidate generation. *ACM SIGMOD Rec.* **29**, 1–12 (2000)
15. Agrawal, R., Srikant, R., et al.: Fast algorithms for mining association rules. In: Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, vol. 1215, pp. 487–499 (1994)
16. Jiang, J.-Y., Ke, Y.-Y., Chien, P.-Y., Cheng, P.-J.: Learning user reformulation behavior for query auto-completion. In: Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval, pp. 445–454. ACM (2014)
17. Whiting, S., Jose, J.M.: Recent and robust query auto-completion. In: Proceedings of the 23rd International Conference on World Wide Web, pp. 971–982. ACM (2014)
18. Cai, F., Liang, S., de Rijke, M.: Prefix-adaptive and time-sensitive personalized query auto completion. *IEEE Trans. Knowl. Data Eng.* **28**(9), 2452–2466 (2016)
19. Cai, F., de Rijke, M.: Selectively personalizing query auto-completion. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 993–996. ACM (2016)
20. Duan, H., Hsu, B.-J.P.: Online spelling correction for query completion. In: Proceedings of the 20th International Conference on World Wide Web, pp. 117–126. ACM (2011)



Investigating the Effectiveness of Client-Side Search/Browse Without a Network Connection

Hussein Suleman^(✉)

University of Cape Town, Cape Town, South Africa
hussein@cs.uct.ac.za
<http://dl.cs.uct.ac.za/>

Abstract. Search and browse, incorporating elements of information retrieval and database operations, are core services in most digital repository toolkits. These are often implemented using a server-side index, such as that produced by Apache SOLR. However, sometimes a small collection needs to be static and portable, or stored client-side. It is proposed that, in these instances, browser-based search and browse is possible, using standard facilities within the browser. This was implemented and evaluated for varying behaviours and collection sizes. The results show that it was possible to achieve fast performance for typical queries on small- to medium-sized collections.

Keywords: Search · Browse · Client-side · Offline · Browser-based

1 Introduction

Digital library systems are often created by using a toolkit for this purpose, with many popular toolkits currently available as open source software [7]. DSpace [8] is a popular example of a toolkit that is often used for institutional repositories while AtoM [12] is a popular example of a toolkit that is often used for heritage collection management. Many users are satisfied with the functionality provided by such toolkits and need look no further for their repository systems.

However, there are particular circumstances where these systems do not work well. Firstly, in some parts of the world, network access is not as reliable as it is elsewhere, so purely online systems will be inaccessible to some potential users (e.g., judges in remote parts of African countries, trying to access case law on tablets). Secondly, most digital repository tools require technical expertise for installation and configuration and some users do not have this expertise or access to experts or the resources to hire expertise (e.g., historians in most South African universities). Thirdly, in low-resource environments (such as most poor countries), there is a higher risk with more complex software systems as long-term maintenance of systems and data collections is more difficult with less funding and few experts. Fourthly, not everyone needs DSpace - it is clearly overkill to install DSpace to manage your personal photo album.

The Five Hundred Year Archive (FHYA) [1] is a project attempting to create an exemplar of material that relates to pre-colonial history in Southern Africa. The number of resources contemplated is of the order of 10000, and a prototype AtoM system was configured for this. The project had some difficulty in finding ongoing developer support as its focus was the data and not the continuing maintenance of software systems. Arguably, many owners of small archives face similar problems, as their collections and resources are not large enough to justify investment in typical repository systems.

This raised the question of whether or not a simpler archiving tool would suffice for such projects. The development of such a prototype simple archiving tool began with the central search and browse interface as an experiment to determine if this could be done without the need for a server at all, using all resources hosted locally and using only the browser for all computation. An earlier project [9] included a simple HTML search using Javascript, but performance was not evaluated and the functionality was restricted to Web pages.

This paper reports on the development and evaluation of a general-purpose offline client-side metadata search and browse system, also referred to as a faceted search system. The research focus was on feasibility and efficiency. From a feasibility perspective, the offline tool worked correctly for a small amount of FHYA test data (approximately 100 items), but it was not known how well this would work for larger realistic-sized collections. The key research question therefore addressed in this paper is: how scalable is this alternative off-line client-side architecture for search and browse systems?

2 System Design

The search implementation, using an extended Boolean model, was designed in accordance with typical information retrieval principles, as described in Managing Gigabytes [14], with browse operations supported through the addition of database-style indices of all items matching a given term.

Thus, there are 2 sets of indices, both stored as XML documents so they can be processed using built-in browser facilities. The search index for the term “study” will contain a listing of identifiers of all items that include the term. The browse index for the field “date” with a value of “1977” will contain a listing of identifiers of all items where the date field has the value “1977”. In both cases, titles for documents are included to support fast single queries without a separate title listing. Table 1 shows a snippet of a typical browse index. Search indices are similar, with the inclusion of a weight for each document.

The search system was comprised of 2 applications:

- **create_index.pl** is a Perl script to build the indices required for search/browse operations. All metadata is read into memory and processed using various data structures to create inverted indices as well as lists of items for each browsable field.
- **search.js** is a Javascript script to execute a search/browse operation and display results within the browser window. The query is extracted from the

Table 1. Snippet of typical browse index

```

<index>
<bif id="571" file="32000/record303578.xml" title="The wages-prices spiral
: a study in distributive shares"/>
<bif id="26332" file="32000/record215502.xml" title="Cerebral palsy"/>
<bif id="1423" file="32000/record212306.xml" title="The waveforms of
atmospherics and the propagation of very low frequency radio waves"/>
<bif id="21682" file="32000/record480241.xml" title="The attitude of the
Tractarians to the Roman Catholic Church, 1833-1850"/>
...

```

browser's search form. The necessary search and browse indices are then loaded and processed to generate a list of results in order of probable relevance and filtered by the specified browse fields.

The system is configurable in that fields for searching and browsing can be specified in a configuration file. Nominally, search operations collapse all fields into a single unstructured piece of text during indexing. The system also includes the ability to specify/index individual fields and subsets of individual fields, allowing queries such as "name:Jak" where the source metadata might have fields named "creator" and "contributor". This field-based search was not evaluated in this paper as the specification of a field simply creates a separate index for that field and this will not affect performance differently from non-fielded queries.

Figure 1 shows a screen snapshot of the search/browse interface used for these experiments.

3 Evaluation

A multi-factor design was adopted to evaluate the performance of the system, considering: size of collection; complexity of search query; complexity of browse query; and variability in queries.

3.1 Dataset

A dataset with human-assigned metadata fields covering a wide range of topics, and with some fields having controlled vocabularies, was deemed necessary.

Many heritage collections have somewhat skewed datasets with more items on some topics than on others.

Electronic thesis metadata, however, has a large amount of variety, as every record is, to some degree, created by a different individual and the records span many disciplines. ETD metadata was harvested using the Open Archive Initiative Protocol for Metadata Harvesting (OAI-PMH) [4] from the NDLTD Union Archive¹.

¹ NDLTD Union Archive, <http://union.ndltd.org/OAI-PMH/>.

Search Results

The screenshot displays a search interface with the following components:

- Query:** A text input field containing "clinical education".
- publisher:** A dropdown menu with "University of Oxford" selected.
- subject:** A dropdown menu with "All" selected.
- date:** A list of years from 2000 to 2016, with 2009 highlighted.
- Results:** A list of 11 search results, each consisting of a numbered title, a URL, and a file path. The results are:
 - [Schools and education in Gloucestershire and the neighbouring count Reformation](#)
32000/record272406.xml
 - [Changing values in the education of an elite : the crisis of the Wilhelm](#)
32000/record153189.xml
 - [The limits of change in Japanese policymaking : the case of educator](#)
32000/record153399.xml
 - [Further education as a business?](#)
32000/record219853.xml
 - [A history of the Central Council for Health Education, 1927-1968](#)
32000/record28884.xml
 - [Autonomy and Citizenship : Implications for Citizenship Education](#)
32000/record219500.xml
 - [Clinical and experimental aspects of denervation and reinnervation](#)
32000/record149729.xml
 - [Health education practice in primary classrooms in Pakistan](#)
32000/record147618.xml
 - [Structural, pharmacological and clinical effects of spinal cord injury o](#)
32000/record146495.xml
 - [Developing a systematic framework for the integration of health ecor practice guidelines](#)
32000/record221141.xml
 - [Infrastructure, education and productivity : a multi-country study](#)
32000/record146881.xml

Fig. 1. Screen snapshot of search/browse interface.

517854 records from UK universities were collected – the UK collection was used because it is a complete record due to a national mandate – in the Dublin Core format. Records were then randomly extracted to create experimental collection with sizes of 32000, 16000, 8000, 4000 and 2000. For comparability, each subset was a proper superset of the next smaller subset - so all records in the 2000 item set were contained in the 4000 item set.

Data was stored in individual XML files and each subset was separately indexed.

3.2 Experimental Design

Table 2 lists the queries used for the search and browse aspects of the experiments. These queries were manually selected such that, in each case, the first 2 returned were moderately popular (at about the 30th percentile), the second 2 were highly popular (at about the 80th percentile) and the final query was among the most popular terms within the index. These were all chosen by inspecting the index for the 2000 item collection, thus ensuring that the results would also appear in every other collection.

For each collection size, 8 experiments were conducted. For each of the experiments, 5 queries appropriate to the experiment were used. Every query was submitted 5 times, and an average measurement was recorded.

The 8 experiments resulted from varying the complexity of search – either no term, one term or two terms – and the complexity of the browse – either no fields

Table 2. Queries used in experiments. S1–S5 are search terms used. B1–B5 are browse fields used.

Search/Browse	Query terms
Search (single term)	S1: comparative S2: simple S3: study S4: london S5: university
Search (multiple term)	S1: comparative study S2: simple relationship S3: clinical education S4: disease multiple S5: london university
Browse (single field)	B1: date=1954 B2: date=1959 B3: date=1977 B4: date=1986 B5: date=2011
Browse (multiple field)	B1: date=1954 and univ=University of Wolverhampton B2: date=1959 and univ=University of the West of Scotland B3: date=1977 and univ=University of Southampton B4: date=1986 and univ=University College London B5: date=2011 and univ=University of Oxford

constrained, one field constrained or 2 fields constrained. As no experiment was conducted with neither search nor browse terms, this yielded 8 combinations.

The search system code was instrumented to execute experiments without user involvement but with a simulation of user selection (by manipulating the HTML DOM to set form values) and unchanged result display within the browser. Measurement error was minimised by recording times automatically (using the Javascript Date object) and determining the complete time for all repetitions, followed by finding the average. Results were reported in a browser window at the end of the process and recorded.

3.3 Results and Discussion

All experiments were run on a Macbook with a 1,3 GHz Intel Core i5 processor, 8 GB of 1867 MHz LPDDR3 RAM, and a solid-state hard drive. Waterfox v56.2.10 was used as a browser - this is a derivative of Mozilla's Firefox, which allows for greater developer control. All background applications were kept constant during the experiments.

After a brief indexing discussion, the detailed results for one collection are presented and discussed, followed by a comparison of performance across collections.

Indexing. Table 3 shows the time taken to index the different collections. The indexing time is roughly proportional to the collection size. This is not a major issue as, in practice, a collection is only indexed after changes are made and many collections, such as historical collections, are not changed often.

Table 3. Index creation time

Collection size	Time (in seconds)
2000	23.57
4000	55.82
8000	68.51
16000	134.99
32000	254.13

Collection Size 16000. Figure 2 illustrates the performance results for the single/multiple term search and single/multiple field browse for the 16000 item collection.

For the search queries, the highly popular term “university” appears in almost every result, hence takes a long time to process, given the complexity of the search algorithm. The moderately-popular search terms return results much faster. Browse operations all take a similar amount of time because the browsing operation filters the complete list so the work performed is similar in every case. All operations, including those that process most/all the items in the collection, return results in less than 500 ms.

Table 4 displays the full set of results with all measurements corresponding to the 16000 item collection. The first 2 columns indicate the complexity of the search and browse respectively, and the individual entries are for specific query combinations.

The final column in Table 4 indicates average times for operations with different complexities. These range from 114–213 ms, which is typically adequate as a response time for users. The highest values in the table are related to the S5 (“university”) columns, and the first 2 browse operation rows with no preceding search filter; these all process the largest number of items before returning results to the user so the performance is understandable.

All Collections. Figure 3 illustrates the performance results for the single/multiple term search for all collections, for the specific terms “study” and “clinical education”. The “study” broader term appears in most documents in larger collections and therefore there is a clear increase in processing time. The more specific query appears in fewer documents and therefore has a less predictable outcome. Overall, all results are processed within 150 ms.

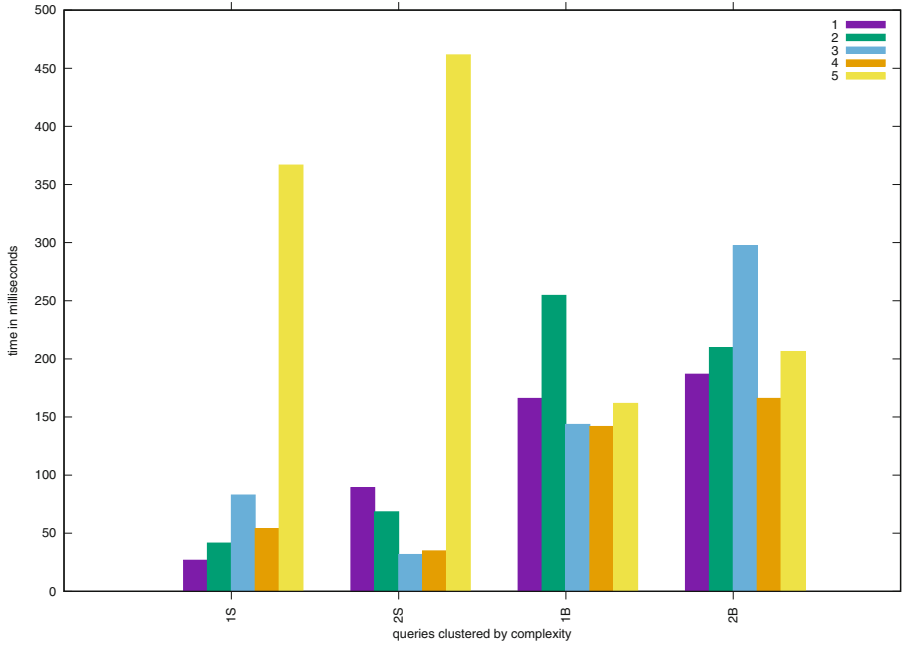


Fig. 2. Average times for queries of difference complexities. 1S/2S/1B/2B are the query complexities while the 5 data points within each cluster are the different queries tested.

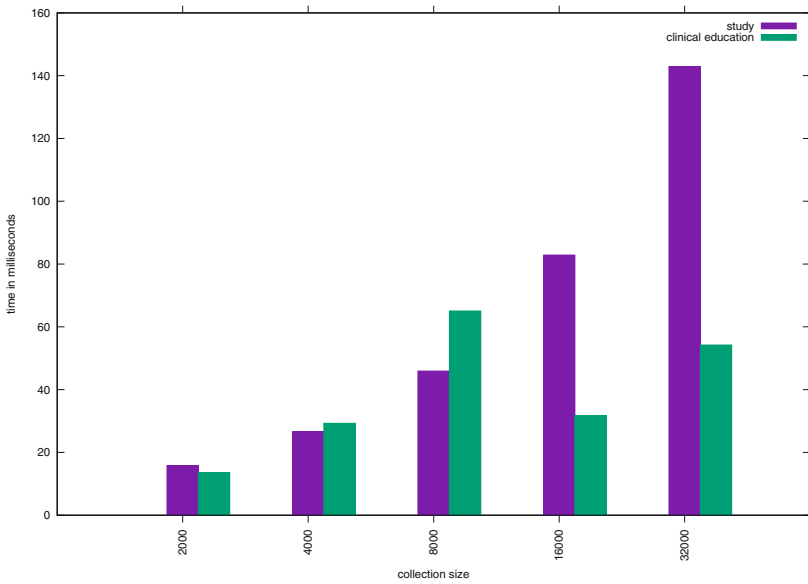


Fig. 3. Average times for search queries of difference complexities across all collection sizes.

Table 4. Detailed performance measurements (in milliseconds) for all query combinations for 16000 record test size. S = Number of search query terms. B = Number of browse fields. S1-S5 are search terms used. B1-B5 are browse fields used. Avg = Average time across all terms/fields for a given complexity.

S\B	S1					S2					S3					S4					S5					Avg						
	B1	B2	B3	B4	B5	B1	B2	B3	B4	B5	B1	B2	B3	B4	B5	B1	B2	B3	B4	B5	B1	B2	B3	B4	B5	B1	B2	B3	B4	B5		
0 1	142	137	133	139	142	155	190	148	143	218	241	285	172	142	160	138	489	137	155	135	154	172	129	130	154	174						
0 2	171	343	545	118	158	155	193	306	175	190	259	169	157	139	314	184	156	179	264	193	166	187	300	133	177	213						
1 0	24	20	25	44	21	29	38	35	30	76	134	67	63	68	83	58	51	52	58	51	442	366	344	332	349	114						
1 1	26	20	20	36	26	21	31	38	24	40	88	81	202	146	223	162	72	56	90	83	410	372	398	394	361	137						
1 2	16	13	18	19	25	14	15	19	20	27	101	58	54	57	68	60	67	87	77	70	772	357	462	403	503	135						
2 0	119	80	89	80	78	67	56	59	124	36	32	31	30	32	34	34	30	45	33	32	392	480	470	513	454	137						
2 1	154	119	85	99	127	53	56	77	63	41	28	22	29	31	40	29	44	34	39	49	480	431	459	507	390	139						
2 2	68	76	72	117	94	95	39	85	150	182	25	23	27	29	36	29	28	32	47	50	498	422	527	410	553	149						

Figure 4 illustrates the performance results for the single/multiple field browse for all collections, for the queries “year = 1977” and “year = 1977 and univ = University of Southampton”. There is a general trend towards increasing times for larger collections, with some specific variation because of the random sample. Overall, all responses are processed within 350 ms. This longer time is due to the entire list of results being filtered, with no pre-filtering from a search query.

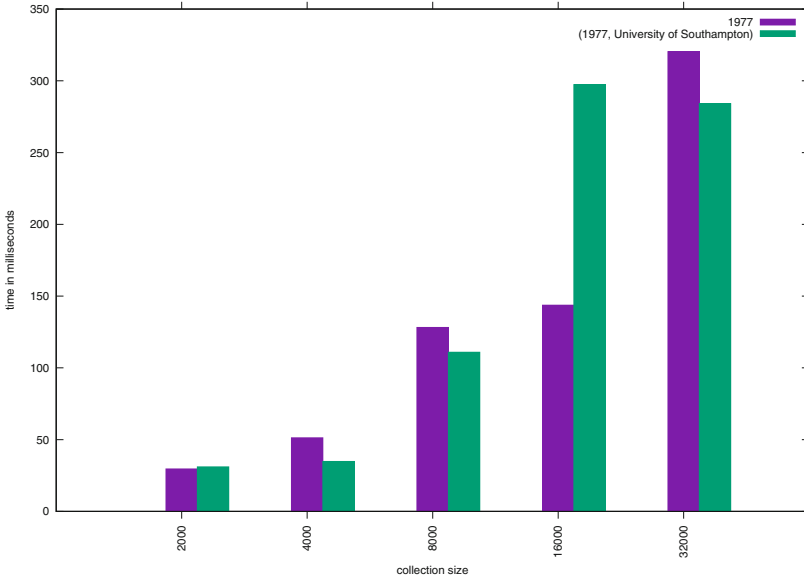


Fig. 4. Average times for browse queries of difference complexities across all collection sizes.

Figure 5 illustrates the performance results for combinations of single/multiple term search and single/multiple field browse for all collections, for the same queries as in the prior figures. The single generic search term “study”, when combined with one or more browse fields, yields longer processing times in many cases. The specific search query “clinical education” is more constrained, with far lower processing times, irrespective of the browse fields specified. All operations are, however, completed within 300 ms.

On average, even for collections with 32000 items, complex search and browse combination queries were successfully processed in less than half a second. Clearly, the query outliers will result in longer times, but this performance is likely to be adequate for most users interacting with a faceted search system.

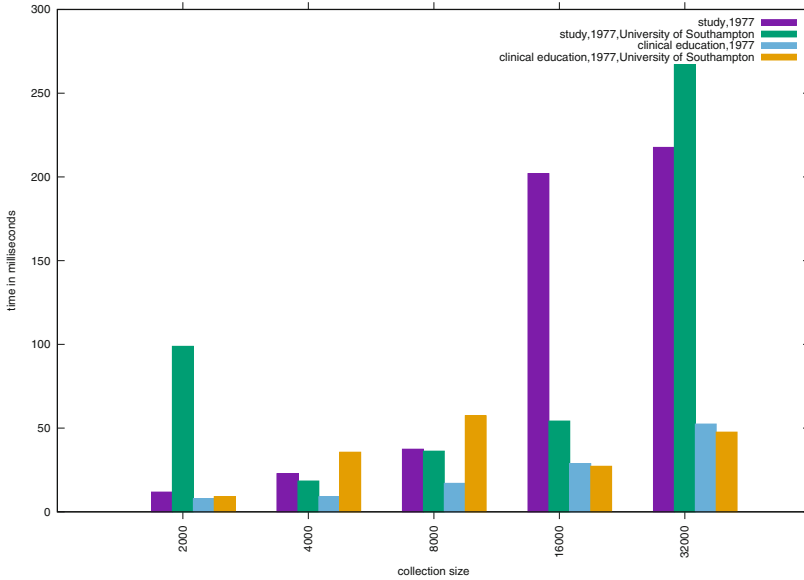


Fig. 5. Average times for faceted search/browse queries of difference complexities across all collection sizes.

4 Related Work

One of the earliest works that demonstrates the notion of simple and static repositories is Project Gutenberg [2], which is an online repository to distribute out-of-print and free ebooks. The core system that manages the data is simple and does not rely on typical archiving tools. All data is required to be in simple formats, even though more advanced formats may be included as well.

The Greenstone project [13] pioneered the idea of a digital library system that could be used offline, where collections were indexed and distributed on CDROM. The technology used to create this experience required the installation of a specific operating system’s Web server on the user’s computer. While the Greenstone approach was innovative and addressed the needs of low-resource environments, it predated the advanced browser technology currently available and exploited in this project.

The SimplyCT project [5,10] tried to define an alternative design approach for digital repositories in low resource environments, such as most African countries. This was exemplified in the Bleek and Lloyd archive, which was completely static and contained a simple HTML-based search service that worked within the browser [9]. Additional experiments with different repositories showed that static file stores could work for a wide range of repository requirements [6,11].

The Open Archives Initiative encouraged the use of simple solutions. While its flagship OAI-PMH standard is considered to embody many of the simplicity principles, the OAI also showed that this could be taken a step further to create

static repository snapshots, enabling interoperability among even participants with little infrastructure [3].

5 Conclusions and Future Work

This experiment sought to investigate the performance of a search and browse system based completely within a browser, with no network connection and no software installation necessary. Using Javascript, such a system is clearly possible with data that is pre-indexed and stored in XML files. The results from extensive experimentation show that such a system has reasonable performance for typical operations on a small- to medium-sized collection.

Most operations completed within half a second, even on a 32000 item collection. This suggests that this offline approach may even work on collections of up to 100000 items in an acceptable amount of time. Given its advantages of portability, simplicity and network-independence, this can provide a useful alternative to indexing systems like Apache SOLR in specific situations.

On reflection, this experiment exploits advances in hardware and browser software in order to overcome other environmental limitations, like expertise, funding and Internet connectivity. It is an attempt to use the latest technological advances to address the limitations of low-resource environments.

Future work may include the use of JSON instead of XML for data storage. This may result in a constant factor improvement, due to small data sizes and faster parsing. Also, partitioning of the index files is being considered to support faster loading of initial results for simpler queries; this may help to address some of the slow browse operations that filter large result sets.

Acknowledgements. This research was partially funded by the National Research Foundation of South Africa (Grant numbers: 85470 and 105862) and University of Cape Town. The authors acknowledge that opinions, findings and conclusions or recommendations expressed in this publication are that of the authors, and that the NRF accepts no liability whatsoever in this regard.

References

1. Archive and Public Culture: The five hundred year archive. www.apc.uct.ac.za/apc/research/projects/five-hundred-year-archive
2. Hart, M.: The history and philosophy of project gutenberg. *Proj. Gutenberg* **3**, 1–11 (1992)
3. Hochstenbach, P., Jerez, H., Van de Sompel, H.: The OAI-PMH static repository and static repository gateway. In: *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 210–217. IEEE Computer Society (2003)
4. Lagoze, C., Van de Sompel, H.: The open archives initiative: building a low-barrier interoperability framework. In: *Proceedings of the 1st ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 54–62. ACM (2001)
5. Phiri, L.: Simple digital libraries. Ph.D. thesis, University of Cape Town (2013)

6. Phiri, L., Williams, K., Robinson, M., Hammar, S., Suleman, H.: Bonolo: a general digital library system for file-based collections. In: Chen, H.-H., Chowdhury, G. (eds.) ICADL 2012. LNCS, vol. 7634, pp. 49–58. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-34752-8_6
7. Pyrounakis, G., Nikolaidou, M., Hatzopoulos, M.: Building digital collections using open source digital repository software: a comparative study. *Int. J. Digit. Libr. Syst. (IJDLS)* 4(1), 10–24 (2014)
8. Smith, M., et al.: DSpace: an open source dynamic digital repository (2003)
9. Suleman, H.: Digital libraries without databases: the bleek and lloyd collection. In: Kovács, L., Fuhr, N., Meghini, C. (eds.) ECDL 2007. LNCS, vol. 4675, pp. 392–403. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74851-9_33
10. Suleman, H.: An African perspective on digital preservation. In: *Multimedia Information Extraction and Digital Heritage Preservation*, pp. 295–306. World Scientific (2011)
11. Suleman, H., Bowes, M., Hirst, M., Subrun, S.: Hybrid online-offline digital collections. In: *Proceedings of the 2010 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists*, pp. 421–425. ACM (2010)
12. Van Garderen, P.: The ICA-AtoM project and technology. In: *Third Meeting on Archival Information Databases. (16–17 julio 2009: Rio de Janeiro)*. Trabajos presentados. Rio de Janeiro: Association of Brazilian Archivists (2009)
13. Witten, I.H., McNab, R.J., Boddie, S.J., Bainbridge, D.: *Greenstone: a comprehensive open-source digital library software system* (2000)
14. Witten, I.H., Moffat, A., Bell, T.C.: *Managing Gigabytes: Compressing and Indexing Documents and Images*. Morgan Kaufmann, Burlington (1999)



SchenQL: A Concept of a Domain-Specific Query Language on Bibliographic Metadata

Christin Katharina Kreutz^(✉), Michael Wolz^{}, and Ralf Schenkel^{}

Trier University, 54286 Trier, Germany
{krechtzch,s4miwolz,schenkel}@uni-trier.de

Abstract. Information access needs to be uncomplicated, users rather use incorrect data which is easily received than correct information which is harder to obtain. Querying bibliographic metadata from digital libraries mainly supports simple textual queries. A user's demand for answering more sophisticated queries could be fulfilled by the usage of SQL. As such means are highly complex and challenging even for trained programmers, a domain-specific query language is needed to provide a straightforward way to access data.

In this paper we present SchenQL, a simple query language focused on bibliographic metadata in the area of computer science while using the vocabulary of domain-experts. By facilitating a plain syntax and fundamental aggregate functions, we propose an easy-to-learn domain-specific query language capable of search and exploration. It is suitable for domain-experts as well as casual users while still providing the possibility to answer complicated queries.

Keywords: Domain-specific query language ·
Bibliographic metadata · Digital libraries

1 Introduction

Scientific writing almost always starts with thorough bibliographic research on relevant publications, authors, conferences, journals and institutions. While web search is excellent for query answering and intuitively performed, not all retrieved information is correct, unbiased and categorized [4]. The arising problem is people's tendency of rather using poor information sources which are easy to query than more reliable sources which might be harder to access [5]. This introduces the need for more formal and structured information sources such as digital libraries specialized on the underlying data which are also easy to query. Often-times, interfaces of digital libraries offer the possibility to execute search on all metadata or query attributes. In many cases, they are not suitable to fulfil users' information needs directly when confronted with advanced query conditions such as "*Which are the five most cited publications written by A?*". Popular examples of full-text search engines with metadata-enhanced search for a subset of

their attributes [6] are dblp [20] or semantic scholar [29]. Complex relations of all bibliographic metadata can be precisely traversed using textual queries with restricted focus and manual apposition of constraints. While SQL is a standard way of querying databases, it is highly difficult to master [34]. Domain experts are familiar with the schema but are not experienced in using all-purpose query languages such as SQL [1,23]. Casual users of digital libraries are not versed in either.

To close this gap, we propose SchenQL, a query language (QL) specified on the domain of bibliographic metadata. It is constructed to be easily operated by domain-experts as well as casual users as it uses the vocabulary of digital libraries in its syntax. While domain-specific query languages (DSLs) provide a multitude of advantages [9], the most important aspect in the conception of SchenQL was that no programming skills or database schema knowledge is required to use it.

The contribution of this paper lies in the presentation of the concept of a domain-specific query language on bibliographic metadata in computer science which is the first to the best of our knowledge. It focuses on retrieval and exploration aspects as well as aggregate functions.

The remainder of this paper is structured as follows: In Sect. 2 related work is discussed before our data model is presented in Sect. 3. Section 4 introduces the structure and syntax of SchenQL. The last Sect. 5 describes future research.

2 Related Work

Areas adjacent to the one we are tackling are *search on digital libraries*, *formalized query languages*, *query languages deriving queries from natural language* and *domain-specific query languages*.

With *search on digital libraries*, several aspects need to be taken into consideration: The MARC format [11] is a standard for information exchange in digital libraries [4]. While it is useful for known-item search, topical search might be problematic as contents of the corresponding fields can only be interpreted by domain-experts [4]. Most interfaces on digital libraries provide field-based boolean search [28] which can lead to difficulties in formulating queries that require the definition and concatenation of multiple attributes. This might cause a substantial cognitive workload on the user [7]. Withholding or restriction of faceted search on these engines fails to answer complex search tasks [6]. We provide a search option on topical information which even casual users can operate while also offering the possibility to clearly define search terms for numerous attributes in one query. Faceted search is possible on almost all attributes.

Numerous works come from the area of *formalized query languages*. Examples for SQL-like domain-unspecific QLs on heterogeneous networks are BiQL [15] which is suitable for network analysis and focuses on create/update functions or SnQL [27], a social network query language specialized on transformation and construction operations. Other SQL-like QLs which are not restricted on a domain, focus on graph traversal while being document-centric [31] or social network analysis while being person-centric [30]. Some SQL-like query languages

operate on RDF [10] or bibliographic databases using the MARC format [22]. While SchenQL is as structured as these formalized query languages, it does not depend on complicated SQL-like syntax. We argue that SQL is too complex to be effectively operated by casual users and hardly even by computer scientists. Our DSL is neither person-centric nor document-centric but both. Its focal point is the retrieval and exploration of data contrasting the creation/transformation focus of several of the presented languages.

Further research applies Qs in a way that users do not interact with it directly in using the system but in their back end. In many cases, graph-like structured data of heterogeneous networks is used to locate information semantically relevant to a unspecific query [2, 14]. Such an indirection could be a future step in the development of SchenQL.

Whether it be translation of natural language for the formulation of XQuery [21], the processing of identified language parts to build parse trees [3], the representation of natural language queries posed to digital libraries in SPARQL [8] or the recent audio to SQL translation available for different databases and domains [34], *analysis of natural language and its conversion to machine processable queries* is an active field of research. Our proposed DSL does not translate natural language to SQL but offers a syntax which is similar to natural language.

Domain-specific query languages come in many shapes. They can be SQL-like [19], visual Qs [1, 12] or use domain-specific vocabulary [32] but are typically specialized on a certain area. They also come in different complexities: For example MathQL [16] is a query language in markup style on RDF repositories but a user needs to be mathematician to be able to operate it. The DSL proposed by [23] stems from the medical domain and is designed to be used by inexperienced patients as well as medical staff. Naturally, there are hybrid forms: Some natural language to machine-readable query options are domain-specific [26] and some DSLs might be transferable to other domains [9]. With SchenQL, we want to provide a DSL which uses vocabulary from the domain of bibliographic metadata while being useful for experts as well as casual users.

3 Data Model

Contrary to attempts of modelling every particularity of bibliographic metadata as seen with MARC format [11] or Dublin Core, we concentrate on a few basic objects in our data model. In our concept, bibliographic metadata consists of persons and publications which they authored or edited. These persons can be affiliated with certain institutions. Manuscripts can be of type book, chapter, article, master or PhD thesis and are possibly published in different venues (conferences or journals). They can reference other papers and oftentimes are cited themselves. Figure 1 describes the difference between citations and references. Figure 2 shows the relations, specializations and attributes of data objects in our data model.

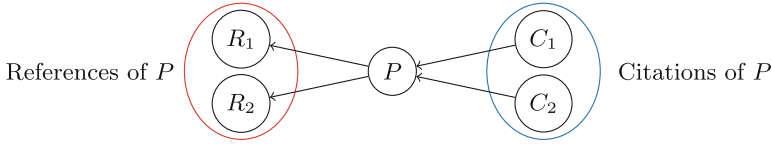


Fig. 1. Nodes symbolize publications, edges between papers symbolize citations. C_1 and C_2 are citations of P , R_1 and R_2 are references of P .

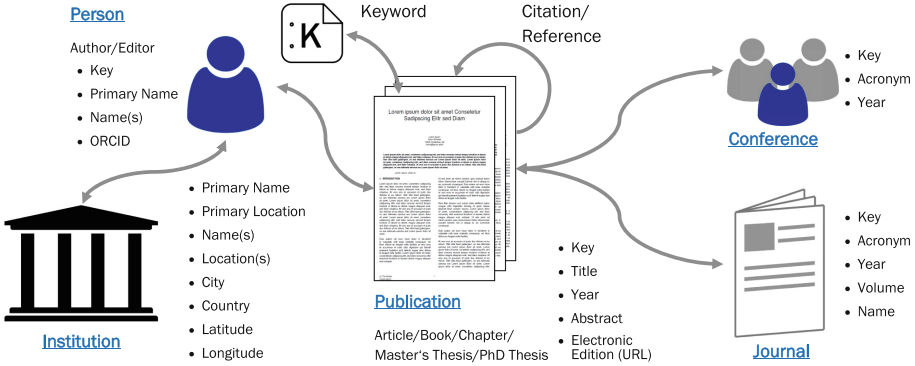


Fig. 2. Relations, specializations and attributes of data objects from our data model. (Color figure online)

4 SchenQL

The SchenQL Query Language was developed to pose the possibility to access bibliographic metadata in a textual manner which resembles natural language for casual as well as expert users of digital libraries in computer science. The fundamental idea in the development of the query language was to hide possibly complex operations behind plain domain-specific vocabulary. Such functionality would enable usage from anyone versed in the vocabulary of the domain without experience in sophisticated query languages such as SQL. SchenQL queries are formulated declarative, not procedural.

4.1 Building Blocks

Base concepts are the basic return objects of the query language. A base concept is connected to an entity of the data model and has multiple attributes. Those base concepts are **publications**, **persons**, **conferences**, **journals** and **institutions**. Upon these concepts, queries can be constructed. Base concepts can be specialized. For example **publications** can be refined by specializations **books**, **chapters**, **articles**, **master** or **PhD theses**. A specialization can be used instead of a base concept in a query.

Restrictions on base concepts are possible by using filters. A filter extracts a subset of the data of a base concept. Literals can be used as identifiers for objects

Table 1. SchenQL base concepts **Publications (PU)**, **persons (PE)**, **conferences (C)**, **journals (J)** and **institutions (I)** with their respective literals (L), specializations (S), filters (F) and standard return values (V).

	PUBLICATION	PERSON	CONFERENCE	JOURNAL	INSTITUTION
L	key, title	key, primary name, orcid	key, acronym	key, acronym	
S	ARTICLE, MASTERTHESIS, CHAPTER, PHDTHESIS, BOOK	AUTHOR, EDITOR			
F	PUBLISHED BY (I), ABOUT (keywords), WRITTEN BY (PE), EDITED BY (PE), APPEARED IN (C J), BEFORE year, IN YEAR year, AFTER year, TITLED title, REFERENCES (PU), CITED BY (PU)	PUBLISHED IN (C J), PUBLISHED WITH (I), WORKS FOR (I), NAMED name, ORCID orcid, AUTHORED (PU), REFERENCES (PU), CITED BY (PU)	ACRONYM acronym, ABOUT (keywords), BEFORE year, IN YEAR year, AFTER year	NAMED name, ACRONYM acronym, ABOUT (keywords), BEFORE year, IN YEAR year, AFTER year, VOLUME volume	NAMED name, CITY city, COUNTRY country, MEMBERS (PE)
V	title	primary name	acronym	acronym	primary name+primary location

from base concepts, they can be used to query for specific data. Attributes of base concepts can be queried. Table 1 gives an overview of literals, specializations, filters and the standard return value for every base concept. In Fig. 2, where base concepts are underlined and written in blue, their attributes are shown.

Functions are used to aggregate data or offer domain-specific operations. Right now, only three functions are implemented in SchenQL: **MOST CITED**, **COUNT** and **KEYWORDS OF**. The function **MOST CITED (PUBLICATION)** can be applied on publications. It counts and orders the number of citations of papers in the following set, and returns their titles as well as their number of citations. By default, the top 5 results are returned. **COUNT** returns the number of objects contained in the following subquery. **KEYWORDS OF (PUBLICATION | CONFERENCE | JOURNAL)** returns the keywords associated with the following base concept. The **LIMIT x** operator with $x \in \mathbb{N}$ can be appended at the end of any query to change the number of displayed results to x .

4.2 Syntax

The syntax of SchenQL follows some simple rules with the goal of being similar to queries formulated in natural language and therefore understandable and easy to construct. Queries are completed with a semicolon, subqueries have to be surrounded by parentheses. It is possible to write singular or plural when

using base concepts or specializations (e.g. CONFERENCE; or CONFERENCES;). Filters follow base concepts or their specializations, can be in arbitrary order and are connected via conjunction. Most filters expect a base concept as parameter (e.g. WRITTEN BY (PERSONS)), several filters expect a string as parameter (e.g. COUNTRY "de"). Specializations can be used in place of base concepts. Instead of a query PERSON NAMED "Ralf Schenkel"; a specialization like AUTHOR NAMED "Ralf Schenkel"; would be possible.

If a filter requires a base concept, parentheses are needed except for the case of using literals for uniquely identifying objects of the base concept. For example PUBLICATIONS WRITTEN BY "Ralf Schenkel"; is semantically equivalent to PUBLICATIONS WRITTEN BY (PERSONS NAMED "Ralf Schenkel");.

COUNT can process any kind of subquery (e.g. COUNT (INSTITUTIONS);). LIMIT x can be appended to any query, MOST CITED requires a subquery which produces objects of base concept PUBLICATION (e.g. MOST CITED (ARTICLES APPEARED IN "icadl") LIMIT 10; returns the ten most cited articles which have appeared in the conference with acronym ICADL). KEYWORDS OF requires a subquery, which returns objects of type PUBLICATION, JOURNAL or CONFERENCE.

Attributes of base concepts can be accessed by putting the queried for attribute in quotation marks in front of a base concept and connecting both parts with an OF (e.g. "key" OF JOURNALS IN YEAR 2010;).

4.3 Implementation and Evaluation

Lexer and parser of the compiler were built using ANTLR [24] with Java as target language. The compiler translates queries from SchenQL to SQL and runs them on a MySQL 8.0.16 database. SchenQL can be used in a terminal client similar to the MySQL shell.

A thorough evaluation of SchenQL using data from a combination of datasets [20, 29, 33] can be found in [18]. It shows the effectiveness and efficiency of the query language as well as user's satisfaction with it when compared with SQL.

5 Conclusion and Future Work

We proposed SchenQL, a domain-specific query language operating on bibliographic metadata from the area of computer science.

Possible improvements of SchenQL include adding logical set operations and incorporating of more data such as CORE rank [13], keywords derived from doc2vec [25] or bibliometric measures such as h-index [17]. More advanced enhancements of functionalities of SchenQL could include the calculation of the influence graph of publications, centrality of authors, the length of a shortest path between two authors and the introduction of aliases for finding co-authors or co-citations as proposed in [15]. Algorithms for social network analysis as PageRank, computation of mutual neighbours, hubs and authorities or connected components could be worthwhile [30]. As user-defined functions [1, 30] and support in query formulation were well-received in other works [28], they

are a further prospect. The incorporation of social relevance in the search and result representation process as shown in [2, 14] would also be a possibility for extension. Via user profiles, written papers and interesting keywords could be stored, which in terms influence results of search and exploration.

SchenQL could also be used as an intermediate step between a user interface and the underlying database [2, 14]. Operability for casual users would be preserved by hiding future, more complex queries in visualization options in addition to a SchenQL query field.

References

1. Amaral, V., Helmer, S., Moerkotte, G.: A visual query language for HEP analysis. In: Nuclear Science Symposium Conference Record, vol. 2, pp. 829–833 (2003)
2. Amer-Yahia, S., Lakshmanan, L.V.S., Yu, C.: SocialScope: enabling information discovery on social content sites. In: CIDR 2009 (2009)
3. Ballard, B.W., Stumberger, D.E.: Semantic acquisition in TELI: a transportable, user-customized natural language processor. In: ACL, pp. 20–29 (1986)
4. Baeza-Yates, R.A., Ribeiro-Neto, B.A.: Modern Information Retrieval - The Concepts and Technology Behind Search, Second edn. Pearson Education Ltd., Harlow (2011). ISBN 978-0-321-41691-9
5. Bates, M.J.: Task force recommendation 2.3: research and design review: improving user access to library catalog and portal information: final report (version 3). In: Proceedings of the Bicentennial Conference on Bibliographic Control for the New Millennium (2003)
6. Beall, J.: The weaknesses of full-text searching. *J. Acad. Librarianship* **34**(5), 439–443 (2008)
7. Berget, G., Sandnes, F.R.: Why textual search interfaces fail: a study of cognitive skills needed to construct successful queries. *Inf. Res.* **24**(1), n1 (2019)
8. Bloehdorn, S., et al.: Ontology-based question answering for digital libraries. In: Kovács, L., Fuhr, N., Meghini, C. (eds.) ECDL 2007. LNCS, vol. 4675, pp. 14–25. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74851-9_2
9. Borodin, A., Kiselev, Y., Mirvoda, S., Porshnev, S.: On design of domain-specific query language for the metallurgical industry. In: Kozielski, S., Mrozek, D., Kasprowski, P., Małysiak-Mrozek, B., Kostrzewa, D. (eds.) BDAS 2015. CCIS, vol. 521, pp. 505–515. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-18422-7_45
10. Buffereau, B., Picouet, P.: STIL: an extended resource description framework and an advanced query language for metadatabases. In: Li Lee, M., Tan, K.-L., Wuwongse, V. (eds.) DASFAA 2006. LNCS, vol. 3882, pp. 849–858. Springer, Heidelberg (2006). https://doi.org/10.1007/11733836_62
11. Crawford, W.: MARC for Library Use: Understanding the USMARC Formats. Knowledge Industry Publications, Inc. (1994)
12. Collberg, C.S.: A fuzzy visual query language for a domain-specific web search engine. In: Hegarty, M., Meyer, B., Narayanan, N.H. (eds.) Diagrams 2002. LNCS (LNAI), vol. 2317, pp. 176–190. Springer, Heidelberg (2002). https://doi.org/10.1007/3-540-46037-3_20
13. <http://www.core.edu.au/conference-portal>
14. Curtiss, M., et al.: Unicorn: a system for searching the social graph. *PVLDB* **6**(11), 1150–1161 (2013)

15. Dries, A., Nijssen, S., De Raedt, L.: BiQL: a query language for analyzing information networks. In: Berthold, M.R. (ed.) *Bisociative Knowledge Discovery*. LNCS (LNAI), vol. 7250, pp. 147–165. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31830-6_11
16. Guidi, F., Schena, I.: A query language for a metadata framework about mathematical resources. In: Asperti, A., Buchberger, B., Davenport, J.H. (eds.) *MKM 2003*. LNCS, vol. 2594, pp. 105–118. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-36469-2_9
17. Hirsch, J.E.: An index to quantify an individual’s scientific research output. *Proc. Natl. Acad. Sci.* **102**(46), 16569–16572 (2005)
18. Kreutz, C.K., Wolz, M., Schenkel, R.: SchenQL - A Domain-Specific Query Language on Bibliographic Metadata. In: *CoRR abs/1906.06132* (2019)
19. Leser, U.: A query language for biological networks. In: *ECCB/JBI 2005*, p. 39 (2005)
20. Ley, M.: DBLP - some lessons learned. *PVLDB* **2**(2), 1493–1500 (2009)
21. Li, Y., Yang, H., Jagadish, H.V.: Constructing a generic natural language interface for an XML database. In: Ioannidis, Y., et al. (eds.) *EDBT 2006*. LNCS, vol. 3896, pp. 737–754. Springer, Heidelberg (2006). https://doi.org/10.1007/11687238_44
22. Lim, E.-P., Lu, Y.: Distributed query processing for clustered and bibliographic databases. In: *DASFAA*, pp. 441–450 (1997)
23. Madaan, A.: Domain specific multi-stage query language for medical document repositories. *PVLDB* **6**(12), 1410–1415 (2013)
24. Parr, T.J., Quong, R.W.: ANTLR: a predicated- LL(k) parser generator. *Softw. Pract. Exper.* **25**(7), 789–810 (1995)
25. Quoc, L., Mikolov, T.: Distributed representations of sentences and documents. In: *ICML*, pp. 1188–1196 (2014)
26. Rohil, M.K., Rohil, R.K., Rohil, D., Runthala, A.: Natural language interfaces to domain specific knowledge bases: an illustration for querying elements of the periodic table. In: *ICCI*CC 2018*, pp. 517–523 (2018)
27. San Martín, M., Gutiérrez, C., Wood, P.T.: SNQL: a social networks query and transformation language. In: *AMW 2011* (2011)
28. Schaefer, A., Jordan, M., Klas, C.-P., Fuhr, N.: Active support for query formulation in virtual digital libraries: a case study with DAFFODIL. In: Rauber, A., Christodoulakis, S., Tjoa, A.M. (eds.) *ECDL 2005*. LNCS, vol. 3652, pp. 414–425. Springer, Heidelberg (2005). https://doi.org/10.1007/11551362_37
29. <https://www.semanticscholar.org/>
30. Seo, J., Guo, S., Lam, M.S.: SociaLite: an efficient graph query language based on datalog. *IEEE Trans. Knowl. Data Eng.* **27**(7), 1824–1837 (2015)
31. Sheng, L., Özsoyoğlu, Z.M., Özsoyoğlu, G.: A graph query language and its query processing. In: *ICDE 1999*, pp. 572–581 (1999)
32. Tian, H., Sunderraman, R., Calin-Jageman, R., Yang, H., Zhu, Y., Katz, P.S.: NeuroQL: a domain-specific query language for neuroscience data. In: Grust, T., et al. (eds.) *EDBT 2006*. LNCS, vol. 4254, pp. 613–624. Springer, Heidelberg (2006). https://doi.org/10.1007/11896548_46
33. https://www.wikidata.org/wiki/Wikidata:Main_Page
34. Xu, B., et al.: NADAQ: natural language database querying based on deep learning. *IEEE Access* **7**, 35012–35017 (2019)

Information Extraction



Formalising Document Structure and Automatically Recognising Document Elements: A Case Study on Automobile Repair Manuals

Hodai Sugino^(✉), Rei Miyata^(✉), and Satoshi Sato

Nagoya University, Nagoya, Japan
{sugino.hodai,miyata.rei}@c.mbox.nagoya-u.ac.jp,
sato.satoshi@d.mbox.nagoya-u.ac.jp

Abstract. Correct understanding of document structures is vital to enhancing various practices and applications, such as authoring and information retrieval. In this paper, with a focus on procedural documents of automobile repair manuals, we report a formalisation of document structure and an experiment to automatically classify sentences according to document structure. To formalise document structure, we first investigated what types of content are included in the target documents. Through manual annotation, we identified 26 indicative content categories (*content elements*). We then employed an existing standard for technical documents—the Darwin Information Typing Architecture—and formalised a detailed document structure for automobile repair tasks that specifies the arrangement of the content elements. We also examined the feasibility of automatic content element recognition in given documents by implementing sentence classifiers using multiple machine learning methods. The evaluation results revealed that support vector machine-based classifiers generally demonstrated high performance for in-domain data, while convolutional neural network-based classifiers are advantageous for out-of-domain data. We also conducted in-depth analyses to identify classification difficulties.

Keywords: Document structure · Genre analysis · Manual annotation · Automatic text classification

1 Introduction

Proper use of document structure is crucial in various practices and applications, such as authoring, summarisation, and information retrieval. For example, in the field of academic writing, certain rhetorical structures—notably IMRaD (Introduction, Method, Result and Discussion)—have been established and widely employed in writing well-organised scientific articles. In the field of technical documentation, although several document structures, such as the Darwin Information Typing Architecture (DITA), have been developed, our understanding of the detailed rhetorical structures has not advanced significantly.

Therefore, with a focus on the procedural documents of automobile technical manuals, we formalise a document structure with the specific intention of enhancing document authoring and information retrieval. The task of formalising document structures can be addressed from two aspects: (1) content (*What content should be included in the document?*) and (2) arrangement (*In what order should the content be arranged in the document?*). In this study, we addressed this task by (1) manually investigating the indicative abstraction of what is written, which we call **content elements**, in the collected documents and (2) mapping identified content elements onto a DITA-based document structure.

With the aim of effective document authoring and information retrieval, we are developing techniques to automatically recognise the structure of given documents. The fundamental problem we face is the scarcity of human-annotated training data. Although recent natural language processing research has greatly advanced through the use of deep neural network methods with large-scale data, in many real-world situations, annotated training data is scarce. Therefore, it is important to examine various methods to explore their feasibilities. In this paper, as a starting point, we present experimental results on the evaluation of automatic annotation of content elements using multiple machine learning methods.

The remainder of this paper is structured as follows. Section 2 describes related work on the investigation and application of document structure. In Sect. 3, we present our data, method, and implementation for document annotation and create a typology of content elements. In Sect. 4, based on the DITA structure, we formalise a document structure and map the content elements onto it. Section 5 presents the experimental results of automatic annotation of content elements with error analysis, and Sect. 6 concludes the paper and suggests future research directions.

2 Related Work

Although each document has a particular communicative role and unique information, document structures can be formalised to a certain degree depending on the genre and domain. Taking Reiter and Dale’s [22] definitions of two types of macro-level planning of natural language generation, namely content determination and document structuring, property of document structures can be divided into **content** (what types of content are included) and **arrangement** (in what order the contents are arranged).

To date, descriptive analyses of document structures and rhetorical structures have been conducted in the field of English for specific purposes (ESP), which is often referred to as ‘genre analysis’ or ‘move analysis’ [2, 3, 26, 27]. For example, Swales [26, p. 141] clarified that introduction sections of research articles include the following rhetorical movements: [Move 1] Establishing territory (claiming centrality → making topic generalisation(s) → reviewing items of previous research); [Move 2] Establishing a niche (counter-claiming/indicating a

gap/question-raising/continuing a tradition); [Move 3] Occupying the niche (outlining purposes/announcing present research → announcing principal findings → indicating research article structure).

A number of genre (move) analyses have been conducted on research articles across different academic fields, such as sociology [5], engineering [20] and empirical law [29], and different sections, such as conclusions [6] and abstracts [10]. Document structures have also been widely utilised in academic writing [28], summarisation [30] and information retrieval [18].

In the field of technical documentation, several guidelines and standards have been developed for authoring well-structured documents [7, 15, 16, 23]. One of the widely acknowledged document standards is the DITA [21]. DITA is a topic-based document architecture for authoring technical documents such as software manuals [1, 11]. It provides several types of *topics* by default, which are self-contained document structures that have certain communicative purposes, such as concept topic, task topic and reference topic. Table 1 shows a part of the machinery task topic, consisting of a set of content elements, such as **context** and **result**. Below the **steps** element, DITA further specifies detailed content elements, such as **substeps** and **tutorialinfo**. Although this structure guides us in composing necessary content in an organised manner, the applicability of DITA to automobile technical manuals is still unexamined.

Table 1. A part of the machinery task defined in DITA

Element	Description
prelreqs	Preliminary requirements: information the user must know before starting
context	Background information
steps	Main content: a series of steps
result	Expected outcome
example	Example that illustrates the task
closereqs	Steps to do after completion of current task

In the practical application of document structures, such as authoring aid and information retrieval, the core task is automatic recognition of document structures. This task can be mainly regarded as classifying a given text according to predefined content elements. In general, text classification tasks have long been addressed using both rule-based methods [13] and machine learning-based methods [24]. In recent years, neural network-based methods have attracted researchers [17, 19, 31]. However, deep neural networks generally require a large amount of data to learn. Because manual annotation of document structures involves much time and cost, it is difficult to obtain training data in many practical situations. Therefore, it is important to examine multiple machine learning methods to explore the feasible ways to build robust text classifiers with limited training data.

3 Investigation of Content Elements

3.1 Document Annotation

To clarify what types of content have been written in automobile repair manuals, based on the open coding method [8], we manually annotated selected text with content elements, i.e., indicative abstraction of the given content.

As the target data, we focused on instructional text about **removal** or **installation** of automobile parts, which are the dominant categories of automobile repair tasks.¹ We randomly extracted 62 documents (31 for **removal** and 31 for **installation**) for annotation to cover a wide range of automobile parts. Although parallel text data is available in English and Japanese, in this investigation, we used the Japanese version.

Figure 1 shows an example of annotation on a part of a document. The minimal text span for annotation is a clause, and we applied annotation to cover the entire target text without duplication. As there is no standard set of content elements in this domain, names and granularity of elements were first defined in an ad hoc manner and refined iteratively through annotation cycles. We also categorised identified elements to create a typology.

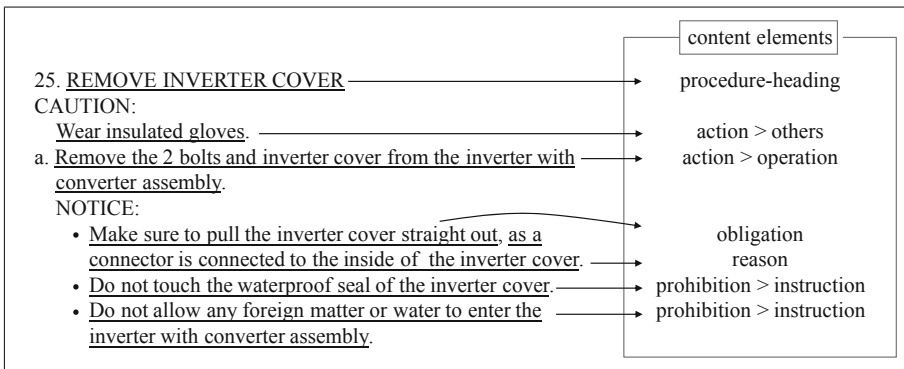


Fig. 1. Example of document annotation. For illustrative purposes, the corresponding English version is presented.

3.2 Typology of Content Elements

The annotation was conducted by one of the authors of the present paper, and the results were verified by another author. As a result of three cycles of annotation and overall verification, we identified 26 types of content elements that

¹ We used the PRIUS Repair Manual (November 2017 version) provided by Toyota Motor Corporation.

Table 2. Typology of content elements of automobile repair manuals

Content element	Token	Type	Example (English translation)
meta			
procedure-heading	764	746	1.REMOVE OUTER MIRROR
target	45	11	for RH Side
indication	14	3	The following procedure is for the LH side
same-procedure	110	20	Use the same procedure for the RH side and LH side
guide	17	17	The necessary procedures (adjustment, calibration, initialization, or registration) that must be performed after parts are removed and installed, or replaced during front drive shaft assembly removal/installation are shown below
confirmation	10	4	Be sure to read Precaution thoroughly before servicing
parts	19	4	The following functions are suspended when auxiliary battery terminal is disconnected/reconnected.
architecture			
if	2	2	The yaw rate and acceleration sensor is built into the airbag ECU assembly
condition	57	39	If the crankshaft position sensor has been struck or dropped,
event	107	22	(w/ Sliding Roof)
item	11	8	after replacing the heated oxygen sensor
start	2	2	just before installing the inverter with converter assembly
end	770	617	Engage the claw to install the clamp cover
action	40	30	Check that the cable is disconnected from the negative (-) auxiliary battery terminal
judgement			
others	8	5	Use tags or make a memo to identify the places to reconnect the brake lines
method	68	51	Install the clip within the range shown in the illustration
prohibition	101	64	Do not touch the engine, exhaust pipe or other high temperature components while the engine is hot
instruction			
obligation	38	15	Doing so may break the mirror surface
danger	16	13	Be sure to follow the procedure described in the repair manual,
recommendation	2	2	It is recommended to install a coil spring stopper belt as shown in the illustration
reason	40	19	Because the nut has its own stopper,
purpose	12	11	This operation is useful to prevent damage to the spiral cable.
reference	120	56	Using a rope or equivalent,
equipment	59	36	Torque: 45 N*m (459 kgf*cm, 33 ft.*lbf)
standard			
supplement	2	2	It is not necessary when replacing the rear disc
related-link	475	382	Click here General>INTRODUCTION>REPAIR INSTRUCTION >PRECAUTION

cover a total of 2913 text spans.² Table 2 presents the resulting typology of the content elements with their frequencies (token/type) and examples.

The content elements are broadly divided into four categories: **meta**, **concept**, **task**, and **reference**.

Meta deals with meta-level information that helps readers understand the content. **Procedure-heading** is a short summary of a small procedural step in the task. **Target** indicates parts of the automobile to be worked on in the procedure. **Indication** informs the reader of the procedure status. **Same-procedure** indicates that the targeted procedure can be conducted in the same way as the previously described procedure. **Guide** leads to another document (**post-task**) or description (**confirmation**).

Concept deals with descriptions of parts and tools, such as **function** and **architecture**, that are important for readers to understand the automobile repair task.

Task is a main content type about the repair task to be performed. **Condition** describes the situation as to **if** or **when** a certain procedure should be conducted. **Action** concisely indicates what should be done, which can be further subdivided according to the type of action: **operation**, **judgement** and **others**. **Explanation** provides detailed information of the task. **Reason** describes why the task is necessary, while **purpose** gives the intended result or gain. **Reference** includes information about **equipment** to be used, the **standard** to be followed, and **supplement** information about the procedure.

Related-link is a hyper link to another page of the repair manual, which has the fixed form ‘Click here <link>’.

Most of the content elements occur (both token and type) well below 100 times, demonstrating the difficulty in collecting sufficient labelled data through human annotation. Another notable observation is that for some elements, the number of types is greatly reduced from that of tokens, which means there is frequent repetition of the same text strings. For instance, the expression RH側はLH側と同様の手順で行う。(‘Use the same procedure for the RH side and LH side.’) was observed 12 times in the collected documents. The degree of textual repetition or formality should be considered to properly deal with linguistic characteristics.

4 Formalisation of Document Structure

In this section, we formalise a document structure suitable for automobile repair procedures. We first modify the DITA machinery task structure (see Table 1 in Sect. 2), and then we map the content elements obtained in Sect. 3 to the structure.

² The feasibility of this annotation scheme will be validated by measuring inter-rater agreement in future work.

4.1 Document Structure of Repair Manuals

The structure of automobile repair procedures can be defined by extending the DITA machinery task structure presented in Table 1. The left side of Table 3 describes the modified DITA structure. The DITA structure is recursive, i.e., the **cmd** in each **step** can have a nested element of **steps**.

The major extension is the addition of the following three elements: **attention**, **conditionalstep** and **followup**. **Attention** covers information about **warning** and **caution**, which are critically important for safely conducting automobile repair tasks. Hence, we separated **attention** from supplementary information (**info**). **Conditionalstep** can be used to describe a step in which implementation depends on a certain precondition. For example, ‘Remove Deck Floor Box RH (w/ Full Size Spare Tire)’ is a conditional step, where the operation (or **cmd**) ‘Remove Deck Floor Box RH’ is carried out only if the targeted automobile satisfies the **condition**, i.e., having ‘Full Size Spare Tire’. **Conditionalstep** is effective in modelling the conditional branching of a sequence of procedures, which is frequently observed in repair tasks. **Followup** is an additional step that may be necessary according to the result of the previous step.

Table 3. DITA-based document structure and content elements

Document structure based on DITA	Content element
attention warning	prohibition, obligation, confirmation
caution	
info	concept, reference, recommendation
steps step cmd	action, equipment, when
conditionalstep case condition	if
cmd	action, equipment, when
tutorialinfo	method, standard
attention warning	prohibition, obligation, event, action
caution	
followup	event, action
info	concept, reference, equipment, recommendation
postreq	post-task

4.2 Arrangement of Content Elements in Document Structure

The typology of the content elements presented in Table 2 does not specify the *arrangement*, i.e., in what order these elements should be compiled in a document. Here, we associate each of the content elements with the DITA-based document structure (Table 3). The content elements in the **meta** category, namely **procedure-heading**, **target**, **indication**, **same-procedure** and **guide**, are not associated with DITA elements because we assume these elements should be managed outside the DITA structure. It should also be noted that **reason** and **purpose** are not presented in Table 3 because they are general elements that appear in all parts of the document structure.

The DITA-based document structure specified with detailed content elements provides a guide to which content should be included and in what order when authoring a procedural repair manual.

5 Automatic Recognition of Content Elements

From the viewpoint of authoring aid, the formalised document structure can also be used for document diagnosis. The diagnosis process is broadly divided into the following two steps: (1) recognising content elements in text and (2) checking whether the arrangement of the elements conforms to the document structure.

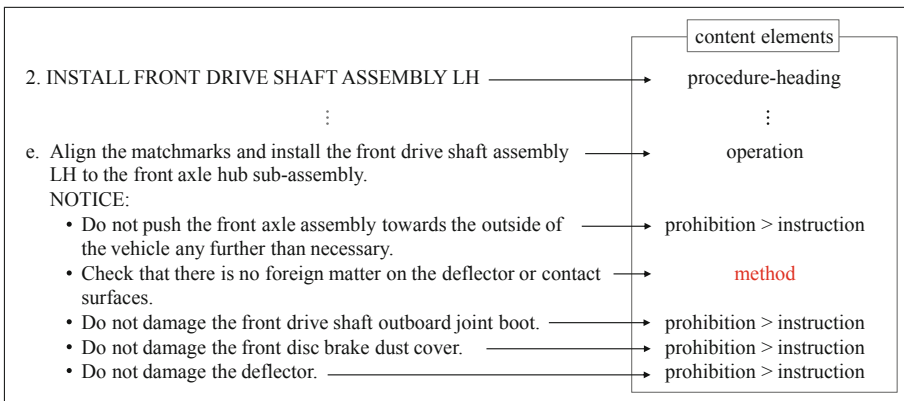


Fig. 2. Example of visualisation of content elements for document diagnosis

Figure 2 shows an example from the repair manual, where an appropriate content element is assigned to every sentence. Here, it can be seen that the **method** element is located in between the **prohibition > instruction** elements, which degrades the cohesion of text. In this section, we focus on the first step, i.e., automatically recognising content elements in text. Recognising content elements can be regarded as a text classification task. Although the minimal text span was a clause at the time of annotation (Sect. 3), we used sentences as units of classification. We implemented binary classifiers for each content element using multiple machine learning methods.

5.1 Experimental Data

To investigate the applicability of our methods to different domains of repair documents, we prepared two types of data: in-domain data and out-of-domain data. For the in-domain data, we used the same document set described in Sect. 3, which deals with the **removal** and **installation** of automobile parts. For the out-of-domain data, we randomly collected 10 documents from the repair manual

that deal with **inspection** of automobile parts. We annotated the documents with the content elements in the same manner as in Sect. 3. The content elements used for annotating the in-domain data (Table 2) are sufficient to annotate whole text spans in the out-of-domain data, which suggests high applicability of our content element set.

We eliminated duplicated sentences from the two data sets. As a result, we obtained 1814 sentences of in-domain data and 183 sentences of out-of-domain data. For each content element classifier, sentences that included at least one text span with the element were used as positive examples, while the remaining sentences were used as negative examples. We did not implement classifiers of content elements with less than 10 sentences as it is difficult to draw reliable insights from such small data.

5.2 Machine Learning Methods

We examined the following four machine learning methods to implement binary classifiers for each content element.

SVM Support vector machine [9] with word-level uni-grams and bi-grams.³

RF Random forest [4] with word-level uni-grams.

MLP Multilayer perceptron [14] with word2vec vectors trained on Japanese Wikipedia.⁴ We chose the best hidden layer size out of 100, 200 and 300 through grid search.

CNN Convolutional neural network [19] with word2vec vectors trained on Japanese Wikipedia. We used filter windows of 3, 4 and 5 with 100 feature maps each, a dropout rate of 0.5, a mini-batch size of 8, and 20 epochs.

We built the models using in-domain data and then applied them to the out-of-domain data without fine-tuning.

5.3 Overall Results

Table 4 shows the overall precision, recall and F-scores⁵ for both the in-domain and out-of-domain data. For the in-domain data, SVM consistently outperformed the others, except in the micro-averaged precision of RF. For the out-of-domain data, SVM and CNN achieved better scores than the others; in particular, the precision scores of CNN were higher than those of the others by a wide margin. The drops in the scores from the in-domain to out-of-domain in the CNN were

³ For Japanese word segmentation, MeCab ver.0.996 (<http://taku910.github.io/mecab>) and mecab-ipa-NEologd (<https://github.com/neologd/mecab-ipadic-neologd>) were used. Although we tried three settings of features (uni-grams, bi-grams and uni-grams+bi-grams), we report only the uni-grams+bi-grams with the highest score in this paper.

⁴ We used publicly available pre-trained vectors. http://www.cl.ecei.tohoku.ac.jp/~m-suzuki/jawiki_vector/.

⁵ The F-score is the harmonic mean value of the precision and recall.

Table 4. Overall results for the four classifiers. For the in-domain data, the best models of SVM, RF and MLP were selected through grid search, and all models were evaluated by stratified 5-fold cross validation. For the out-of-domain data, the best models trained on the in-domain data were used. Macro-average computes the metrics separately for each element and takes the average. Micro-average totals the results of all elements to compute the average.

		micro-average			macro-average		
		P	R	F	P	R	F
in-domain	SVM	0.964	0.937	0.951	0.921	0.765	0.816
	RF	0.967	0.888	0.926	0.861	0.622	0.694
	MLP	0.885	0.799	0.840	0.685	0.496	0.549
	CNN	0.939	0.866	0.901	0.836	0.645	0.705
out-of-domain	SVM	0.703	0.734	0.718	0.758	0.762	0.687
	RF	0.687	0.520	0.592	0.850	0.553	0.542
	MLP	0.650	0.452	0.533	0.741	0.502	0.523
	CNN	0.870	0.605	0.713	0.935	0.642	0.687

smaller than those in the SVM. This implies that the CNN is robust to different domains.

In the following two sections, with a focus on the SVM and CNN, we further investigate the results for the in-domain and out-of-domain data, respectively.

5.4 Analysis of In-Domain Data

Table 5 shows the evaluation scores for each of the content elements for the in-domain data. SVM and CNN both obtained almost perfect scores for the **meta** elements. The sentences in these content elements contain fixed phrases that do not appear in the other content elements, such as the phrase ... 側と同じ要領で... ('... the same procedure as for ...') in the sentences of the **same-procedure** element.

On the contrary, the scores of the **method**, **reason** and **purpose** elements were lower than those of the other elements. One of the reasons for this is the ambiguity of particular words. Table 6 shows example sentences that have the same word **ため**, which can be used to mean either **reason** ('because') or **purpose** ('in order to'). To cope with such ambiguity, we must consider the detailed features of linguistic elements that connect to the words. For example, the verb preceding **ため** is a strong indicator for distinguishing its meaning; **ため** tends to mean **reason** when connected to state verbs, and it tends to mean **purpose** when connected to action verbs. Designing proper features using part-of-speech information and external linguistic knowledge can help improve classification performance.

Another reason is the lack of context. For example, as the **action** elements and **method** element have similar linguistic and semantic characteristics, even

Table 5. Results for the in-domain data (**bold:** F > 0.9, *italics:* F < 0.5)

Content element	#	SVM			CNN		
		P	R	F	P	R	F
meta>procedure-heading	603	0.995	1.000	0.998	0.952	0.975	0.963
meta>target	35	1.000	1.000	1.000	1.000	1.000	1.000
meta>same-procedure	20	1.000	1.000	1.000	1.000	0.900	0.933
task>condition>if>event	36	0.853	0.643	0.720	0.893	0.500	0.635
task>condition>if>item	75	0.988	0.973	0.980	0.972	0.920	0.945
task>action>operation	518	0.933	0.969	0.951	0.934	0.892	0.912
task>action>judgement	29	1.000	0.833	0.902	0.967	0.800	0.860
task>explanation>method	54	0.806	0.496	0.595	0.860	<i>0.367</i>	0.506
task>explanation>prohibition>instruction	64	0.985	0.938	0.959	0.882	0.749	0.798
task>explanation>prohibition>danger	15	1.000	0.667	0.780	0.800	0.400	0.520
task>explanation>obligation	12	0.800	0.333	<i>0.467</i>	0.467	0.233	<i>0.280</i>
task>reason	20	0.950	0.600	0.706	0.833	0.400	0.528
task>purpose	10	0.600	0.300	<i>0.400</i>	0.200	0.100	<i>0.133</i>
task>reference>equipment	101	0.979	0.950	0.964	0.946	0.792	0.859
macro-average		0.921	0.765	0.816	0.836	0.645	0.705
micro-average		0.964	0.937	0.951	0.939	0.866	0.901

Table 6. Example sentences that include the word *ため*

reason	リヤディスクブレーキパッドサポートプレートは形状がそれぞれ異なるため、取り付けの際に場所が分かるように識別マークを付けておく。 (Because each rear disc brake pad support plate has a different shape, be sure to put an identification mark on each rear disc brake pad support plate so that it can be reinstalled to its original position.)
purpose	スパイラルケーブル断線防止のため、ステアリングホイールが回転しないようにシートベルトを通して保持する。 (In order to prevent the spiral cable from breaking, the steering wheel is held through the seat belt so as not to rotate.)

Table 7. Results for the out-of-domain data (**bold:** F > 0.9, *italics:* F < 0.5)

Content element	#	SVM			CNN		
		P	R	F	P	R	F
meta>procedure-heading	29	0.509	1.000	0.674	0.963	0.897	0.929
task>condition>if>event	37	1.000	0.865	0.928	0.957	0.595	0.733
task>action>operation	57	0.671	0.895	0.767	0.754	0.754	0.754
task>action>judgement	38	0.667	0.053	<i>0.098</i>	1.000	0.026	<i>0.051</i>
task>reference>equipment	16	0.941	1.000	0.970	1.000	0.938	0.968
macro-average		0.758	0.762	0.687	0.935	0.642	0.687
micro-average		0.703	0.734	0.718	0.870	0.605	0.713

human annotators cannot perfectly distinguish the **action** and **method** elements without context information. To take context information into account,

it is useful to employ sequential models of neural networks that encode not only the sentence to be classified but also the surrounding sentences [12, 17, 25].

5.5 Analysis of Out-of-domain Data

Table 7 shows the results for the out-of-domain data for the SVM and CNN. In particular, the recall values of **judgement** are no more than 0.1. The major factor is supposed to be the difference in vocabulary (particularly verbs) between the training data and the test data, i.e., the in-domain and out-of-domain data. For example, the verb 点検する (‘inspect’) never appears in the in-domain data, but it appears 27 times in the out-of-domain data.

For the **equipment** element, we obtained high evaluation scores for both SVM and CNN. This is because the **equipment** element has similar linguistic forms across domains, such as ... を用いて (‘Using ...’) and ... で (‘by ...’).

As we pointed out in Sect. 5.3, the CNN is less affected by domain differences. This is attributable to the word representation used as the input for the classifiers. While the discrete bag-of-words was used in the SVM, distributed representations of words based on the pre-trained word2vec vectors were used in the CNN, which helped improve its ability to handle unknown words and its applicability to the out-of-domain data.

6 Conclusion and Future Work

In this study, with a focus on procedural documents of automobile repair manuals, we revealed the content of documents in the form of content elements and formalised document structure. Through manual annotation, we identified 26 content elements of four categories: **meta**, **concept**, **task** and **link**. We formulated the document structure by adding **attention**, **conditionalstep** and **followup** elements to the DITA structure in consideration of the specific issues of automobile repair manuals, such as the risk of repair tasks and conditional branching of procedures. Although the document structure was created by analysing only the documents of ‘removal’ and ‘installation’, we assume that it is widely applicable to many procedural repair documents.

We also examined the feasibility of automatic recognition of content elements through a sentence classification experiment using four machine learning methods. For the in-domain data, the SVM achieved a high performance with a micro-averaged F-score more than 0.9, while for out-of-domain data, the SVM and CNN achieved comparable results with micro-averaged F-scores of approximately 0.7. In-depth analyses of the results suggested major factors for the low classification performance, such as the ambiguity of words and the lack of context. Our results also implied that distributed word representations, such as word2vec, can help improve the ability to handle words that are not observed in the training data, leading to model robustness to different domains.

The document structure we defined is useful for various practices and applications, such as authoring, summarisation, and information retrieval. In addition,

automatic recognition of the document elements can be applied to document diagnosis of automobile repair manuals.

In future work, based on our analyses described above, we plan to refine the sentence classifiers by (1) designing rich features to capture the detailed characteristics of ambiguous words, (2) using contextual information surrounding the text to be classified, and (3) seeking and implementing better word representations. In parallel with this, we intend to conduct finer-grained recognition of the content element span in documents. Finally, we will also extend our document formalisation framework and automatic classification methods to others domains and languages to cover a wide range of technical documents.

Acknowledgement. This work was partly supported by JSPS KAKENHI Grant Numbers 17H06733 and 19H05660. The automobile manuals used in this study were provided by Toyota Motor Corporation.

References

1. Bellamy, L., Carey, M., Schlotfeldt, J.: DITA Best Practices: A Roadmap for Writing, Editing, and Architecting in DITA. IBM Press, Upper Saddle River (2012)
2. Bhatia, V.K.: Worlds of Written Discourse: A Genre-Based View. Continuum International, London (2004)
3. Biber, D., Conrad, S.: Register, Genre, and Style. Cambridge University Press, New York (2009)
4. Breiman, L.: Random forests. *Mach. Learn.* **45**(1), 5–32 (2001)
5. Brett, P.: A genre analysis of the results section of sociology articles. *Engl. Specif. Purp.* **13**(1), 47–59 (1994)
6. Bunton, D.: The structure of PhD conclusion chapters. *J. Engl. Acad. Purp.* **4**(3), 207–224 (2005)
7. Carey, M., Lanyi, M.M., Longo, D., Radzinski, E., Rouiller, S., Wilde, E.: Developing Quality Technical Information: A Handbook for Writers and Editors. IBM Press, Upper Saddle River (2014)
8. Corbin, J., Strauss, A.: Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory, 4th edn. Sage Publications, Los Angeles (2014)
9. Cortes, C., Vapnik, V.: Support-vector networks. *Mach. Learn.* **20**(3), 273–297 (1995)
10. Cross, C., Oppenheim, C.: A genre analysis of scientific abstracts. *J. Documentation* **62**(4), 428–446 (2006)
11. Day, D., Priestley, M., Schell, D.: Introduction to the Darwin Information Typing Architecture: Toward portable technical information (2005). <http://www.ibm.com/developerworks/xml/library/x-dita1/x-dita1-pdf.pdf>
12. Dernoncourt, F., Lee, J.Y.: PubMed 200k RCT: a dataset for sequential sentence classification in medical abstracts. In: Proceedings of the 8th International Joint Conference on Natural Language Processing (IJCNLP), Taipei, Taiwan, pp. 308–313 (2017)
13. Hayes, P.J., Andersen, P.M., Nlrenburg, I.B., Schmandt, L.M.: TCS: a shell for content-based text categorization. In: Proceedings of the 6th Conference on Artificial Intelligence for Applications (CAIA), Santa Barbara, California, USA, pp. 320–326 (1990)

14. Hinton, G.E.: Connectionist learning procedures. *Artif. Intell.* **40**(1), 185–234 (1989)
15. Horn, R.E.: *Mapping Hypertext: The Analysis, Organization, and Display of Knowledge for the Next Generation of On-Line Text and Graphics*. Lexington Institute, Arlington (1989)
16. Horn, R.E.: Structured writing as a paradigm. In: Romiszowski, A., Dills, C. (eds.) *Instructional Development: State of the Art*. Educational Technology Publications, Englewood Cliffs (1998)
17. Jin, D., Szolovits, P.: Hierarchical neural networks for sequential sentence classification in medical scientific abstracts. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Brussels, Belgium, pp. 3100–3109 (2018)
18. Kando, N.: Text-level structure of research articles and its implication for text-based information processing systems. In: *Proceedings of the 19th British Computer Society Annual Colloquium on Information Retrieval Research (BCS-IRSG)*, Aberdeen, UK, pp. 68–81 (1997)
19. Kim, Y.: Convolutional neural networks for sentence classification. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, pp. 1746–1751 (2014)
20. Maswana, S., Kanamaru, T., Tajino, A.: Move analysis of research articles across five engineering fields: what they share and what they do not. *Ampersand* **2**, 1–11 (2015)
21. OASIS: Darwin Information Typing Architecture (DITA) Version 1.3. <http://docs.oasis-open.org/dita/dita/v1.3/dita-v1.3-part3-all-inclusive.html>
22. Reiter, E., Dale, R.: *Building Natural Language Generation Systems*. Cambridge University Press, Cambridge (2000)
23. Rubens, P. (ed.): *Science and Technical Writing: A Manual of Style*. Routledge, New York (2001)
24. Sebastiani, F.: Machine learning in automated text categorization. *ACM Comput. Surv.* **34**(1), 1–47 (2002)
25. Song, X., Petrak, J., Roberts, A.: A deep neural network sentence level classification method with context information. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Brussels, Belgium, pp. 900–904 (2018)
26. Swales, J.M.: *Genre Analysis: English in Academic and Research Settings*. Cambridge University Press, Cambridge (1990)
27. Swales, J.M.: *Research Genres: Explorations and Applications*. Cambridge University Press, Cambridge (2004)
28. Swales, J.M., Freak, C.B.: *Academic Writing for Graduate Students: Essential Tasks and Skills*, 3rd edn. University of Michigan Press, Ann Arbor (2012)
29. Tessuto, G.: Generic structure and rhetorical moves in English-language empirical law research articles: sites of interdisciplinary and interdiscursive cross-over. *Engl. Specif. Purp.* **37**, 13–26 (2015)
30. Teufel, S., Moens, M.: Summarizing scientific articles: experiments with relevance and rhetorical status. *Comput. Linguist.* **28**(4), 409–445 (2002)
31. Zhang, Y., Wallace, B.C.: A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. In: *Proceedings of the 8th International Joint Conference on Natural Language Processing (IJCNLP)*, Taipei, Taiwan, pp. 253–263 (2017)



Weakly-Supervised Relation Extraction in Legal Knowledge Bases

Haojie Huang, Raymond K. Wong^(✉), Baoxiang Du, and Hae Jin Han

School of Computer Science and Engineering, University of New South Wales,
Sydney, NSW 2052, Australia

{haojie.huang,haejin.han}@student.unsw.edu.au, wong@cse.unsw.edu.au,
b.du@unsw.edu.au

Abstract. Intelligent legal information systems are becoming popular recently. Relation extraction in massive legal text corpora such as precedence cases is essential for building knowledge bases behind these systems. Recently, most works have applied deep learning to identify relations between entities in text. However, they require a large amount of human labelling, which is labour intensive and expensive in the legal field. This paper proposes a novel method to effectively extract relations from legal precedence cases. In particular, relation feature embeddings are trained in an unsupervised way. With limited labelled data, the proposed method is shown to be effective in constructing a legal knowledge base.

Keywords: Relation extraction · Weakly supervised learning

1 Introduction

With the increasing popularity of artificial intelligence for legal inference (e.g., ROSS¹), information extraction techniques are strongly desired to construct a legal knowledge base automatically. Relation extraction is one of the most important types of information extraction, which is the process of generating relational data from natural language text.

Apart from the traditional rule-based methods, supervised learning is the most popular method to enable automatic relation extraction [13, 19]. Although these methods can achieve high accuracy, they barely perform well without a large amount of high-quality labelled data, which is extremely expensive to produce in practice. Instead of fully supervised learning methods, Ren et al. introduce a framework CoType [16], which applies a distant supervision scheme to solve the training data problem. Distant supervision methods make use of a third-party knowledge base to replace manual labelling. Therefore, the label data may contain a lot of noise. To solve the problem, they accomplish the joint learning of entity and relation embedding to overcome the noisy data. However,

¹ <https://rossintelligence.com>.

the drawback of the CoType framework comes from the high dependency of its vector embeddings. Moreover, its dependencies on a third-party knowledge base generate biased labels and make it hard to transfer to real applications.

To tackle these problems, we propose a new method **RF-Extract**, which consists of an unsupervised training phase and a weakly-supervised extraction phase for relation extraction. Compared to the CoType, RF-Extract is much simpler in training. It takes a set of relation features and the two related entities to train relation embeddings. In the extraction phase, we introduce methods for selecting seed examples for weakly-supervised learning. We implement our prototype on an unlabelled legal dataset and get a remarkable result by only having 400 sentences labelled for training.

2 Related Work

Traditional researches on legal knowledge bases are about legal expert systems, which focus on constructing rule-based systems for legal reasoning and decision making [4,20]. The construction of such systems depends on existing ontologies predefined by human experts. For example, CORBS [4] utilise the legal core ontology (LKIF-Core) [6] for guiding the extraction process. Gupta et al. [5] makes use of the DOLCE+DnS Ultralite (DUL) ontology and design an entity-relation model for storing the extracted structures from judicial decision documents. More recent systems attempt to obtain semantic information about legal cases without applying any ontologies. This can be achieved by natural language processing techniques such as word embeddings. For example, Nejadgholi et al. [14] implement a system for semantic search on Canadian immigration cases. Their system performs word2vec [11] on a large scale of legal cases, and trains a legal fact detecting classifier and a text vectoriser in a supervised manner. Although the system allows semantic search, it does not capture the relation structure between legal entities. To overcome the problem, our RF-Extract focuses on the task of relation extraction.

Many recent advances in relation extraction utilise deep learning. The two most popular models are Convolutional Deep Neural Network (CNN) [8,19] and Long Short Term Memory (LSTM) [13]. These methods are classified as supervised learning, and they require a large amount of labelled training data. In comparison, distant-supervised relation extraction scheme eliminates the process of a manually labelling training set [12]. It refers to use higher-level, less precise weak label distribution to replace human supervision. Such labelling commonly incorporates heuristic rules or existing resources [15]. For example, CoType [16] is a distant supervision approach that generates a training set of relations and entity pairs using Freebase [1] and WordNet [18]. The generated labelled data was then translated into a graph model for further training. Li et al. used structure perceptron with efficient beam search to train entity and relation mentions simultaneously [9]. In order to address noisy labelled data, multi-instance learning [17] was adopted to recognise true label among noisy labels. Hoffmann et al. combined sentence level features with corpus level components to learn overlapping relations through multi-instance learning [7].

3 Our Approach

RF-Extract consists of two phases: the training phase and the extraction phase. The approach starts with the preprocessing by filtering the AustLII dataset and keeps a large set of sentences that are likely to contain desired entities and relations. We then pass these sentences through an entity extractor to annotate entities such as people, assets and locations as the unlabelled set, on which the training phase is performed. A small portion of this unlabelled set is randomly selected for human experts to label the relation types, and it is used as the training set in the extraction phase. We perform the random centroid method to select the seeds. A demonstration of our approach is available online².

3.1 The Training Phase

RF-Extract consists of a training phase and an extraction phase. The training phase model is extended from skip-gram word2vec [11]. More similar to CoType, we consider the connection between entity mentions and relation features. More formally, the connection can be regarded as the conditional probability of a relation feature f_i given two entities e_{j1} and e_{j2} are the two entities in a sentence S_j . Hence, we train the relation feature embeddings by optimising the objective $p(\mathbf{f}_i | \mathbf{e}_{j1}, \mathbf{e}_{j2}, S_j)$, and we adopt the negative sampling strategy to sample false feature-entity mappings to achieve a more efficient optimisation process. The final objective function can be derived as Eq. (1).

$$p(\mathbf{f}_i | \mathbf{e}_{j1}, \mathbf{e}_{j2}, S_j) = \log \sigma(\mathbf{z}_j^T \mathbf{f}_i) + \sum_{v=1}^V \mathbb{E}_{F'_i \sim P_n(z)} [\log \sigma(-\mathbf{z}_j^T \mathbf{f}_{i'})] \quad (1)$$

where $\sigma(x)$ is the sigmoid function and \mathbf{f}_i is a relation feature encoded as a one-hot vector. The entity pair \mathbf{e}_{j1} and \mathbf{e}_{j2} , which are the one-hot entity vectors, can be further concatenated into a vector z_j . The first term in the equation models the observed co-occurrence of relation features and entities, and the second term performs negative samples from unrelated entities.

Relation features such as “verb position”, “POS tagging of the entity words” and “numerical value” can be done automatically in the preprocessing step. We assumed the entities are already extracted by other pre-trained models. In this way, we don’t require any manual labelling to train this relation feature embedding model, and hence, the training phase can be regarded as an unsupervised learning process. At the end of the training phase, each relation feature has been mapped into a vector space with dimension d and can be expressed as $\mathbf{f}_i \in \mathbb{R}^d$. Since every 2 entity pairs e_{j1} and e_{j2} can generate a set of relation features R_S , the relation vector between the 2 entities $rel(e_{j1}, e_{j2})$ can be represented by the mean of all \mathbf{f}_i .

² <http://dragon26.cse.unsw.edu.au/fc>.

3.2 The Extraction Phase

Relation Type Inference. Relation type inference is achieved by a semi-supervised approach. A set of seeds (a sentence, two entities, and a relation) is selected from the training set with labelled relation types. For a testing example (a known sentence and 2 extracted entities, but an unknown relation type), we first extract its relation features $\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_n$ automatically followed by computing its relation vector. Finally, we apply kNN to look up its neighbours among the seed examples and claim the closest one (to be precise, the one chosen from voting from a set of closest examples) as the final relation type. Since the number of seed examples is limited, their quality is the key to the accuracy of the algorithm. To solve this problem, we propose several methods for selecting high-quality seed examples.

Seed Selection Methods. Our seed selection methods are based on cluster centroids, which refers to the geometric centre of a set of points in a high dimensional vector space.

- **Random Centroid.** Randomly pick a vector from each relation type as the centroid.
- **Average Centroid.** Average all the vectors from a relation type and use the result as the centroid
- **Squared Average Centroid.** Positive and negative numbers may cancel each other when calculating the average vector. The squared average mechanism firstly converts all number to positive by squaring them, followed by calculating an average centroid based on the positive vectors.

In practice, we generate centroids v_c for each of the relation types. Then we select an equal amount of k training samples for each relation type based on their similarity to centroid v_c . In terms of similarity calculation, we also introduce a dimension elimination mechanism where we found that not all the dimensions are helpful for selecting seed samples. A parameter p (percent) is introduced for removing some of the vector dimensions when computing similarities. For example, $p = 10\%$ refers to only the first 10% vector dimensions are considered for similarity calculation.

4 Experiments

4.1 Datasets and Setup

Our legal knowledge base is built using data from **AustLII**³, an open-access legal database. The extraction of entities is done through DBpedia Spotlight [3] and Stanford CoreNLP [10] with some predefined heuristics. For the experiments, we focus on the subset of AustLII, namely the AustLII-FC dataset, by extracting structured information from Family Court legal cases between 1980 and 2018. Our final AustLII-FC dataset contains over 6M sentences from 15,827 legal cases.

³ <http://www.austlii.edu.au/>.

4.2 Relation Extraction Prototype on AustLII-FC

Our experiments show that random seed selection method produces reasonable accuracy. This suggests a small set of randomly selected sentences labelled manually is sufficient for training in practice. Therefore, we label 400 randomly picked sentences from the AustLII-FC dataset and consult legal professionals to label them from 9 specialised legal relations for training. The remaining sentences with extracted entities are staying unlabelled and are used for testing. All these 400 labelled relations are used as the seeds. For all unlabelled entity pairs, we calculate their cosine distances to the labelled relations.

Table 1. Sample extractions using RF-Extract on the AustLII-FC dataset.

Sentence	Entity 1	Entity 2	Relation Type	Distance
He did not work for a period of two months from March 1989 during which time he and the mother jointly cared for the child	the mother	the child	person/person/guardian	0.25521
That the husband take steps to ensure that there is paid direct to the wife maintenance for the said child in the sum of \$40 per week, otherwise usual order.	the wife	\$40	person/money/receive	0.32353
"At the conclusion of the week, and pending the resumption of the hearing on 1 April 1980, I ordered that the wife have the sole use and occupation of the matrimonial home ."	the wife	the matrimonial home	person/object/asset	0.41574
The issue in these proceedings is not whether or not the husband has been guilty of a serious criminal offence, namely, the sexual abuse of one or more of the children	the husband	the children	person/person/abuse	0.44209
"174(8) specifying \$255.60 as the amount to be paid by the wife to the Solicitors for costs and indicating that no costs of the taxation were included therein."	the wife	\$255.60	person/money/pay	0.44258

Table 1 presents some sample relation extraction results using the AustLII-FC dataset. The cosine distance between each testing sample and their nearest seed is calculated in the distance column. A shorter distance indicates a more accurately extracted relation type between two entities. In the first example, the true relation type between "**the mother**" is predicted to be the guardian of "**the child**". This prediction is accurate since the entity pair shares many features with entity pair "**the husband**" and "**the child**" in sentence "In April 2012 the wife went to the US for 10 days in the school holidays during which time the husband cared for the children." For example, the first entity appeared before the second entity, the connecting word for two relations are both "cared for".

4.3 A Family Law Knowledge Base Case Study

In the legal domain, a speedier decision simulator which makes a systematised resolution for each dispute is desired. A decision simulator generates decisions

by discovering relevant information from past cases. However, most of the legal documents are stored in an unstructured manner, and there are no universal standards of recording in court files in many countries [2]. To allow comparing past cases, a structured knowledge graph of the legal cases are required. Our method can then be adopted to extract relations and create such a structured output from these documents.

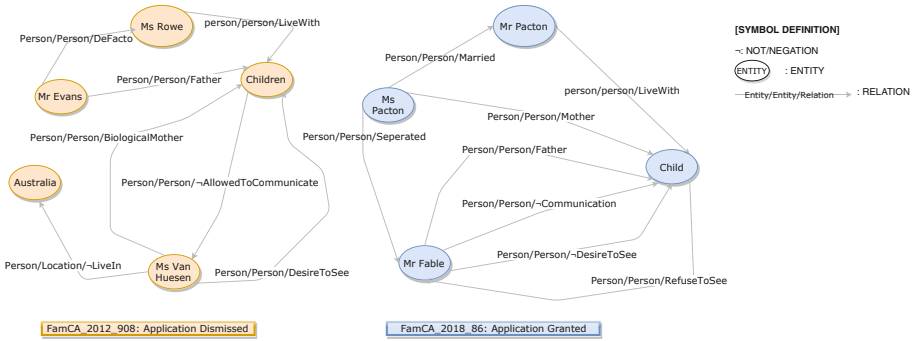


Fig. 1. Two knowledge graph fragments generated from the AustLII-FC dataset.

A knowledge graph is formed with a large number of relations. In the previous section, we have Table 1 showing the sample relation output extracted from the AustLII documents. When more relations are discovered, they can be visualised in a graphical format. For instance, Fig. 1 is an excerpt from two legal cases in a knowledge graph format. Both cases are aiming to decide whether the application of child adoption should be granted or dismissed. In these two cases, the relation between children/child and biological mother/father is one of the main factors when deciding the final judge. Similarly, a decision simulation system may also compare relations such as “**Person/Person/DesireToSee**” with past cases when simulating the final judgement.

5 Conclusion

In this paper, we have presented a novel method, RF-Extract, to extract and classify relations in legal knowledge bases. It works using feature embedding and is weakly supervised that requires a small amount of labelled examples as seeds. We use seed selection methods to help the RF-Extract gain a high accuracy in specialised domains such as legal domain. We believe that all these advantages allow our proposed framework effectively applied to other real-world datasets, in addition to legal data. For ongoing work, we are investigating n -ary relation extraction, which enriches the details in a complex knowledge graph.


References

1. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: a collaboratively created graph database for structuring human knowledge. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, pp. 1247–1250. ACM (2008)
2. Burstynier, N., Sourdin, T., Liyanage, C., Ofoghi, B., Zeleznikow, J.: Using technology to discover more about the justice system, vol. 44, p. 1. HeinOnline (2018)
3. Daiber, J., Jakob, M., Hokamp, C., Mendes, P.N.: Improving efficiency and accuracy in multilingual entity extraction. In: Proceedings of the 9th International Conference on Semantic Systems (I-Semantics) (2013)
4. El Ghosh, M., Naja, H., Abdulrab, H., Khalil, M.: Towards a legal rule-based system grounded on the integration of criminal domain ontology and rules. *Proc. Comput. Sci.* **112**, 632–642 (2017)
5. Gupta, A., et al.: Toward building a legal knowledge-base of Chinese judicial documents for large-scale analytics. In: Legal Knowledge and Information Systems (2017)
6. Hoekstra, R., Breuker, J., Di Bello, M., Boer, A., et al.: The LKIF core ontology of basic legal concepts. *LOAIT* **321**, 43–63 (2007)
7. Hoffmann, R., Zhang, C., Ling, X., Zettlemoyer, L., Weld, D.S.: Knowledge-based weak supervision for information extraction of overlapping relations. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, vol. 1. pp. 541–550. Association for Computational Linguistics (2011)
8. Kalchbrenner, N., Grefenstette, E., Blunsom, P.: A convolutional neural network for modelling sentences. arXiv preprint [arXiv:1404.2188](https://arxiv.org/abs/1404.2188) (2014)
9. Li, Q., Ji, H.: Incremental joint extraction of entity mentions and relations. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), vol. 1, pp. 402–412 (2014)
10. Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S.J., McClosky, D.: The Stanford CoreNLP natural language processing toolkit. In: Association for Computational Linguistics (ACL) System Demonstrations, pp. 55–60 (2014). <http://www.aclweb.org/anthology/P/P14/P14-5010>
11. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, pp. 3111–3119 (2013)
12. Mintz, M., Bills, S., Snow, R., Jurafsky, D.: Distant supervision for relation extraction without labeled data. In: Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, vol. 2, pp. 1003–1011. Association for Computational Linguistics (2009)
13. Miwa, M., Bansal, M.: End-to-end relation extraction using LSTMs on sequences and tree structures. arXiv preprint [arXiv:1601.00770](https://arxiv.org/abs/1601.00770) (2016)
14. Nejadgholi, I., Bougueng, R., Witherspoon, S.: A semi-supervised training method for semantic search of legal facts in Canadian immigration cases. In: JURIX, pp. 125–134 (2017)
15. Ratner, A., Bach, S., Varma, P., Ré, C.: Weak supervision: the new programming paradigm for machine learning. Hazy research

16. Ren, X., et al.: CoType: joint extraction of typed entities and relations with knowledge bases. In: Proceedings of the 26th International Conference on World Wide Web, pp. 1015–1024. International World Wide Web Conferences Steering Committee (2017)
17. Wu, J., Zhu, X., Zhang, C., Cai, Z.: Multi-instance multi-graph dual embedding learning. In: 2013 IEEE 13th International Conference on Data Mining (ICDM), pp. 827–836. IEEE (2013)
18. Yosef, M.A., Bauer, S., Hoffart, J., Spaniol, M., Weikum, G.: HYENA: hierarchical type classification for entity names. In: Proceedings of COLING 2012: Posters, pp. 1361–1370 (2012)
19. Zeng, D., Liu, K., Lai, S., Zhou, G., Zhao, J.: Relation classification via convolutional deep neural network. In: Proceedings of COLING 2014, The 25th International Conference on Computational Linguistics: Technical Papers, pp. 2335–2344 (2014)
20. Zurek, T.: Conflicts in legal knowledge base. *Found. Comput. Decis. Sci.* **37**(2), 129–145 (2012)



Social Media Aware Virtual Editions for the Book of Disquiet

Duarte Oliveira¹, António Rito Silva¹ , and Manuel Portela² 

¹ ESW - INESC-ID, Instituto Superior Técnico, University of Lisbon,
Lisbon, Portugal

{duarte.f.oliveira,rito.silva}@tecnico.ulisboa.pt

² CLP-Centre for Portuguese Literature, University of Coimbra, Coimbra, Portugal
mportela@fl.uc.pt

Abstract. Automatically collecting citations on social media about a literary work can provide interesting feedback about its current impact and relevance. The *Book of Disquiet* by Fernando Pessoa is made of a set of fragments put together by its editors. The *LdoD Archive* is a collaborative digital archive for Fernando Pessoa's *Book of Disquiet* (LdoD) that supports the creation of virtual editions by the archive users, simulating the process followed by four editors of the book, who made different decisions on the number and order of fragments to include in their edition.

In this paper we describe the design and implementation of a solution to collect citations from the *Book of Disquiet* by Fernando Pessoa on Twitter. We explore the affordances of the *LdoD Archive*, such that it becomes sensitive to its citations on Twitter. Therefore, we propose a new model that uses an algorithm for the automatic collection of citations, allowing the automatic generation of virtual editions of the book, which reflect its impact according to Twitter posts.

This Social Media Aware LdoD Archive supports a new set of features, such as: the collection of citations from Twitter, their transformation into annotations of the fragments belonging to a virtual edition, the visualization of the meta information associated with a citation in the context of the cited fragment, and the customization of the Social Aware virtual editions.

Keywords: Digital humanities · Automatic citation collection · Digital archive

1 Introduction

The *LdoD Archive*¹ explores the fragmentary and unfinished structure of Fernando Pessoa's *Book of Disquiet* (*LdoD* from the Portuguese title *Livro do Desassossego*) to foster a collaborative environment where users can experiment with

¹ Publicly available at <https://ldod.uc.pt>.

the creation of their own virtual editions [5, 6]. Besides representing four well-known experts' editions of the book, the *LdoD Archive* enables the platform users to select fragments for inclusion in their virtual editions and, additionally, to annotate and classify them according to a taxonomy. Both, annotations and classifications, can be applied to a fragment as a whole or to a part of it, for instance, a complete sentence or just a couple of words.

Citations from the *LdoD* in social media are frequent due to the appeal of the *LdoD* for general readers. The goal of this research is to explore how the *LdoD Archive* can be extended to incorporate those citations in social media so that they are stored in the Archive making it possible to access the sources of citation through the cited fragments. In some sense, we intend to make the Archive sensitive to social media. This automatic collection of social media citations is part of the component dedicated to the reception of the *LdoD*, which includes both the critical reception by expert scholars (in reviews, essays, and journal articles) and the online reception by general readers.

In the next section we describe other approaches on the automatic collection of citations, then in Sect. 3 we describe the *LdoD Archive* domain model extension, and the pipeline for collection and classification of citations. Section 4 presents the new functionalities of the *LdoD Archive* that take advantage of the automatic collection of citations. Finally, Sect. 5 evaluates the performance of the collection pipeline, and Sect. 6 presents the conclusions.

2 Related Work

Using social media as a source of information for impact analysis is frequently made, for instance, in order to infer the impact of scientific publications [3, 7, 8].

This same approach has been applied in the context of literary works, to identify tweets that cite parts of a work [1], through a process which defines a pipeline to collect tweets using a set of keywords, and do citation detection using Lucene². However, they only get a recall value of 59%, which reflects a large number of false negatives. Therefore, we intend to leverage on this pipeline to obtain a better recall by developing a flexible substring matching algorithm to even consider substrings of what people write in a tweet, which can be valid *LdoD* citations. This flexibility is important because users usually write additional information in the beginning or end of their tweets and posts. Besides that, users frequently misspell some words.

String matching algorithms, such as Boyer-Moore string-search algorithm [2], do not provide a feasible solution because we want to be rigorous but also tolerant enough. The Jaro-Winkler distance [9] measures the similarity score between two strings by counting the minimum number of operations required to transform one string into another. By defining a certain level of variation, this algorithm can be adapted to our goals.

² <https://lucene.apache.org>.

3 Automatic Collection and Citation Generation

The result of the automatic collection of citations is the enrichment of the *LdoD Archive* data. Therefore, the current domain model has to be extended to support the new information and enhance the archive with the social media aware functionalities.

3.1 Domain Model

The Social Media Aware LdoD Archive supports a set of new features, namely, the collection of citations on Twitter³, their transformation into annotations of the fragments belonging to a virtual edition, the interconnection of the new Social Media Aware Virtual edition with the sources of citations, the visualization of the meta information associated with a citation in the context of the cited fragment, the customization of the Social Aware virtual editions, and the automatic regeneration of virtual editions to consider variations on its set of criteria, as which posts to consider given a period of time, and the most recent citations.

The following domain model, Fig. 1, represents how the actual *LdoD Archive* model was extended to support the new set of features. The entities and relationships added to the previous model [5] are shaded in gray.

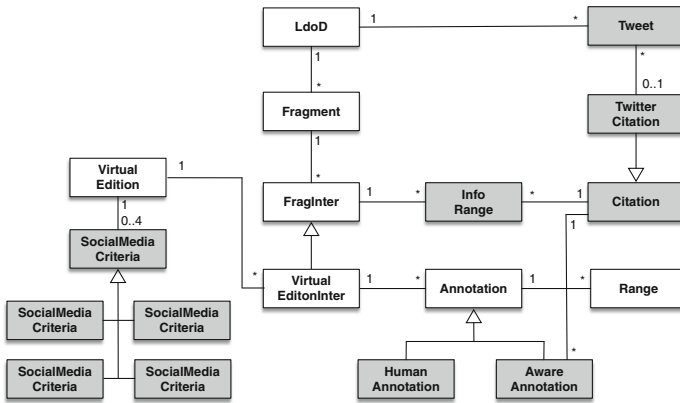


Fig. 1. LdoD archive model extension

The Citation class represents the citations from *LdoD* that were identified in social media. Currently, the *LdoD Archive* only collects citations from Twitter. Therefore, there is a single subclass of *Citation*, *TwitterCitation*. This class is associated with a *Twitter* class, which contains all the data collected with

³ <https://twitter.com>.

the tweet: source link (URL), date, tweet text, tweet ID, location, country, username, user profile URL, user image URL and the original tweet ID (in case we are dealing with a retweet instead of the original tweet). Note that some instances of **Tweet** may not be associated with a **TwitterCitation** instance, represented by 0.1 multiplicity in the diagram, which corresponds to one of the situations of false positives, as we will explain later. **Citations** are associated to a fragment and are afterwards associated to a particular transcription of a fragment, the one that is used in the virtual edition. This was achieved by extending the former **Annotation** class into two subclasses, **HumanAnnotation** and **AwareAnnotation**, where the former represents the previous annotation made manually by the users, while the latter is the automatically generated social aware annotation. Since citations correspond to parts of a fragment, for instance a sentence, the **AwareAnnotations** also apply to the part of the fragment that is cited, and the system is able to automatically identify it, class **InfoRange**. Finally, in order to enrich the virtual editions with the new set of features, a set of social media criteria, represented by the class **SocialMediaCriteria**, can be added to a virtual edition. The Social Aware Archive interprets these criteria and generates the **AwareAnnotations** that fulfil the conditions, for instance, if the **TimeWindow** of a virtual edition specifies the last two months, only citations tweeted in the last two months are considered for the creation the **AwareAnnotations** for that virtual edition. Note that this customization mechanism is dynamic, which means that whenever a criterion is changed the set of **AwareAnnotations** is automatically regenerated.

3.2 Pipeline

The overall process that collects citations from Twitter, classifies them and, finally, generates social aware annotations is presented in Fig. 2.

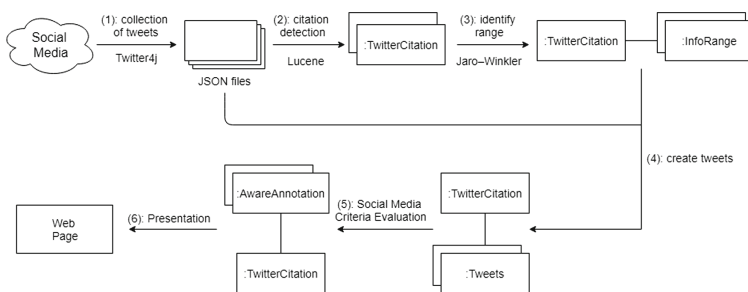


Fig. 2. Automatic collection and citation generation pipeline

The first step in the process of integrating social media citations into the *LdoD Archive* is the collection of citations from social media, which is indicated in Fig. 2 by (1). In this step, a daily query is made on social media (though

Twitter API allows the search of last week tweets), through `Twitter4J`⁴, using a set of keywords. This set can be configured but currently it searches for all the tweets that contain the words, “Livro do Desassossego”, “Fernando Pessoa”, “Bernardo Soares” and “Vicente Guedes”. During this first step, a JSON file is created, for each one of the keywords, with the information of all the tweets that contain that keyword.

In the second step (2) Twitter citations are created from the tweets previously collected. To do so, the content of the tweets in the JSON files is compared to the inverted index of fragments produced by Lucene and, depending on a threshold, a Twitter citation is created if the content is similar enough (using the Boolean Model and Vector Space Model together with normalization methods [4]).

In the third step (3) we identify the citation inside the cited fragment. To do so, the content of a tweet citation is compared to its fragment text and a match is considered when a substring of the tweet text is found. A valid substring has a predefined number of minimum words to be considered as a valid quote, e.g. 10 words. Most of the tweet texts are not exact quotes because users often write additional information in their texts, therefore it was necessary to implement an algorithm that searches for a substring of the tweet text in its fragment text. As a result, an `InfoRange` instance, containing the correct location of the citation in the text, is created for each interpretation of the fragment.

Next, in the fourth step (4), `Tweet` instances are created to all entries in the JSON files, and are associated with the respective `TwitterCitation` instances. After this step there are two types of false positives: tweets that contain one of the keywords but are below the Lucene threshold, which are represented by `Tweet` instances without an associated `TwitterCitation`; and tweets above the threshold that do not contain any substring of the fragment, which are represented as `TwitterCitation` instances but do not have any `InfoRange` associated. Both types of information are not discarded to allow the tuning of steps (2) and (3).

Whenever a user creates, or edits, a social media criterion, the fifth step (5) occurs, where all the citations associated with the fragments contained in the virtual edition are filtered according to the set of virtual edition social media criteria, and `AwareAnnotation` instances are automatically created or deleted, depending on the changes on the social media criteria. Finally, when a user selects a fragment (6) of a virtual edition that contains social aware annotations, she can visualize the cited part of the text, and the link for the citing tweet.

4 Social Media Aware Archive

The list of all generated citations is available⁵, and it is updated daily.

Another feature supported by the social aware edition is the use of citations to rearrange the editions and even build virtual editions from scratch. In what concerns the former, the edition can be ordered, for instance, in descendent order

⁴ <http://twitter4j.org>.

⁵ <https://ldod.uc.pt/citations>.

in terms of the number of citations a fragment has, and fragments can be filtered based on a geographic location/country. Regarding the latter, it is possible to define an edition whose fragments are not selected by the user, but depend on the defined social media criteria, for instance, only containing the fragments that were cited over a certain period by users in Brazil. Currently, the archive contains a public virtual edition *Citações no Twitter*⁶, which contains the set of fragments that were cited in the last 30 days, ordered by descending order of the number of citations. This virtual edition is automatically updated every day and provides an interesting feedback on which are the most cited fragments and sentences of the *LdoD*.

Besides the automatic social aware virtual edition, the *LdoD Archive* allows the access to the tweets that cite the *LdoD* (left side in Fig. 3) and annotates the cited portion in the respective virtual edition (right side in Fig. 3).

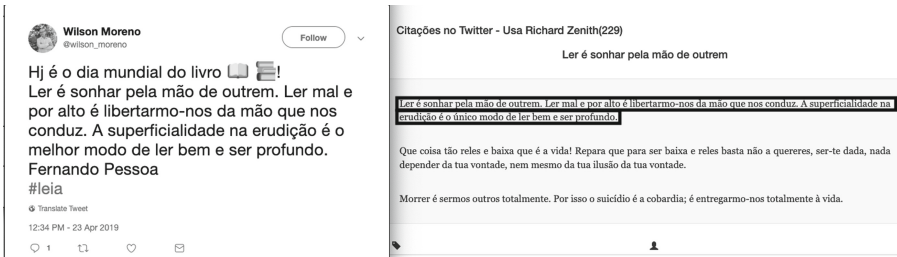


Fig. 3. Tweet and correspondent annotation in the archive

Note that the tweet contains more text than the actual citation, but it is identified due to our flexible substring algorithm.

5 Evaluation

To evaluate our solution, we present statistical results about the absolute number of citations, info ranges and annotations created, while varying the keywords used for searching on Twitter, the Lucene score threshold, and word window for the string matching algorithm. We also have analyzed the performance levels of the different steps of the pipeline using precision and recall measures.

The evaluation is done on a group of 81.833 different tweets (including retweets) that were collected during a six month period.

5.1 Results

Table 1 presents the total number of collected tweets, number of citations and info ranges created for each keyword, depending on the applied Lucene score.

⁶ <https://ldod.uc.pt/edition/acronym/LdoD-Twitter>.

Table 1. The number of tweets collected for each keyword, during a six month period, and the number of identified citations and their info ranges, depending on the applied Lucene threshold.

<i>Keywords</i>	<i>Tweets</i>	>20	>30	>40
<i>“livro do desassossego”</i>	1067	302/590	162/497	103/396
<i>“bernardo soares”</i>	2272	605/901	247/477	159/402
<i>“fernando pessoa”</i>	79.271	6069/6459	1588/3753	762/2360
<i>“vicente guedes”</i>	104	17/-	2/-	1/-
<i>Total</i>	82.714	6993/7950	1999/4727	1025/3158
<i>Total (without repeated tweets)</i>	81.833	6621/7161	1835/4272	925/2801

Note that the superior number of info ranges, in all cases, except for the “vicente guedes” keywords, is due to the fact that the info range applies to a transcription of a fragment (an interpretation) and for each fragment there can be 2 to 7 interpretations, for instance, one for the authorial source transcription, and one for each of the 4 expert editions transcriptions, in which case it corresponds to 5 info ranges, if the matching algorithm is able to create a info range for each one of them. In the case of the tweets collected with the “vicente guedes” keywords, none of the citations identified by Lucene have a match in any transcription, and so, the number of transcriptions multiply factor does not apply.

As expected, the number of citation and info ranges instances was bigger when we apply to Lucene a threshold of 20. It is the most flexible and tolerant threshold of the three, producing 3.6 times more citations and 1.7 times more info ranges than when the threshold is set to 30.

Although it is obvious that by applying a lower threshold on Lucene more citations are generated, it does not reflect the performance of the threshold. By using a lower threshold a much larger number of tweets are submitted to the info range identification step, which raises the question whether the extra citations obtained using a lower threshold are worth the computational power required to identify the info ranges. This and other performance analysis are discussed in the next subsections.

5.2 Twitter4j Performance

Table 2 shows the precision of Twitter4j module according to the keywords that were used to collect citations of the *LdoD* from Twitter. To compute these values we selected, for each one of the keywords, 100 random tweets that were collected during the six month period and determined how many were in fact citations from the book. Note that these percentages correspond to citations that originate, at least, one info range.

Observing the table, we can notice that the keyword that was more helpful and precise while searching for tweets when we used Twitter’s API was “livro

Table 2. Twitter4j precision according the keywords used for collecting tweets.

	<i>“livro do desassossego”</i>	<i>“bernardo soares”</i>	<i>“fernando pessoa”</i>	<i>“vicente guedes”</i>
<i>Precision</i>	28%	15%	7%	0%

do desassossego”, followed by “bernardo soares” and “fernando pessoa”. On the other hand, searching for the keyword “vicente guedes” did not bring any results. These results were expected since “livro do desassossego” is more specific and precise than all the others. And “bernardo soares” is more precise than “fernando pessoa” naturally because it was the heteronym Bernardo Soares who wrote the *LdoD*. Unfortunately, searching for Vicente Guedes (who was another heteronym that wrote this book) did not produce any results, which denotes how the readers, in general, assign the authorship of the *LdoD* to Pessoa’s heteronyms.

It is also interesting to compare this table with the absolute results from the previous section, in which the majority of citations, info ranges and annotations were created using the keyword followed by “bernardo soares” and “livro do desassossego”. Summing up, when we search for the keyword “livro do desassossego” we have the highest probability of finding a tweet that is a quote of the book (corresponding to 28% of probability). On the other hand, and based on the previous section, it is the keyword “fernando pessoa” that originates more annotations since it is also this keyword that collected more tweets, though it requires a much higher computational effort, because many other citations of Fernando Pessoa are tweeted, besides the ones belonging to the *LdoD*.

5.3 Lucene Performance

To measure the Lucene performance, we use the same 100 random tweets for each of the keywords, and we carefully looked into the fragments transcription, using the search user interface⁷ of the *LdoD Archive* and manually verify if the tweet’s texts contain citations from the *LdoD*.

As mentioned, the content of the tweets in the JSON files is compared to the inverted index of fragments produced by Lucene and, depending on a threshold, a Twitter citation is created if the content is similar enough. To evaluate Lucene we changed the threshold of the Lucene score in order to see the impact in terms of precision and recall. Table 3 presents the results.

In terms of precision, it is possible to conclude that it is higher when the Lucene threshold is also higher (independently of the keyword used). This happens because higher Lucene thresholds originate a smaller amount of false positives (FP) (tweets that Lucene considered as being citations but in fact they are not). For example, the keyword “livro do desassossego” originated 0 false positives for a Lucene score of 40 versus the 10 false positives originated for a score of 20. This means that to obtain a higher precision it would be recommended to use a higher threshold.

⁷ <https://ldod.uc.pt/search/simple>.

Table 3. Lucene performance for score >20, 30, and 40.

	<i>TP</i> 20/30/40	<i>FP</i> 20/30/40	<i>TN</i> 20/30/40	<i>FN</i> 20/30/40	<i>Precision</i> 20/30/40	<i>Recall</i> 20/30/40
“livro do desassossego”	17/14/11	10/3/0	62/69/72	11/14/17	0.63/0.82/1.0	0.61/0.5/0.39
“bernardo soares”	15/7/7	16/7/4	69/78/81	0/8/8	0.48/0.5/0.64	1.0/0.47/0.47
“fernando pessoa”	5/5/4	10/0/0	83/93/93	2/2/3	0.33/1.0/1.0	0.71/0.71/0.57
“vicente guedes”	0/0/0	12/2/1	55/65/66	0/0/0	0.0/0.0/0.0	-/-/-

In terms of recall, by looking at the same three tables we can observe the inverse phenomenon. The higher the threshold, the lower the recall. This happens because higher thresholds are too rigorous, therefore producing a great amount of false negatives (FN) (tweets that Lucene considered as not being citations but in fact are *LdoD* citations) and reducing drastically the number of true positives (TP). For example, by looking at the keyword “bernardo soares” it is possible to observe that, for a threshold of 20, the number of false negatives is 0 and the number of true positives is 15 versus the 8 false negatives and only 7 true positives obtained by using a Lucene threshold of 40. This means that to obtain a higher recall it would be recommended to use a lower threshold since it would preserve the majority of true positives and reduce the number of false negatives, because lower thresholds are more flexible and tolerant.

Since the *LdoD* is not so frequently cited in social media, our goal was to collect as many citations as possible, therefore lowering the Lucene threshold, even if it meant a higher computational effort in the string matching step.

5.4 String Matching Algorithm Performance

Finally, to analyse the performance of the substring matching algorithm, we tested it using different Lucene thresholds and word windows. Table 4 illustrates the exact same process that was done in the previous section, but in this case we are testing the precision and recall of our string matching algorithm for word window >10, using the same group of tweets.

The string matching algorithm receives as input only the positives detected by Lucene. For example, we receive as input 27 (15 + 0 + 10 + 2) citations created by Lucene using a threshold of 20 for citations collected with the keyword “livro do desassossego”. These 27 correspond to the 17 true positive and 10 false positive detected by Lucene with threshold >20.

In terms of precision, by analyzing Table 4 we conclude that our algorithm is 100% precise for all keywords and for all Lucene thresholds. This happened because our algorithm did not produce any false positives.

In terms of recall, it is important to remember that it depends on the number of false negatives, tweets that our algorithm considered as not being citations

Table 4. Lucene performance for score >20, 30, and 40, and word window >10.

	<i>TP</i> 20/30/40 <i>ww</i> > 10	<i>FP</i> 20/30/40 <i>ww</i> > 10	<i>TN</i> 20/30/40 <i>ww</i> > 10	<i>FN</i> 20/30/40 <i>ww</i> > 10	<i>Precision</i> 20/30/40 <i>ww</i> > 10	<i>Recall</i> 20/30/40 <i>ww</i> > 10
“ <i>livro do desassossego</i> ”	15/13/11	0/0/0	10/3/0	2/1/0	1.0/1.0/1.0	0.89/0.93/1.0
“ <i>bernardo soares</i> ”	13/7/7	0/0/0	16/7/4	2/0/0	1.0/1.0/1.0	0.87/1.0/1.0
“ <i>fernando pessoa</i> ”	4/4/4	0/0/0	10/0/0	1/1/0	1.0/1.0/1.0	0.8/0.8/1.0
“ <i>vicente guedes</i> ”	0/0/0	0/0/0	12/2/1	0/0/0	-/-/-	-/-/-

but are in fact *LdoD* citations. By observing Table 4, we can see that there are 1 or 2 false negatives per keyword. These false negatives were due to the fact that the word window of 10 was not small enough to detect quotes that were short sentences with less than 10 words. Note that we get a recall of 100% for every keyword when using a Lucene threshold >40, because there are no false negatives. This happens because with this threshold we get high levels of precision, consequently the majority of false positives are filtered before applying the string matching algorithm.

Table 5. Lucene performance for score >20, 30, and 40, and word window >5.

	<i>TP</i> 20/30/40 <i>ww</i> > 5	<i>FP</i> 20/30/40 <i>ww</i> > 5	<i>TN</i> 20/30/40 <i>ww</i> > 5	<i>FN</i> 20/30/40 <i>ww</i> > 5	<i>Precision</i> 20/30/40 <i>ww</i> > 5	<i>Recall</i> 20/30/40 <i>ww</i> > 5
“ <i>livro do desassossego</i> ”	16/13/11	3/1/0	7/2/0	1/1/0	0.84/0.93/1.0	0.94/0.92/1.0
“ <i>bernardo soares</i> ”	14/7/7	4/2/1	12/5/3	1/0/0	0.78/0.78/0.86	0.93/1.0/1.0
“ <i>fernando pessoa</i> ”	4/4/4	4/0/0	6/0/0	1/1/0	0.5/1.0/1.0	0.8/0.8/1.0
“ <i>vicente guedes</i> ”	0/0/0	0/0/0	12/2/1	0/0/0	-/-/-	-/-/-

Table 5 presents the results obtained when the word window is >5. We can observe that precision values decrease, as it was expected, since we are now more tolerant, thus originating false positives. On the other hand, the recall remained basically the same because the number of false negatives did not change, which means that the false negatives are quotes with less than 5 words.

With the results from this section we conclude that if we use a word window of at least 10 words our algorithm is 100% precise, independently of the Lucene threshold used in the previous step of our solution. In terms of recall, we obtained

better results when we combined our algorithm with a higher Lucene threshold. However, using a Lucene threshold >40 has the cost of discarding tweets that contain citations, as we have seen in the previous section.

5.5 Discussion

Table 6 shows the absolute number and percentage of citations detected by Lucene that latter originated info ranges. For example, by using a Lucene threshold >20 and a word window >10 only 23% of the citations that we collected from Twitter were in fact *LdoD* citations.

Table 6. Number and percentage of citations that originate info ranges.

	<i>Lucene</i> > 20	<i>Lucene</i> > 30	<i>Lucene</i> > 40
<i>Word Window</i> > 5	3072/6621 (46%)	1112/1835 (61%)	718/925 (78%)
<i>Word Window</i> > 10	1542/6621 (23%)	908/1835 (50%)	631/925 (69%)

When a citation originates an info range it means that the citation really exists in the fragment text, we make this assumption by the results of 100% in Table 4, when we used a word window >10 . On the other hand, we can not assume this with a word window >5 , Table 5. As described, a word window >5 is not 100% precise, consequently creating incorrect annotations sometimes. Since we want 100% precise results in what concerns the creation of annotations it would not be recommended to use a word window >5 .

To obtain a higher percentage of useful citations and to not fill the database with useless citations, it would be appropriate to use a Lucene threshold >40 . However, we are not interested in this, because by using a Lucene threshold >40 it would be too rigorous and we would be discarding useful citations increasing the amount of false negatives. If we can discard citations that are useless (by running a script that cleans citations that did not originate info ranges) we prefer a Lucene threshold >20 because it is the one that creates the greatest number of citations that originate info ranges (1542) and we also guarantee that the database is clean over time.

6 Conclusions

The Social Media Aware Virtual Editions feature is an extension of the dynamic nature of the *LdoD Archive*, and in particular of its virtual editing functionalities. By collecting online evidence about the reading of the *Book of Disquiet* through automatic citation collection, this new feature contributes to an understanding of reading as a social practice. The collection of citations by undifferentiated readers through social media will be part of a textual environment that will also include the reception of the *Book of Disquiet* by experts, including marked up

essays and reviews published during the last four decades (1982–2018). Thus the Social Media Aware Virtual Editions will speak to the *LdoD Archive*'s rationale of creating a dynamic textual environment which explores digital humanities methods for creating multiple perspectives on the literary operations of reading, writing, and editing.

This paper presents how this social aware functionalities were designed and implemented, and offers a detailed performance analysis of the automatic collection and citation generation pipeline, which allowed us to fine tune its parameters. On the other hand, this solution is easily extensible to other social media sources, like Facebook posts and the web pages resulting from queries done using a search engine.


Acknowledgement. This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UID/CEC/50021/2019.

References

1. Barata, C.: Impacto de obras literárias nas redes sociais - the impact of literary works in social networks. Master's thesis, Faculty of Sciences, University of Lisbon, Portugal (2014)
2. Boyer, R.S., Moore, J.S.: A fast string searching algorithm. *Commun. ACM* **20**(10), 762–772 (1977)
3. Eysenbach, G.: Can tweets predict citations? Metrics of social impact based on Twitter and correlation with traditional metrics of scientific impact. *J. Med. Internet Res.* **13**(4), e123 (2011)
4. Liu, B.: *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*. Springer, Heidelberg (2011). <https://doi.org/10.1007/978-3-642-19460-3>
5. Portela, M., Silva, A.R.: A model for a virtual LdoD. *Digit. Sch. Hum.* **30**(3), 354–370 (2014)
6. Silva, A.R., Portela, M.: TEI4LdoD: textual encoding and social editing in web 2.0 environment. *J. Text Encoding Initiat.* **8** (2014)
7. Thelwall, M., Tsou, A., Weingart, S., Holmberg, K., Haustein, S.: Can tweets predict citations? Metrics of social impact based on Twitter and correlation with traditional metrics of scientific impact. *J. Med. Internet Res.* **13**(4), e123 (2011)
8. Weller, K., Puschmann, C.: Twitter for scientific communication: how can citations/references be identified and measured? In: 2011 ACM WebSci Conference, WebSci 2011, pp. 1–4, June 2011
9. Winkler, W.E.: The state of record linkage and current research problems. Technical report, Statistical Research Division, U.S. Bureau of the Census (1999)



Language Identification for South African Bantu Languages Using Rank Order Statistics

Meluleki Dube and Hussein Suleman^(✉) 

University of Cape Town, Cape Town, South Africa
DBXMEL004@myuct.ac.za, hussein@cs.uct.ac.za
<http://dl.cs.uct.ac.za/>

Abstract. Language identification is an important pre-process in many data management and information retrieval and transformation systems. However, Bantu languages are known to be difficult to identify because of lack of data and language similarity. This paper investigates the performance of n-gram counting using rank orders in order to discriminate among the different Bantu languages spoken in South Africa, using varying test and training data sizes. The highest average accuracy obtained was 99.3% with a testing size of 495 characters and training size of 600000 characters. The lowest average accuracy obtained was 78.72% when the testing size was 15 characters and learning size was 200000 characters.

Keywords: N-grams · Bantu languages · Rank order statistics

1 Introduction

Language identification is the process of automatically determining the natural language of any electronic text [8]. It is often the precursor to language-dependant processing such as machine translation, search algorithms, text-to-speech, etc. [6]. It is currently a hard task to identify text written in Bantu languages due to the lack of support for these languages on the Web. This, in turn, has led to the Bantu languages being excluded from services like translation services and voice synthesis services and also means that processing of these languages is made harder [3].

The South African Bantu languages used in this paper include: isiZulu, isiNdebele, isiXhosa, siSwati, Setswana, Sesotho, Pedi, Tsonga and Venda; these are all national languages of South Africa, excluding the 2 languages of European origin (English and Afrikaans). The first four languages (isiZulu, isiNdebele, isiXhosa and siSwati) belong to the same family, which is the Nguni languages [1] that share much vocabulary. The goal of this paper is to explore the degree to which texts in these Bantu languages can be differentiated using algorithmic language identification.

There are currently different techniques in use to identify languages. These include: Naïve Bayesian, normalized dot-product, centroid-based, and relative

entropy and n-gram counting using rank order statistics [1]. In this paper, rank order statistics, as described by Cavnar and Trenkle [2], are applied to South African Bantu languages and evaluated. Three key aspects are investigated: the effect of training data size; the effect of test data size; and the impact of language similarity.

2 Literature Review

2.1 N-Gram Count Using Rank Orders

An n-gram is a sequence of n consecutive characters from some text, with the value of n often being 1, 2, or 3 [9]. N-grams are usually called unigrams when $n = 1$, bigrams when $n = 2$ and trigrams when $n = 3$ e.g. trigrams for the Ndebele word, *'bhala'* (translation: *'write'*) are *bha*, *hal*, *ala*.

Cavnar and Trenkle [2] devised an approach for text categorisation that is tolerant of textual errors, achieving 99.8% accuracy (correct classification rate) in one of their tests. Their system also achieved 80% accuracy in classifying articles from a different computer-oriented newsgroup. This approach is adopted in this research to investigate whether it is possible to achieve high accuracy ratings for classifying Bantu languages. The system entails calculation and comparisons of profiles of n-gram frequencies, where a profile of n-gram frequencies refers to the n-gram frequencies ranked in descending order [2]. The reason this is possible is due to Zipf's principle of least effort, which states that the frequency of occurrence of a word is almost equal to the inverse of its rank [7]. This then suggests that 2 separate texts in a particular language, say Ndebele, will have almost the same n-gram distribution. Thereby if we can compare 2 profiles and determine that they are not far apart by some measure, there is then a high probability that the 2 profiles belong to the same classification group. The exact method of computing the distances is described within the methodology section in this study.

2.2 Related Work on Bantu Languages

Zulu, Botha, and Barnard [11] investigated if it is possible to quantify the extent to which the South African languages differ, given that the 9 different languages cluster into families. For example, isiNdebele, isiXhosa, siSwati and isiZulu are all in the Nguni sub-family. Measurements for similarity were made and, for example, isiNdebele and isiZulu had a distance of 232 between them, isiNdebele and isiXhosa had a distance of 279 between them and isiNdebele and siSwati had a distance of 257 between them. In contrast, English and isiNdebele had a distance of 437, while English and isiZulu had a distance of 444. It can be seen that there is a strong affinity within the group and a marked difference from English, which is not Nguni.

Combrinck and Botha [4] presented a statistical approach to text-based automatic language identification, which focused on discrimination of language models as opposed to the representation used. They used a subset of South African

Bantu languages (isiZulu, isiXhosa, Tswana, Sepedi), as well as Swazi. Botha and Barnard [1] looked at the factors that affected language identification with a focus on all South African languages, including English and Afrikaans. They included multiple algorithms, but concluded that SVM and Naïve Bayes performed the best, with an accuracy of up to 99.4% for a test data size of 100 characters. In contrast, this paper considers only the Bantu languages and investigates a rank-order algorithm that is arguably simpler and faster to update/train.

3 Methodology

The methodology employed in this paper is adapted from the work by Cavnar and Trenkle [2]. Given that their system achieved a maximum accuracy of 99.8% for classifying languages, the goal is to find out if a combination of parameters can result in a comparable accuracy.

3.1 Corpora

Text corpora for the 9 South African Bantu languages were acquired from the South African Centre for Digital Language Resources. The files were first manually cleaned to remove metadata.

A maximum training size of 600000 was decided on to test how well these algorithms work for languages with lesser amounts of data. The corpus for each language was sampled to create these subsets, by extracting 2000-character segments at periodic intervals from within the full corpora to ensure maximum variability in the data.

3.2 Language Detection Algorithm

Given a set of corpora for different languages, a model is created for each language in the training dataset. Each such model is a vector of the most highly-occurring trigrams, sorted in order of frequency, for the language. In order to detect the language of some unknown test data, a similar list of frequent trigrams is calculated. The test data trigram vector is then compared against each of the language models and the most similar is chosen as the language of the test data.

This calculation for distance/similarity between 2 language models is presented in Fig. 1 as an example. The ranks of each pair of corresponding n-grams is subtracted, and the sum of these differences is then the similarity metric. There is a maximum of 300 n-grams in each model and this maximum value is used where one list contains an n-gram but the other does not.

3.3 Experimental Design

In evaluating the system, 10-fold cross validation was employed.

One language dataset was divided into 10 sections/folds. 9 folds were used as training data and 1 fold was held back and used for testing. Testing was performed by dividing the test fold into small chunks and determining the language

Trigrams for the testing data that is in isiNdebele arranged in the order of their frequencies (highest to lowest)	Trigrams for the training data (model for isiNdebele language) arranged in the order of their frequencies (highest to lowest)	Out of order number for the model and the testing data given by the absolute value of the difference between rank in mode- rank in testing data
nga	nga	$ 0-0 =0$
la	oku	$ 1-2 =1$
oku	la	$ 2-1 =1$
a n	ela	Max
ana	ana	$ 4-4 =0$
enz	nam	$ 5-6 =1$
ela	enz	$ 6-3 =3$
		$\therefore \text{distance} = 0 + 1 + 1 + \text{Max} + 0$ $\quad \quad \quad + 1 + 3$ $\quad \quad \quad = 6 + \text{Max}$

Fig. 1. Example of similarity metric calculation

of each chunk. This was repeated 10 times, using different testing folds each time. During this process, the other language models were held constant based on the complete dataset for each. The final predicted language accuracy is the average over all the languages predicted in each of the 10 tests. This process was then repeated for each language.

The major parameters in this experiment were the different training data sizes and test chunk sizes. In this experiment, the test chunk size was varied from 15 characters to 510 characters and the training data size was varied from 100000 characters to 600000 characters.

4 Results and Analysis

The maximum achieved accuracy was 99.3%. This was when the language model size was 600000 characters and the testing chunk size was 450 characters. The minimum accuracy achieved, on the other hand, was 78.72%, which was obtained when the training data size was 200000 and the testing chunk size was 15 characters.

The results, to a certain extent, adhere to the hypothesis. Figure 2 shows that both increasing the testing data size and the data size for creating the models will effectively increase the accuracy of the system. The increase in accuracy due to increase in training data size is seen in the shift to the right of the different graphs. The system never reaches a 100% accuracy and the 100% accuracy line acts as an asymptote of the graph. However, after a certain point in time, increasing the testing data size becomes less useful, as there is no increase in the overall accuracy of the system. Therefore, the combination that yields the optimum results will be taken as the combination that yields the maximum accuracy.

Consider the matrix in Fig. 3, which shows the number of times the system mistakenly recalled one language as the other, for when the model size was 100000 and testing chunk size was 15.

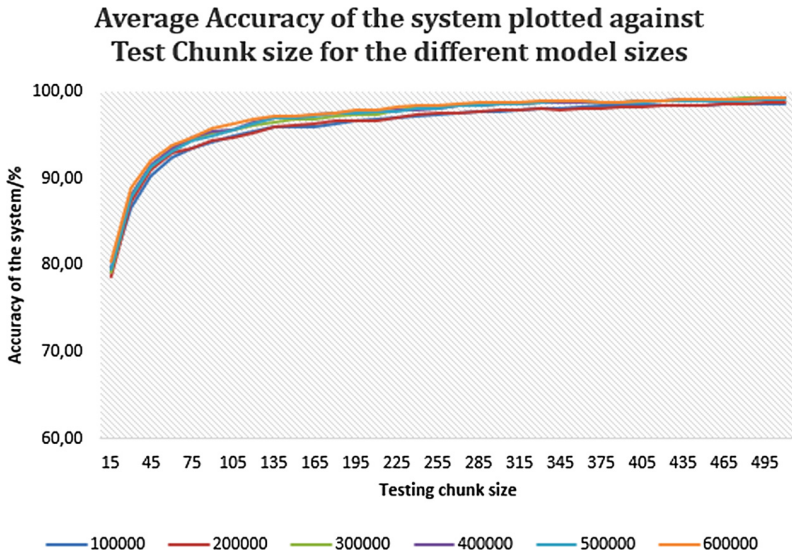


Fig. 2. Average accuracy of the system plotted against the testing chunk size for the different training data sizes

	Ndebele	Pedi	Sotho	Tswana	Swati	Tsonga	Venda	Xhosa	Zulu
Ndebele	519	1	1	3	16	10	5	98	247
Pedi	3	786	22	80	4	2	1	2	0
Sotho	9	7	783	90	5	0	2	2	2
Tswana	0	51	100	737	1	5	2	0	4
Swati	40	3	4	2	788	6	3	27	27
Tsonga	11	2	2	9	8	854	11	0	3
Venda	4	1	2	1	2	10	873	3	4
Xhosa	84	1	5	1	41	6	5	519	238
Zulu	105	1	2	1	63	2	3	156	567

Fig. 3. Predicted languages when different languages were used as testing model. The heading/first row indicates the predicted languages and the heading/first column indicates the languages used for testing. Testing data size used was 15 characters and training data size used was 100000 characters.

As seen from Fig. 3, the reason for the low average accuracy for the system is the failure to properly identify some of the Nguni languages, in particular isiNdebele. The system managed to correctly identify isiNdebele 57.67% of the time within the 900 trials. Other languages achieved 81.89% and higher accuracies for the same testing chunk size and model size. This behaviour also persisted for larger training data sizes.

When using training data in isiNdebele, this effect is more clearly seen. As already stated, the low accuracies are mostly due to failure to correctly classify the Nguni languages, which share considerable vocabulary so are difficult to tell apart.

This effect was also reported by Zulu, Botha and Barnard [11] when using smaller model sizes and minimal test chunk sizes.

5 Conclusions and Future Works

In this paper, we set out to investigate if the method of counting n-grams using rank orders can be used to determine the language of text in a Bantu language. The similarity metric proposed by Cavnar and Trenkle [2] was adopted.

A key goal was to determine combinations of testing chunk size and training data size that would give the optimum accuracy values. A 2-factor experiment was conducted, with n-fold cross-validation to determine the average system accuracy and the accuracy within each parameter combination. The results indicate that it is indeed possible to use rank orders to perform language identification among the South African Bantu languages, provided that the model sizes are sufficiently large to ensure a relatively high accuracy across all languages.

As expected, the accuracy for languages within a family of related languages was lower (e.g., in the case of isiNdebele) when smaller models were used for training and smaller test chunks were used. This is not necessarily problematic, as similar languages will share morphological analysis, translation and other tools, so the exact language may not impact on the eventual use case.

These results confirm prior results on different subsets of languages and using different techniques. But, in particular, they confirm that the rank order technique used can be applied to a regionally-defined set of related and unrelated low resource languages, as can be found in many parts of the world. This work was initially motivated by the creation of low resource language archives and multilingual search across low-resource languages. The results are promising in that they show even very short social media posts in low resource languages can be identified using these techniques.

Acknowledgements. This research was partially funded by the National Research Foundation of South Africa (Grant numbers: 85470 and 105862) and University of Cape Town. The authors acknowledge that opinions, findings and conclusions or recommendations expressed in this publication are that of the authors, and that the NRF accepts no liability whatsoever in this regard.

References

1. Botha, G.R., Barnard, E.: Factors that affect the accuracy of text-based language identification. *Comput. Speech Lang.* **26**(5), 307–320 (2012)
2. Cavnar, W.B., Trenkle, J.M., et al.: N-gram-based text categorization. In: *Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, vol. 161175. Citeseer (1994)

3. Chavula, C., Suleman, H.: Assessing the impact of vocabulary similarity on multi-lingual information retrieval for Bantu languages. In: Proceedings of the 8th Annual Meeting of the Forum on Information Retrieval Evaluation, pp. 16–23. ACM (2016)
4. Combrinck, H.P., Botha, E.: Text-based automatic language identification. In: Proceedings of the 6th Annual Symposium of the Pattern Recognition Association of South Africa (1995)
5. Dunning, T.: *Statistical Identification of Language*. Las Cruces, Computing Research Laboratory (1994)
6. Duvenhage, B., Ntini, M., Ramonyai, P.: Improved text language identification for the South African languages. In: 2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech), pp. 214–218. IEEE (2017)
7. Li, W.: Random texts exhibit zipf’s-law-like word frequency distribution. *IEEE Trans. Inf. Theory* **38**(6), 1842–1845 (1992)
8. McNamee, P.: Language identification: a solved problem suitable for undergraduate instruction. *J. Comput. Sci. Coll.* **20**(3), 94–101 (2005)
9. Ndaba, B., Suleman, H., Keet, C.M., Khumalo, L.: The effects of a corpus on isizulu spellcheckers based on n-grams. In: 2016 IST-Africa Week Conference, pp. 1–10. IEEE (2016)
10. Poole, D., Mackworth, A.: *Artificial intelligence foundations of computational agents*. 2010 (2017)
11. Zulu, P., Botha, G., Barnard, E.: Orthographic measures of language distances between the official South African languages. *Literator: J. Lit. Crit. Comp. Linguist. Lit. Stud.* **29**(1), 185–204 (2008)

Posters



Creating a Security Methods Taxonomy for Information Resource Management

Yejun Wu¹ and Fansong Meng²

¹ School of Library and Information Science, Louisiana State University,
Baton Rouge, LA 70803, USA
wuyj@lsu.edu

² College of Information and Communication, National University of Defense
Technology, Wuhan, Hubei, China
mengfansong17@nudt.edu.cn

Abstract. People who are concerned with security are naturally interested in methods of achieving security. This paper proposes an approach to creating a security methods taxonomy which can be used for security management and the management of security-related information resources when security methods are the focus. The framework of the taxonomy is based on facet analysis and a tetra-facet model of security, which identifies four facets of security: subject/scope of security, object of protection, source of insecurity, and method of protection. A taxonomy of security methods can be created by combining two of the facets: source of insecurity and method of protection.

Keywords: Security methods · Taxonomy · Information resource management

1 Introduction

On May 16, 2019, the U.S. Department of Commerce applied export control restrictions to China telecommunications equipment maker Huawei Technology Ltd. and 68 of its affiliates deemed a risk to U. S. national security [1]. This is an example of using political and legal methods to achieve national security, and raises a question: what methods can be used to achieve security? Security in this paper is a concept related to threat, safety, and protection. People who are concerned with security are naturally interested in methods of protecting security (or security methods). The goal of this research is to develop a taxonomy of security methods. The research questions of this study are:

- How do existing classification systems classify security methods?
- How can we categorize security methods for facilitating security management and the management of security-related information resources?

2 Novelty and Significance of the Study

Security is an “ambiguous term” and there is not “a precise and universally accepted definition of security” [2]. Various aspects of security are studied in various disciplines and application domains, and the community of security studies has greatly expanded the scope of security in the past thirty years. As a result, there is no comprehensive classification scheme of security. Wu and Meng introduced an approach to developing a comprehensive taxonomy of security, and proposed to develop a taxonomy of security methods in a future study [3]. This research is new in the sense that there is no taxonomy of security methods and no research on the development of such a taxonomy, and it applies facet analysis to the hypernym of a subject term (i.e., security) instead of the subject term itself (i.e., security methods) in taxonomy development. The taxonomy can be useful for managing information resources about security when security methods are focused, and can also be useful for security management since multiple organizations may use certain security methods to manage multiple aspects of security.

3 Methodology

In our previous work [3], we collected 120 journal articles, book chapters, and online documents that are closely related to security definition, dimension, feature, taxonomy and classification from 102 security-related journals and websites. We have collected existing classification snippets of security from knowledge organization systems such as Library of Congress Classification (LCC), Library of Congress Subject Headings (LCSH), and ACM Computing Classification System (ACMCCS), or inferred classification snippets of security from functional components of major government agencies and international organizations with security management tasks (such as the Department of Homeland Security, the Social Security Administration, and the World Bank [4]), and other sources [5]. These resources are analyzed to create a list of terms related to security methods. Facet analysis and a tetra-facet model of security, which was created in our previous study, will be used to build the taxonomy of security methods.

4 Findings

There is no comprehensive taxonomy of security methods except in the information security sub-domain. LCC and LCSH contain fragmented taxonomies of security [3], but does not contain any section of security methods. ACMCCS has a “security and privacy” section, which has the following major sub-categories: cryptography, formal methods and theory of security, security services, intrusion/anomaly detection and malware mitigation, security in hardware, systems security, network security, database and storage security, software and application security, human and societal aspects of security and privacy [4]. It includes some information security-related subject terms (such as system security, network security) and some security methods (such as cryptography, formal methods, and security services, which further includes

authentication, access control, and authorization) [4]. Chakrabarti and Manimaran proposed a taxonomy of Internet infrastructure security attacks and solutions [5], which can be merged into ACMCCS. In our previous work [3], we identified “method of protection” as a facet of security, and developed an incomplete, generic list of these methods, including physical method, economic method, financial method, technological method, military method, cultural/psychological method, legal/political/social method, and comprehensive method. These findings and resources lay the foundation for our approach to developing a taxonomy of security methods, which is addressed in the next section.

5 Developing a Taxonomy of Security Methods

Inspired by the security methods in the information security domain, we can work with the experts in various dimensions of security (such as national security, economic security, human security, health security, food security, personal security, international security, military security, environmental security) to develop a security methods taxonomy for each domain. However, these security dimensions have overlaps [3]. So this approach is not flawless. Ranganathan’s facet analysis can be used to separate the elements of complex subjects in relation to a set of abstract fundamental concepts [6], and the elements are called facets, which are “homogeneous or semantically cohesive categories” [7]. However, performing facet analysis on “security methods” will get the generic list of security methods (such as physical method, financial method) introduced in Sect. 4. Therefore we retreated to the facets of “security” for a possible solution.

In our previous work, we found that security is expressed in this pattern: “a subject wants to protect objects against sources of insecurity using certain methods”, and identified four facets of security – “subject/scope of security, object of protection, source of insecurity, and method of security” [3]. Using this tetra-facet model of security to analyze the security methods in the information security domain, we find that information security methods can be the combination of the “subject/scope” facet of security (which is “information”) and the “technological method” facet of “method of protection,” or the combination of the “object of protection” facet of security (which is “information” or “information systems”) and the “technological method” facet of “method of protection.” Using the “subject/scope” facet of security to develop the taxonomy of security methods is not ideal because the subjects of security can be overlapped, as addressed above. Using the “object of protection” (such as body, property, territory, governance, economy, environment, food/water, job, health, energy, information) to organize security methods is a possible solution, but we would need to make a long list of the objects we have and want to protect, and they can also be overlapped. A better solution is to use the “source of insecurity” (i.e., various kinds of threats, fears, vulnerabilities, and risks) to organize security methods because the threats and risks emerging from the “object of protection” are the things that trigger people’s security alarms.

Source of insecurity includes threats, fears, vulnerabilities, and risks, such as war and violence between/within states, weapons proliferation, migration, hunger, infectious disease, environmental degradation, hazards, climate change, poverty,

underdevelopment, social inequalities, man-made risks due to modernization (such as financial crisis), and socially-created disasters (such as destruction of water and electricity systems) [3]. Sources of micro-level personal and community insecurity, as well as emotional (vs. physiological) insecurity, can be added in the future. Table 1 shows a snippet of the framework of the taxonomy, which is created by combining the “source of insecurity” facet and the “method of protection” facet.

Table 1. A snippet of security methods taxonomy

Source of insecurity	Method of protection	Security method instance
Weapon proliferation	Technological method	Control of defense technology
	Legal method	Examination of weapon trade
	Political method	International arms embargo treaties
Climate change	Technological method	New energy technology
	Economic method	Carbon trade
	Political method	International gas reduction treaties

6 Conclusion, Implications and Future Work

There is no comprehensive taxonomy of security methods except that the ACM CCS has a section of “Security and Privacy,” which includes some security methods. Based on our tetra-facet model of security and facet analysis, we proposed a framework for the development of the taxonomy of security methods using two facets of security, which are “source of insecurity” and “method of protection,” whereas the “source of insecurity” is the leading facet. The paper demonstrates the application of facet analysis to the hypernym concept of a subject term (i.e., security) instead of the subject term itself (i.e., security methods) in taxonomy development. The taxonomy of security methods can be used for security management and the management of security-related information resources when security methods are the focus. In the future, we plan to develop a full list of instances of the two facets so that a full taxonomy of security methods can be developed, and also invite experts to evaluate the taxonomy.

References

1. Kang, C., Sanger, D.: NYC: Huawei Is a Target as Trump Moves to Ban Foreign Telecom Gear. The New York Times. <https://www.nytimes.com/2019/05/15/business/huawei-ban-trump.html>. Accessed 15 May 2019
2. Horrigan, B., et al., Security studies. In: Kurtz, L. (ed.) Encyclopedia of Violence, Peace and Conflict. Elsevier, 2nd edn. (2008)
3. Wu, Y., Meng, F.: Categorizing security from security management and information resource management. *J. Strateg. Secur.* **11**(4), 72–84 (2019)
4. Library of Congress Classification Outline, <https://www.loc.gov/catdir/cpsolcco/>; Library of Congress Subject Headings, <https://www.loc.gov/aba/publications/FreeLCSH/freelcsh.html>; The 2012 ACM Computing Classification System – Introduction, <https://www.acm.org/>

[publications/class-2012](#); The U.S. Social Security Administration, <https://www.ssa.gov/site/menu/en/>; The World Bank, <http://www.worldbank.org/>; The Department of Homeland Security, <https://www.dhs.gov/topics>

5. Chakrabarti, A., Manimaran, G.: Internet infrastructure security: a taxonomy. *IEEE Netw.* **16** (6), 13–21 (2002)
6. Douglas, J.F.: Facet analysis. In: Miriam, A.D. (ed.) *Encyclopedia of Library and Information Science*. Marcel Dekker Inc., New York (2003)
7. Elaine, S.: *The Intellectual Foundation of Information Organization*, p. 139. MIT Press, Cambridge (2000)



A Dynamic Microtask Approach to Collecting and Organizing Citizens' Opinions

Masaki Matsubara¹(✉), Yuhei Matsuda², Ryohei Kuzumi³,
Masanori Koizumi¹, and Atsuyuki Morishima¹

¹ Faculty of Library, Information and Media Science, University of Tsukuba,
1-2 Kasuga, Tsukuba-shi, Ibaraki 305-8550, Japan
{masaki,koizumi,mori}@slis.tsukuba.ac.jp

² Graduate school of Library and Media Studies, University of Tsukuba, 1-2 Kasuga,
Tsukuba-shi, Ibaraki 305-8550, Japan
yuhei.matsuda.2017b@mlab.info

³ Public Relations Policy Division, Tsukuba city hall, 1-1-1 Kenkyu-Gakuen,
Tsukuba-shi, Ibaraki 305-8555, Japan
pln032@city.tsukuba.lg.jp

Abstract. Citizens' opinions are important information resources for democratic local governments. Since a mere collection of opinions is not easy to analyze, the collected opinions should be organized, so that the governments can effectively analyze it. This paper proposes a scheme where citizens take part in organizing and classifying opinions while answering the questionnaire. In the scheme, we collect citizen opinions in a structured form, with a microtask interface that changes the list of choices dynamically. Our system has been being used by Tsukuba city for several real-world opinion-collection projects.

Keywords: Wisdom of crowds · Civic-Tech · Public opinion data

1 Introduction

Citizens' opinions are important information resources for democratic local governments [2]. Since a mere collection of opinions is not easy to analyze, the collected opinions should be organized, so that the governments can effectively use it.

Recently, local governments have started to use information technologies for collecting public opinions [1, 4, 5]. A common feature of these attempts is that IT is used as a tool for *implementing the traditional methods*, i.e., polls and surveys collecting opinions [3]. However, a questionnaire with a fixed number of choices has the following two points: (1) we cannot enumerate a perfect set of choices in advance, and (2) there is a need for citizens to express their opinions in free forms. Tsukuba city often receives complaints from citizens every time they conduct

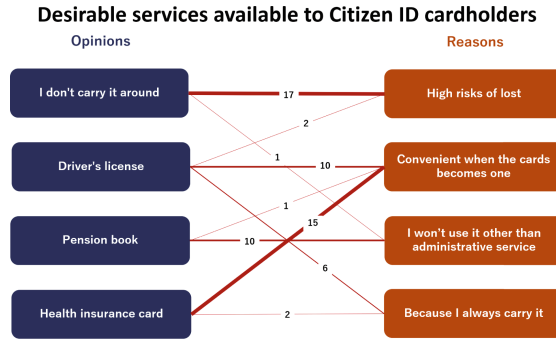


Fig. 1. Example of an output of our proposed scheme. The output is given in a bipartite graph where two sets of nodes are opinions and reasons. The edges connect opinions and reasons, with the label representing the number of votes.

surveys only with a fixed set of choices - they want to express their opinions in a free-text form. On the other hand, collecting free-form opinions requires the government staff to organize the collected opinions, which is a cumbersome task for the government staff.

This paper introduces a scheme where citizens take part in organizing and classifying opinions while answering the questionnaire, and reports some of the real-world projects in Tsukuba-city. In the scheme, we collect citizens' opinions in a structured way, with a microtask interface that changes dynamically the list of choices; if it receives a new opinion, it will be shown in the list of the choices in the task for future workers. The task also asks citizens what the reasons are for their opinions. The scheme outputs the collected opinions in a bipartite graph where two types of nodes are opinions and reasons behind them (Fig. 1). Therefore, the result is already organized in a structured form *at the moment it is obtained*, which makes it easier to analyze the collected opinions; we use the power of citizens to help us organize the collected opinions for the analysis.

2 Proposed Method

Figure 2 shows our proposed framework of the citizen opinion collection. It has two features. First, the set of choices grows; each task not only shows a set of choices to vote for, but also has a free-text form to which he or she can input his or her opinion, which appears in the list of choices in future tasks. Therefore, other citizens can choose it in performing the task. If a citizen vote for an opinion from the listed choices, we count up the number of votes for it.

Second, the opinions are given in a structured form where the opinions and reasons for them are separated; After choosing his or her opinion, the citizen then moves to another step where he can enter the reason for it. What he does in the step is the same as that in the first step - he can choose one from a given set of reasons or he can write his own reason.

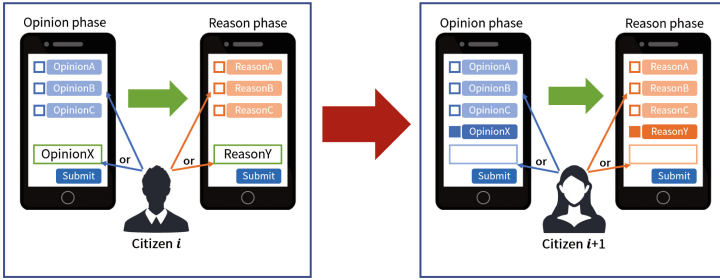


Fig. 2. Interface of the proposed framework. In the opinion phase, citizens can select the listed opinions or input new one into the free-text form. In the reason phase, citizens can select or input their reason in the same manner. When citizen i submits new item with the free-text form, the inputted item will be listed when citizen $i + 1$ performs.

As a result, the scheme outputs the collected opinions in a bipartite graph where two types of nodes are opinions and reasons behind them (Fig. 1).

3 Real-World Deployments

We tried several real-world deployments, including (1) asking “re-development of the city’s central area” at a special workshop event, (2) asking “reason of moving to Tsukuba city” by putting posters with QR code on the chairs in the waiting space for counter services of Tsukuba City Hall (Fig. 3), and (3) asking “desirable services available to Citizen ID cardholders” by using crowdsourcing platforms.

In our experience, we found that attracting people by the poster is not easy and using crowdsourcing services or special events attract more people. Note that crowdsourcing services are not appropriate for some topics for which opinions by local citizens is especially important. How to attract people is beyond the scope of the paper, and we omit further discussions.

Table 1. Result of real-world deployments

Places	Targets	# Responses	# Opinions	# Reasons
City’s central area	Workshop visitor	32 (1 week)	22	22
City Hall	Who moved to Tsukuba	77 (87 days)	6	9
Crowdsourcing	Who lives in Ibaraki	161 (2 weeks)	16	14

As a result, we got 32 responses in a week, 77 responses in 87 days, and 161 responses in 2 weeks, by a local workshop, posters at City Hall, and crowdsourcing, respectively (Table 1).



Fig. 3. Posters put at the back of chairs in the lobby of the City Hall. Each poster shows a QR code to access our system.

The ratio between the number of opinions and responses shows that our method effectively organizes the collected opinions. For some projects, we compared the output with the same set of opinions organized by the government staff and found that they are similar to each other.

4 Conclusion

This paper introduced a scheme that collects citizens’ opinions in a structured way, with a microtask interface that changes the list of choices dynamically. Our scheme “citizen-source” organizing opinions being collected on the fly while they answer the questionnaire. Our system has been being used by Tsukuba City in real-world opinion-collection projects. The future work includes the exploration of effective usages of the output bipartite graphs and the analysis of the quality of citizens’ organization of collected opinions.

References

1. Chua, A.Y., Goh, D.H., Ang, R.P.: Web 2.0 applications in government web sites: Prevalence, use and correlations with perceived web site quality. *Online Inf. Rev.* **36**(2), 175–195 (2012). <https://doi.org/10.1108/14684521211229020>
2. Fishkin, J.: *When the People Speak: Deliberative Democracy and Public Consultation*. Oxford University Press, Oxford (2011). <https://books.google.co.jp/books?id=iNsUDAAAQBAJ>
3. Price, V.: Public opinion research in the new century reflections of a former poq editor. *Public Opinion Q.* **75**(5), 846–853 (2011). <https://doi.org/10.1093/poq/nfr055>

4. Sandoval-Almazan, R., Gil-Garcia, J.R.: Assessing local e-government: an initial exploration of the case of Mexico. In: Proceedings of the 4th International Conference on Theory and Practice of Electronic Governance. ICEGOV 2010, pp. 61–65. ACM, New York (2010). <https://doi.org/10.1145/1930321.1930335>
5. Vargas, A.M.P.: A proposal of digital government for Colombia. In: Proceedings of the 11th International Conference on Theory and Practice of Electronic Governance. ICEGOV 2018, pp. 693–695. ACM, New York (2018). <https://doi.org/10.1145/3209415.3209476>



Sentiment Analysis of Tweets Mentioning Research Articles in Medicine and Psychiatry Disciplines

Sampathkumar Kuppan Bharathwaj, Jin-Cheon Na^(✉),
Babu Sangeetha, and Eswaran Sarathkumar

Wee Kim Wee School of Communication and Information,
Nanyang Technological University, 31 Nanyang Link,
Singapore 637718, Singapore

{bharathw001, sangeeth006, sarathku002}@e.ntu.edu.sg,
tjcna@ntu.edu.sg

Abstract. Recently altmetrics (short for alternative metrics) are gaining popularity among researchers to identify the impact of scholarly publications among the general public. Although altmetrics have been widely used nowadays, there has been a limited number of studies analyzing users' sentiments towards these scholarly publications on social media platforms. In this paper, we analyzed and compared user sentiments (positive, negative and neutral) towards scholarly publications in Medicine and Psychiatry domains by analyzing user-generated content (tweets) on Twitter. We explored various machine learning algorithms, and constructed the best model with Support Vector Machine (SVM) which gave an accuracy of 91.6%.

Keywords: Altmetrics · Twitter · Medicine · Psychiatry · Sentiment analysis · Text classification

1 Introduction

In digital libraries, it is important to measure and present the impact of articles for helping users to find useful articles. As a measurement, citation count has been a strong indicator upon which to draw conclusions about research impact. It is also the mechanism behind other major research measures such as journal impact factor and h-index. However, citation count and those traditional metrics are increasing failing to keep pace with the new ways that researchers can generate impact in today's digital world [1]. Among the attempts to develop non-citation-based metrics, altmetrics (short for alternative metrics) receive considerable attention due to the ease with which data can be collected and the availability of a wide range of open data sources. Altmetrics intend to assess Web-driven scholarly interactions, such as how often research is tweeted, blogged about, downloaded, or bookmarked. Thus, to evaluate the significance of scholarly works better, we should pay more attention to informal discussions on social media. Hence, we propose to analyze the sentiment of the users towards scholarly publications on a social media platform, Twitter. Analyzing the sentiment of

tweets mentioning research articles and classifying them as positive, negative or neutral would help us understand the accurate meaning of Twitter metric, i.e. the number of mentions of an article on Twitter.

2 Related Work

Arredondo [2] conducted sentiment analysis of social media posts discussing articles on Twitter, Facebook, Google+, and Blogs using the NLTK tool, and reported that among 58,529 social media posts, 26.28% were positive, 57.56% were neutral, and 16.16% were negative. The majority of posts mentioning research articles were neutral and carried no opinion or subjectivity at all. Na [3] manually analyzed 2,016 tweets citing academic papers in the field of psychology, and reported that the majority of the tweets (84.77%) hold neutral attitude towards the papers they cited, and the second most common category was positive (10.07%). Raamkumar et al. [4] conducted the sentiment analysis of tweets mentioning research articles in the Computer Science domain using a sentiment analysis tool, TextBlob. Of 49,849 tweets for 12,967 associated papers, 97.16% of the tweets had neutral sentiments, and positive sentiments were found in 2.8% of the tweets while negative sentiments in 0.05% of the tweets.

In the context of altmetrics, previous studies analyzed the tweets mainly using existing general-purpose sentiment analysis tools. Therefore, in this study, using machine learning approach, we aim to develop an effective sentiment classifier for tweets mentioning research articles, and the classifier is applied to investigate the sentiment polarity distribution of the tweets mentioning journal articles in Medicine and Psychiatry disciplines.

3 Methodology

Top 20 journals' articles indexed by the Science Citation Index (SCI) and published between 2015 and 2017 were selected from the Medicine domain. We also took top 20 journals' articles indexed by the Social Science Citation Index (SSCI) in the Psychiatry domain for the year 2016. The top 20 journals were identified using high Journal Impact Factors (JIFs). Altmetric.com keeps track of how people are dealing with research articles online through various social media platforms like Twitter and Facebook. We used Altmetric API to get the Altmetric ID of each article of the selected journals using its Digital Object Identifier (DOI). The Altmetric ID of the article was then used for scrapping tweets discussing the article from altmetric.com. We removed tweets without any content and identical retweets from the collected tweets, and kept only English tweets. Accordingly, we collected a total of 142,198 English tweets for 4,039 articles in the Medicine domain and 21,109 English tweets for 1,000 articles in the Psychiatry domain. Once we scraped the tweets from the two domains, initially 2,000 tweets were labelled manually by three coders. Agreements between the coders were evaluated, and Cohen's kappa values are 0.88, 0.85, and 0.88 for the three pairs of the coders. Then, the rest tweets were labelled independently by one coder. Finally,

1,099 negative, 2,000 positive and 8,000 neutral tweets were obtained from Medicine and Psychiatry fields.

For data cleaning, we removed usernames (the term that follows the mention character @), URLs, non-ASCII text, special characters, punctuations, and extra spaces. After the cleaning step, we expanded English contractions (e.g., expand “you’re” to ‘you’ and ‘are’), converted uppercase characters to lowercase characters, lemmatized words, and replaced the numbers with their word equivalents. Since we are conducting the sentiment analysis of tweets mentioning research articles, the tweets are likely to contain article titles that may contain subjective words that influence the polarity of the tweets while classifying with the model. Therefore, article title words except stop words were removed from the tweets.

For training a model, as the labelled dataset is highly imbalanced and taking all the neutral tweets (8,000) will make the model biased towards neutral class, we used 3,200 neutral tweets and 1,000 tweets for each positive and negative. For test dataset, we used 300 neutral tweets, 200 positive tweets, and 99 negative tweets. We explored various machine learning algorithms, i.e. Support Vector Machine (SVM), Naive Bayes (NB), Random Forest (RF), and Artificial Neural Networks (ANN). Then the SVM model gave the best accuracy of 91.6% with 86%, 93%, and 91% F1 scores for negative, and neutral, and positive classes, respectively, with the following features: bag of words (TF-IDF), number of positive words (used MPQA, a subjectivity lexicon), number of negative words (MPQA), number of positive/negative POS tagged words (MPQA), overall compound score (used Vader, a lexicon and rule-based sentiment analysis tool), compound score of positive words (Vader), compound score of negative words (Vader), polarity of the last word of the tweet, number of hashtags, number of exclamation marks, number of question marks, number of negations, number of user mentions, and number of uppercase characters. Since we are dealing with the imbalanced dataset, class weight was used with the ‘balanced’ option in SVM to avoid the model being biased towards a majority class, i.e. neutral class.

4 Prediction and Discussion

The best SVM model was used to predict the sentiment of the tweets obtained from Medicine and Psychiatry domains. Table 1 shows the sentiment distribution of the tweets associated with the Medicine domain ranging from 2015 to 2017 years and the Psychiatry domain in the year 2016. From the results, we can see that the majority of the users tend to express neutral opinions towards the research articles and the neutral tweet distribution in the Medicine domain is ranging from 79.75% to 89.05% and in the Psychiatry domain, neutral tweets are 87.23%. The percentages of positive opinions expressed in the Medicine domain range from 6.48% to 12.31%, and comparatively the users expressed less negative sentiments towards the research articles, which are ranging from 4.02% to 7.94%. In the Psychiatry domain, 6.7% of the posts carry negative opinions, which are slightly more in percentage compared to 6% of the posts

with positive opinions. From the results, we can infer that people are a bit keener on discussing their negative opinions related to mental health problems compared to the Medicine domain.

Table 1. Polarity distribution of tweets in medicine and psychiatry domains

Polarity	Medicine domain						Psychiatry domain	
	2015		2016		2017		2016	
	Tweet count	% of tweets	Tweet count	% of tweets	Tweet count	% of tweets	Tweet count	% of tweets
Positive	1193	6.48%	6935	7.52%	3888	12.31%	1269	6.00%
Negative	741	4.02%	4151	4.50%	2507	7.94%	1426	6.70%
Neutral	16478	89.05%	81123	87.98%	25182	79.75%	18414	87.23%
Total	18412	100.00%	92209	100.00%	31577	100.00%	21109	100.00%

In Fig. 1, we plot line graphs with the tweet counts of the positive, negative, and neutral tweets over the three years (2015, 2016, and 2017) in the Medicine domain, and we can see the increasing trend of subjective (positive and negative) tweets and decreasing trend of neutral tweets. From this, we can infer that social media platforms such as Twitter are gradually gaining popularity among the general audiences who are willing to express their opinions towards the research articles.

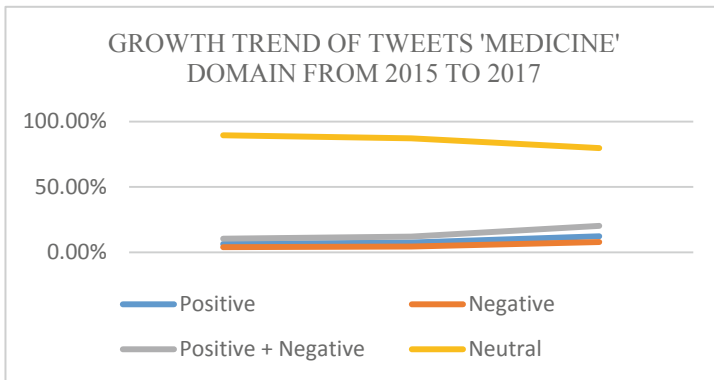


Fig. 1. Growth trend of positive and negative tweets in the medicine domain

References

1. Alternative Metrics Initiative Phase 1 White Paper. National Information Standards Organization (NISO), Baltimore. http://www.niso.org/apps/group_public/download.php/13809/Altmetrics_project_phase1_white_paper.pdf
2. Arredondo, L.: A Study of Altmetrics Using Sentiment Analysis (2018). <http://commons.lib.niu.edu/handle/10843/17878>

3. Na, J.-C.: User motivations for tweeting research articles: a content analysis approach. In: Allen, R.B., Hunter, J., Zeng, M.L. (eds.) ICADL 2015. LNCS, vol. 9469, pp. 197–208. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-27974-9_20
4. Sesagiri Raamkumar, A., Ganesan, S., Jothiramalingam, K., Selva, M.K., Erdt, M., Theng, Y.-L.: Investigating the characteristics and research impact of sentiments in tweets with links to computer science research papers. In: Dobreva, M., Hinze, A., Žumer, M. (eds.) ICADL 2018. LNCS, vol. 11279, pp. 71–82. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-04257-8_7



Measurement and Visualization of IIF Image Usages

Chifumi Nishioka¹ and Kiyonori Nagasaki²

¹ Kyoto University Library, Kyoto, Japan
nishioka.chifumi.2c@kyoto-u.ac.jp

² The University of Tokyo, Tokyo, Japan
nagasaki@dhii.jp

Abstract. In these years, a lot of libraries and museums have adopted IIF (International Image Interoperability Framework), which promotes mutual use of images among institutions. In IIF-compatible digital collections, images are retrieved via IIF Image API with specifying regions of images. Thus, we can investigate the detailed image usage by examining requested regions. In this paper, we demonstrate an application that measures and visualizes which regions of images are looked by users. Specifically, we count the number of accesses to each pixel of images. Since a pixel is the smallest unit that composes an image, it enables to show the detailed image usage. Finally, we visualize the result as heat maps and display heat maps over original images on Mirador, a IIF-compatible image viewer. Thus, users can interactively investigate which regions of images are looked by users with panning and zooming.

Keywords: Visualization · Usage statistics · Digital humanities · Image

1 Introduction

It is important for libraries and museums to understand how digital collections and their contents have been used for many reasons such as improvement of digital collections [2] and accountability for stakeholders. The usage has been analyzed following two steps as below:

1. **Selection of a metric.** A metric that suits for the purpose of the analysis is chosen. Then, the metric is measured based on data such as server logs and loan records. So far, metrics such as the number of accesses to materials (e.g., historical manuscripts) and images have been widely employed [3].
2. **Visualization of the result.** The result of the analysis is visualized to facilitate users to understand. For instance, bar charts and line charts have been used.

In these years, a lot of digital collections from libraries and museums have adopted IIF (International Image Interoperability Framework) [5], which promotes mutual use of images among institutions. IIF defines a set of APIs to

enable interoperable use of images over different institutions. In IIIF-compatible digital collections, images are fetched via IIIF Image API¹, whose syntax is defined as: `{scheme}://{server}/{prefix}/{identifier}/{region}/{size}/{rotation}/{quality}.{format}`. Every time an image is zoomed and panned on an image viewer, the image is called via IIIF Image APIs with varying the value of region. Therefore, we can analyze the detailed image usage by examining the values of region. In this paper, we demonstrate how to measure the detailed image usage and how to visualize the result of the analysis. Specifically, we employ the number of accesses to each pixel as a metric. Since a pixel is the smallest unit that composes an image, we can see the detailed image usage. Finally, we visualize the result of the analysis by heat maps. Heat maps are shown over original images using Mirador², an IIIF-compatible image viewer, which has a function of panning and zooming images. Thus, users (e.g., administrator of a digital collection) can interactively see which regions of images have been looked.

2 Approach to Measure and Visualize IIIF Image Usage

Below, we describe how to measure the number of accesses to each pixel, how to generate heat maps, and how to display heat maps for users in each paragraph.

Measure the Number of Accesses to Each Pixel. For each image, an $H \times W$ matrix, where all elements are 0, is generated. H and W are the height and width of the image in pixel, respectively. Thus, each element of the matrix corresponds to each pixel of the image. The height and width of images are programatically retrieved by `info.json`³ provided by the IIIF Image API. Subsequently, the requested images and regions of each image are acquired by parsing server logs of IIIF Image API. Based on the requested regions, the number of accesses to each pixel of each image is counted and recorded to the matrices.

Generation of Heat Maps. After counting the number of accesses to each pixel, the result is outputted as a heat map. The RGB value of each pixel is calculated considering the minimum and maximum values of the number of accesses to pixel in an image.

Display of Heat Maps. Generated heat maps are displayed over corresponding target images, in order to enable users (e.g., administrator) to understand the detailed image usage. The IIIF Presentation API enables to overlay images by specifying to display the two images on a page. Mirador, one of the most popular image viewers among the IIIF community, implements a function of overlay display of images as shown in Fig. 1. One can manipulate the visibility and opacity for each image in the left side panel and interactively see which regions of images are looked by users with panning and zooming.

¹ <https://iiif.io/api/image/>.

² <https://projectmirador.org/>.

³ <https://iiif.io/api/image/2.1/#image-information-request-uri-syntax>.

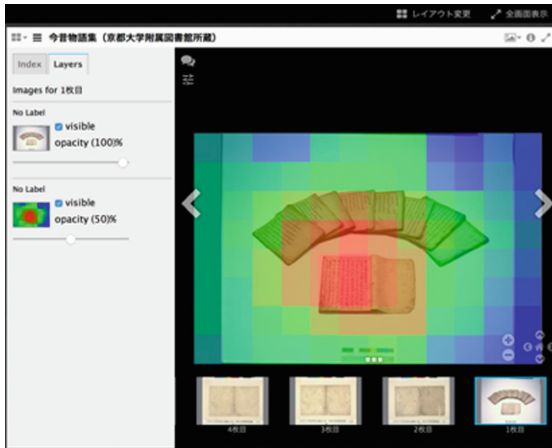


Fig. 1. Overlay display of a heat map and its target image using Mirador. Photograph courtesy of the Main Library, Kyoto University - Konjaku monogatarihuu.

We show two examples of the result of the analysis of the image usage in below. In Fig. 2a, we observe a slight tendency that accesses are made along with the Japanese archipelago. As shown in Fig. 2b, we observe images with concentrated accesses in specific regions. Referrers of server logs reveal that these regions are referenced from an external platform. For the future work, indication of the referrer might enable to show the motivation and background behind accesses to users.

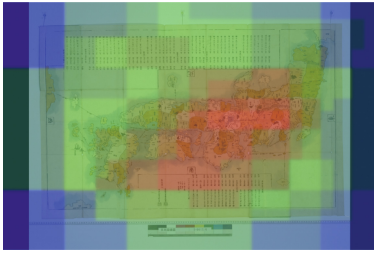
3 Possible Applications

This section lists possible applications of the analysis of the image usage.

Collaborative Research Platform. The data model used in IIF follows the Web Annotation Data Model⁴. Thus, IIF facilitates to share not only images but also information accompanying images (e.g., annotations such as transcriptions). For this reason, IIF-compatible collaborative research platforms have been developed [4]. Heat maps enable researchers to understand regions of images already examined by collaborators.

Transcription Platform. In the decades, a lot of transcription projects and platforms have been launched [1]. Transcribers zoom and pan images during generating transcriptions. If a platform is compatible with IIF, it is possible to verify a pattern such as whether there is a difference in transcription performance (e.g., accuracy) between regions being zoomed and those being not zoomed. If we find a pattern, we can facilitate verification process for transcriptions.

⁴ <https://www.w3.org/TR/annotation-model/>.



(a) An image where we observe a slight tendency that accesses are made along the Japanese archipelago. Photograph courtesy of the Main Library, Kyoto University - Nihonkokubizu.



(b) An image with concentrated accesses in specific regions. Photograph courtesy of the Main Library, Kyoto University - The story of Benkei, a tragic warrior.

Fig. 2. Examples of the result of the analysis of the image usage

4 Conclusion

In this paper, we show a demonstration, where the detailed image usage are measured and visualized for IIF-compatible digital collections. Specifically, we count the number of accesses to each pixel of images. Thus, it enables the fine-grained analysis of image usage, because a pixel is the smallest unit of an image. Finally, we visualize the result as heat maps and show heat maps over target images using Mirador. Therefore, users can interactively investigate which regions of images are looked with panning and zooming.

References

1. Carletti, L., Giannachi, G., Price, D., McAuley, D., Benford, S.: Digital humanities and crowdsourcing: an exploration. In: *Museums and the Web* (2013)
2. Hughes, L.: Evaluating and Measuring the Value, Use and Impact of Digital Collections. *Facet* (2011). <https://doi.org/10.29085/9781856049085>
3. Jones, S., Cunningham, S.J., McNab, R., Boddie, S.: A transaction log analysis of a digital library. *Int. J. Digit. Libr.* **3**(2), 152–169 (2000)
4. Sato, M., Ota, I.: Collaboration system based on crowdsourcing with Mirador - proposal of a system to support analysis and theory in collaborative research of humanities. *IPSJ SIG-CH Technical Report 2017-CH-114(7)*, pp. 1–6 (2017)
5. Snyderman, S., Sanderson, R., Cramer, T.: The international image interoperability framework (IIF): a community & technology approach for web-based images. In: *Archiving Conference*, vol. 2015, pp. 16–21. Society for Imaging Science and Technology (2015)



Improving OCR for Historical Documents by Modeling Image Distortion

Keiya Maekawa^(✉), Yoichi Tomiura, Satoshi Fukuda, Emi Ishita, and Hideaki Uchiyama

Kyushu University, 744 Motoooka, Nishi-ku, Fukuoka, Japan
maekawa.keiya.946@s.kyushu-u.ac.jp,
tom@inf.kyushu-u.ac.jp, {fukuda.satoshi.528,
ishita.emi.982, uchiyama.hideaki.667}@m.kyushu-u.ac.jp

Abstract. Archives hold printed historical documents, many of which have deteriorated. It is difficult to extract text from such images without errors using optical character recognition (OCR). This problem reduces the accuracy of information retrieval. Therefore, it is necessary to improve the performance of OCR for images of deteriorated documents. One approach is to convert images of deteriorated documents to clear images, to make it easier for an OCR system to recognize text. To perform this conversion using a neural network, data is needed to train it. It is hard to prepare training data consisting of pairs of a deteriorated image and an image from which deterioration has been removed; however, it is easy to prepare training data consisting of pairs of a clear image and an image created by adding noise to it. In this study, PDFs of historical documents were collected and converted to text and JPEG images. Noise was added to the JPEG images to create a dataset in which the images had noise similar to that of the actual printed documents. U-Net, a type of neural network, was trained using this dataset. The performance of OCR for an image with noise in the test data was compared with the performance of OCR for an image generated from it by the trained U-Net. An improvement in the OCR recognition rate was confirmed.

Keywords: OCR error · Information retrieval · Historical document image

1 Introduction

In the modern age, information is often provided and circulated as digital data. In contrast, many archives and libraries still hold printed historical records and documents. Even in those institutions, some of the records have already been digitized: many of them are not in text form but only exist as image data. Our ultimate goal is to help researchers search these documents effectively and efficiently to find documents related to their information needs. However, image format is not suitable for searching for information. It is necessary to extract text information from the images to enable search for information related to researchers' needs. We focus on old records and documents, as shown in Fig. 1, most of which have deteriorated. In this paper, we call an image of a deteriorated document a *distorted image*. It is difficult to automatically

convert distorted images to correct text with optical character recognition (OCR). This causes low accuracy of information retrieval. In order to solve this problem, not only development of robust search method [1] for text containing OCR errors, but also improvement of the performance of OCR for a distorted image [2] is necessary.

In this research, we attempt to improve the recognition accuracy of OCR, by preprocessing a distorted image with a neural network to generate a clear (undistorted) image for input to the OCR. We need to construct a very large amount of training data to train the neural network that converts the image. It is difficult to manually create clear images not containing deterioration from distorted images. However, it is relatively easy to automatically add noise to clear images so that the created images mimic the deterioration in the actual historical documents.

We collected portable document format (PDF) files of official documents that did not exhibit deterioration, automatically added noise mimicking deterioration to the images of the collected documents and created pairs combining each distorted image with the original clear image. Then, using a part of the dataset, we then trained U-Net, a type of neural network, to generate a clear image from a distorted image. Finally, we investigated the effect of preprocessing by the trained neural network, using the rest of the dataset. The experimental results indicated a significant improvement in the recognition accuracy of OCR by preprocessing with the trained U-Net.

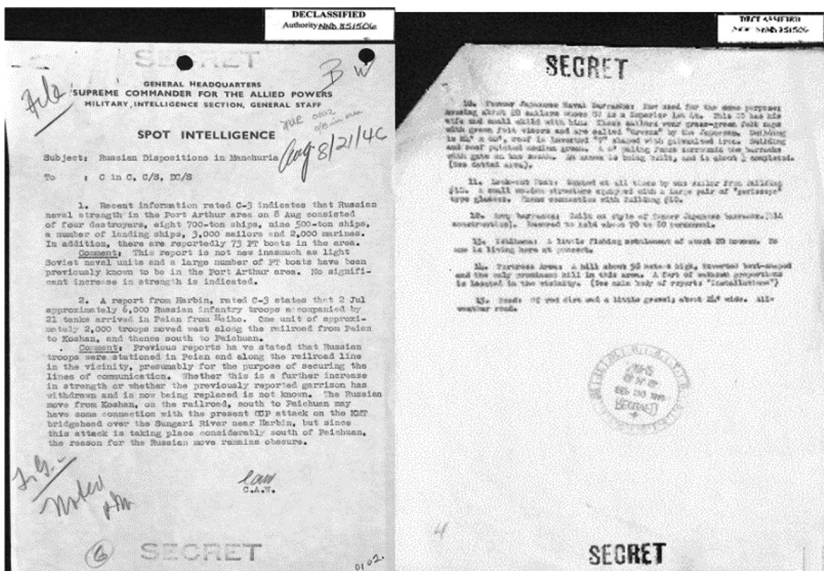


Fig. 1. Real historical printed material images

2 Generating Distorted Image

The target of this research is the collection of images of the historical documents of RG554 in the digital collection of the Japan National Diet Library (Fig. 1). The original RG554 is preserved at the U.S. National Archives. By observing these distorted images, we confirmed that the distortions were due to partial paper discoloration, contrast reduction, feathering, ghosting, and noise caused by dust and blur, as shown in Fig. 2. We generated distorted images by adding noise imitating these deteriorations to clear images of documents having a format similar to the target documents. We added Gaussian noise (mean = 0, standard deviation = 0.2) to imitate paper discoloration; added salt-and-pepper noise (replace rate = 0–0.01) before smooth blurring (filter size = 3–5) to imitate feathering, blur of character, and dust on scanning; and reduced contrast (by converting the pixel value range from [0, 255] to about [50, 210]) randomly.

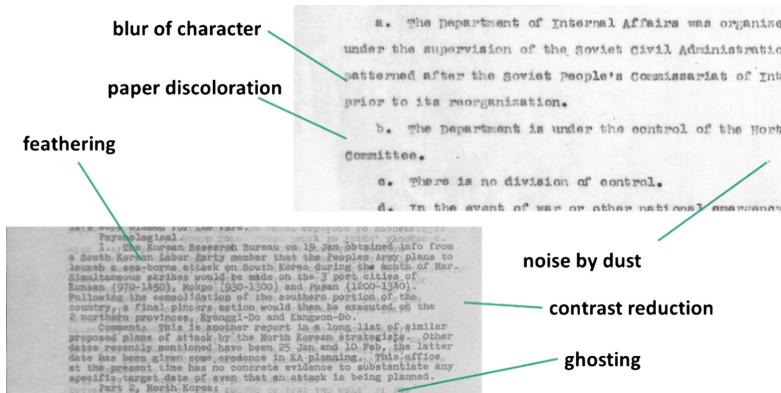


Fig. 2. Types of distortion in historical printed material images

3 Experiment

3.1 Experimental Setting

U-Net, a type of CNN composed of only convolution layers, was used for the architecture of the neural network in the experiment. The training data described in the previous section were used as input data and label data, because both input and output were images [3]. The output image was the binary conversion of the input image. The optimization algorithm for mini-batch learning was set to Adam, the base learning rate was set to 10^{-4} , and binary cross entropy was used as an error function.

First, we collected 27 PDFs of documents as clear images, through the United Nations official document retrieval system. The format of these documents was similar to the target documents. The total number of pages in the collected documents was 440. We then converted them to images with a size of 1024×1024 in the JPEG format,

which we regarded as clear (undistorted) images. We generated distorted images from these clear images using the method described in Sect. 2. Finally, we converted the image files to text using a commercial OCR software, for evaluation purposes.

We trained U-Net using 404 pairs comprising a distorted image and its original clear image. We measured the error rate of OCR for each of the remaining 36 distorted images. We then measured the error rate of OCR for each of the images generated from the 36 distorted images by the trained U-Net. To calculate the OCR error rate, we used the text generated by OCR from the original clear images as the correct data.

3.2 Result and Discussion

We compared the error rate of OCR for text generated from two categories of images: *Noise* and *Predict*. *Noise* means the images to which noise (distortion) has been added, as described in Sect. 2, and *Predict* means the images generated from the images in *Noise* by the trained U-Net. We used the evaluation tool *ocrevalUAtion* for the calculation of character error rate (CER) and word error rate (WER). Table 1 shows the results of the experiment. This indicates that removing the image noise by the trained U-Net increased the accuracy of OCR.

Table 1. CER and WER of generated text

Data image	CER	WER
Noise	78.98	87.49
Predict	9.46	30.20

4 Conclusion and Future Work

We proposed a method for converting images of deteriorated historical printed documents to text. In our method, images of deteriorated documents are converted to images not containing deterioration, and text is extracted from the converted images using a commercial OCR software. The experimental results indicated that our approach has great potential. An OCR system based on a CNN architecture could directly extract text from deteriorated document images. However, this approach would also require numerous pairs comprising a deteriorated image and the extracted text from it, for training. Our approach could be effectively adapted to this case. In the future, we will apply a method for creating distorted images that further mimic the actual deterioration, using Cycle-GAN [4].

References

1. Ghosh, K., Chakraborty, A., Parui, S.K., Majumder, P.: Improving information retrieval performance on OCRred text in the absence of clean text ground truth. *Inf. Process. Manag.* **52** (5), 873–884 (2016)

2. Chen, Y., Wang, L.: Broken and degraded document images binarization. *Neurocomputing* **237**, 272–280 (2017)
3. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) *MICCAI 2015*. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
4. Zhu, J.-Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: *ICCV*, pp. 2223–2232 (2017)

Author Index

- Adhikari, Bibhas 71
Alabdulkarim, Sarah 205
Alexeev, Alexey 23
Al-Khalifa, Hend 205
Almatarneh, Sattam 23
Al-Matham, Rawan 205
Alotaibi, Reem 205
Ateeq, Wed 205
- Baskaran, Subashini 40
Bharathwaj, Sampathkumar Kuppan 303
Bhowmick, Plaban Kumar 213
Bruder, Ilvio 139
- Chen, Xiaoyu 33
Chen, Xingyu 16
Collier, Edward 3
- Das, Ananda 181
Das, Partha Pratim 181
Doucet, Antoine 102
Du, Baoxiang 263
Duan, Xu 33
Dube, Meluleki 283
Düffer, Martin 139
- Erdt, Mojisola 54
- Fukuda, Satoshi 312
- Galušćáková, Petra 167
Gamallo, Pablo 23
Goyal, Pawan 71
Gruszczyński, Włodzimierz 125
Gupta, Shreya 40
- Hamdi, Ahmed 102
Han, Hae Jin 263
Hauke, Jörn 63
Hazra, Rima 71
Heuer, Andreas 139
Huang, Haojie 263
- Ikeda, Daisuke 167
Ishita, Emi 312
Iwaihara, Mizuho 16
- Jain, Naman 78
- Koizumi, Masanori 298
Kreutz, Christin Katharina 239
Kumar, Mahalakshmi Suresh 40
Kuzumi, Ryohei 298
- Linhares Pontes, Elvys 102
Liu, Qun 3
Lumpa, Mushashu 151
- Maekawa, Keiya 312
Matsubara, Masaki 298
Matsuda, Yuhei 298
Meng, Fansong 293
Mihara, Tetsuya 95
Miyata, Rei 249
Morishima, Atsuyuki 298
Mukherjee, Animesh 71
Mukhopadhyay, Supratik 3
- Na, Jin-Cheon 40, 303
Nagamori, Mitsuharu 95, 116
Nagarajan, Aarthi 54
Nagasaki, Kiyonori 308
Nishioka, Chifumi 63, 308
- Oard, Douglas 167
Ogrodniczuk, Maciej 125
Oishi, Kosuke 95
Oliveira, Duarte 271
- Pena, Francisco J. Ribadas 23
Portela, Manuel 271

Sadhu, Susmita 213
Sakaguchi, Tetsuo 116
Sangeetha, Babu 303
Sarathkumar, Eswaran 303
Sato, Satoshi 249
Schenkel, Ralf 239
Scherp, Ansgar 63
Sesagiri Raamkumar, Aravind 54
Sidere, Nicolas 102
Silva, António Rito 271
Singh, Mayank 71, 78
Sugimoto, Shigeo 95, 116
Sugino, Hodai 249
Suleman, Hussein 151, 227, 283
Suzuki, Tokinori 167

Thalhath, Nishad 116
Theng, Yin-Leng 54
Tomiura, Yoichi 312

Uchiyama, Hideaki 312

Vanderschantz, Nicholas 189
Vijayakumar, Harsha 54

Wolz, Michael 239
Wong, Raymond K. 263
Wu, Yejun 293

Yang, Zhiqin 189

Zheng, Han 33