2nd International Workshop on News Recommendation and Analytics (NRA2014)

In conjunction with 22nd Conference on User Modelling, Adaptation and Personalization (UMAP 2014) 11 July 2014, Aalborg, Denmark

Jon Atle Gulla, Ville Ollikainen, Özlem Özgöbek, and Nafiseh Shabib

Department of Computer and Information Science
Norwegian University of Science and Technology
Trondheim, Norway
VTT Technical Research Centre of Finland, Finland
{jag, ozlemo,shabib}@idi.ntnu.no, ville.ollikainen@vtt.fi

Abstract. The 2nd International Workshop on News Recommendation and Analytics (NRA) brings together researchers on news analytics and stakeholders from the media industry. A particular focus is on news recommender systems, that tailor content from media houses and social sites to the preferences and context of individual readers. The workshop includes one invited speaker from the media industry and five academic papers addressing different aspects of news recommendation.

1 Preface

As the amount of data on the internet increases it is getting harder to find the information that people are looking for. Recommender systems are built to bring the most relevant information to users within the huge amount of data on the internet using the users' personal interests and preferences. Even though there is steady progress in recommender systems and also visible progress in news recommender systems, there are many challenges that need to be solved or improved for the systems to receive widespread acceptance. Compared to recommender systems in domains like music, movies and books, news recommender systems pose some particular challenges that call for new and deeper analyses of both users and content: The news domain is marked by (i) dynamic streams of news articles where different news sources on the internet publish hundreds of new articles every hour, (ii) willingness to read news articles that are independent from user interests like breaking news, (iii) unstable user interests that change much faster than in other domains (the taste of movies or food of a user takes years to change), (iv) recency issues that render old news stories less interesting than recent ones, and (v) unstructured subjective content that create content analysis problems and may turn recommendations unreliable. These issues also complicate the modelling and monitoring of user interests and preferences, since users are not giving explicit signals of their interests and information about

users need to be deduced from their observed attitude towards news. The news domains intrinsic complexities combined with the commercial interests of media companies is a good basis for innovative approaches to both news content analysis and news recommendation.

The news domain is characterized by a constant flow of unstructured, fragmentary, and unreliable news stories from numerous sources and different perspectives. Finding the right information, either in terms of individual news stories or aggregated knowledge from analyzing entire news streams, is a tremendous challenge that necessitates a wide range of technologies and a deep understanding of user preferences, news contents, and their relationships.

This workshop addresses primarily news recommender systems and news analytics, with a particular focus on user profiling and techniques for dealing with and extracting knowledge from large-scale news streams. The news streams may originate in large media companies, but may also come from social sites, where user models are needed to decide how user-generated content is to be taken into account. This workshop aims to create an interdisciplinary community that addresses design issues in news recommender systems and news analytics. It intends to bring together researchers, media companies, and practitioners around the topics of designing and evaluating novel news recommender systems and analytics in order to: (1) share research on news recommendation techniques and evaluation methodologies (2) explore key components in news analytics and solutions, and (3) identify emerging research topics in the news domain.

Topics of interests include but are not limited to:

- News semantics and ontologies
- News summarization, classification and sentiment analysis
- Recommender systems and news personalization
- Group recommendation for news
- User profiling and news context modeling
- News evolution and trends
- Large-scale news mining and analytics
- Evaluation methods
- News from social media
- Big Data technologies for news streams
- News recommendation and analytics on mobile platforms

2 Program Committee

- Paolo Rosso, Universidad Politcnica de Valencia, Spain
- Bei Yu, Syracuse University, USA
- Francesco Ricci, Free University of Bozen-Bolzano, Italy
- Alejandro Bellogin, Universidad Autonoma de Madrid, Spain
- Mohamad Ali Nematbakhsh, University of Isfahan, Iran
- Xiaomeng Su, Telenor Group, Norway
- Olli Alm, Helsinki Metropolia University of Applied Sciences, Finland

– Donn Morrison, NTNU, Norway
– Ido Guy, IBM, Israel
– Bahareh Heravi, Digital Enterprise Research Institute, Ireland
– Nava Tintarev, University of Aberdeen, Scotland, U.K.
– Humberto Castejon, Telenor Group, Norway
– Ville Ollikainen, VTT Technical Research Centre of Finland, Finland
– Jon Atle Gulla, NTNU, Norway
– Nafiseh Shabib, NTNU, Norway
– Özlem Özgöbek, NTNU, Norway and Ege University, Turkey

3 Accepted Papers

The workshop is composed of paper presentations and a key note speech on news analytics and recommendation in data driven journalism. It aims to create an interdisciplinary community that addresses design issues in news recommender systems and news analytics, and promote fruitful collaboration opportunities between researchers, media companies and practitioners.

– Keynote Speech - *Alexander Øhrn, Cxense*
– Data Sets and News Recommendation - *Özlem Özgöbek, Nafiseh Shabib and Jon Atle Gulla* [3]
– Using a Rich Context Model for a News Recommender System for Mobile Users - *Alisa Sotsenko, Marc Jansen and Marcelo Milrad* [1]
– Stories around You: Location-based Serendipitous Recommendation of News Articles - *Yonata Andrelo Asikin and Wolfgang Wörndl* [2]
– Method for Novelty Recommendation Using Topic Modelling - *Matúš Tomlein and Jozef Tvarožek*[4]
– Building Rich User Profiles for personalized news recommendations - *Youssef Meguebli, Mouna Kacimi, Bich-Lien Doan and Fabrice Popineau* [5]

4 Previous Workshops

The workshop on News Recommendation and Analytics is based on the previous International News Recommender Systems Workshop and Challenge [1] that was held in conjunction with the 7th ACM Recommender Systems Conference in 2013. With this workshop we have expanded the scope with news analytics, which is closely linked with news recommendation and should encourage more submissions.

[1] http://recsys.acm.org/recsys13/nrs/

References

1. M. J. Alisa Sotsenko and M. Milrad. Using a rich context model for a news recommender system for mobile users. In *Proceedings of 2nd International Workshop on News Recommendation and Analytics*, 2014.
2. Y. A. Asikin and W. Wörndl. Stories around you: Location-based serendipitous recommendation of news articles. In *Proceedings of 2nd International Workshop on News Recommendation and Analytics*, 2014.
3. J. A. G. Özlem Özgöbek, Nafiseh Shabib. Data sets and news recommendation. In *Proceedings of 2nd International Workshop on News Recommendation and Analytics*, 2014.
4. M. Tomlein and J. Tvarožek. Method for novelty recommendation using topic modelling. In *Proceedings of 2nd International Workshop on News Recommendation and Analytics*, 2014.
5. B.-L. D. Youssef Meguebli, Mouna Kacimi and F. Popineau. Building rich user profiles for personalized news recommendations. In *Proceedings of 2nd International Workshop on News Recommendation and Analytics*, 2014.

Data Sets and News Recommendation

Özlem Özgöbek, Nafiseh Shabib, Jon Atle Gulla

Department of Computer and Information Science
Norwegian University of Science and Technology
Trondheim, Norway
{ozlemo,shabib,jag}@idi.ntnu.no

Abstract. Datasets are important for training and testing many information processing applications. In the field of news recommendation, there are still few available datasets, and many feel obliged to use non-news datasets to test their algorithms for news recommendation. This paper presents some of the most common datasets for recommender systems in general, and explains why these datasets do not fully satisfy the needs in news recommendation. We then discuss the ongoing process of building up an entirely new dataset for Norwegian news in the SmartMedia project. In particular, we go through some of the features of news datasets that separate them from many other datasets and are crucial for their use in news recommendation.

1 Introduction

A dataset is a collection of data that is used to train and test new systems under development. As real systems work on data, it is vital to validate and verify their behavior with extensive datasets prior to their deployment. Moreover, with the increasing popularity of data-driven learning applications, high-quality datasets have become critical for training these applications to perform at an acceptable level of precision.

Scientific methods rest on systematic use of measurements and their subsequent analysis. According to [4], datasets serve at least four different purposes in scientific research: Verification of publications (scientific publications can be verified by repeating the same study with the same data), longitudinal research (long term availability of the data for a long period of research), interdisciplinary use of data (usage of the same dataset for different purposes may lead to new insights and scientific development), and valorisation (Dataset ownership enables the acquisition of new research projects [4]). Also, Dekker claims that datasets are becoming more valuable as products by themselves and justify their own publications in the scientific community.

The nature of datasets depends on both the type of application and the choice of domain. In news recommendation, both machine learning techniques and traditional search technologies are applied and need to be verified against suitable datasets. Machine learning techniques require training and test datasets that are feature-rich and may involve aspects that are directly present in the news

itself, while search applications are usually tested with a narrower focus and no regard of user differences.

More complicated, though, is the fact that it is difficult to replicate the news domain as a fixed controlled document set. We want the datasets to mirror the users preference of news in real news contexts, which means that we need the dynamic and unpredictable nature of news to be reflected in the way these datasets are built up. This is a challenging task and partly explains why there are only a few small news datasets and no large-scale datasets available.

2 Related Work and Comparison of Existing Datasets

Recommender system has been identified as the way to help individuals to find information or items that are most likely to be interesting to them or to be relevant to their needs [1] and it is still very interesting area in the research and real world setting. Thus, monitoring the operation of a recommender system is a challenging task and it is common to evaluate recommendation algorithms with available public dataset (e.g. MovieLens, Netflix, Million Song Dataset). Furthermore, the datasets are used as benchmarks to develop new recommendation algorithms and to compare them to other algorithms in given settings [10]. In the news domain, recommender systems are increasingly applied, but still we are facing lack of publicly available dataset that completely interoperate in news domain. In this section, we present an overview of different datasets, which are available in different domains and then in the next section we introduce our dataset.

2.1 MovieLens Dataset

MovieLens is a movie recommender system project at the University of Minnesota, led by the GroupLens Research Group [1]. There are three datasets of different sizes that have been collected in different time periods [2]. All data is collected through the MovieLens web site. The 100K and 1M datasets contain simple demographic information about the users (age, gender, occupation, zip) while the 10M data set only contains user id. For the 100k dataset the data was collected during the seven-month period from 19 September 1997 to 22 April 1998. For the 1M dataset the data was collected from 6040 users who joined MovieLens in 2000. The 10M dataset contains $10,000,054$ ratings (ranging from 1 to 5) and $95,580$ tags applied to $10,681$ movies by $71,567$ users [3].ngs and 95580 tags applied to 10681 movies by 71567 users [4].

[1] http://grouplens.org
[2] http://grouplens.org/datasets/movielens/
[3] http://files.grouplens.org/datasets/movielens/ml-10m-README.html
[4] http://files.grouplens.org/datasets/movielens/ml-10m-README.html

2.2 Netflix Dataset

On October 2, 2006, Netflix, the world's largest online DVD rental service, announced the 1-million Netflix Prize for improving their movie recommendation service [5]. To aid contestants, Netflix publicly released a dataset containing $100,480,507$ movie ratings, created by $480,189$ Netflix subscribers between December 1999 and December 2005.

2.3 MoviePilot Dataset

The MoviePilot dataset was released as part of the Context-Aware Movie Recommendation 2011 Challenge at ACM RecSys. There were two tracks in this challenge. In the Context-Aware Movie Recommendation (CAMRa) Challenge [9] they requested participants to identify which members of particular households were responsible for a number of event interactions with the system in the form of ratings. The contest provided a training dataset with information about ratings in a movie RS, including the household members who provided the ratings, and the associated time stamps. The goal was to identify the users who had been responsible for certain events (ratings), and whose household and time stamp were given in a randomly sampled test dataset. This task is assumed to be equivalent to the task of identifying active users requesting recommendations at a particular time. In another track, the main task of the challenge was recommending a given set of items to a household of users. The MoviePilot dataset contains 290 unique households with between two to four members, and a total of 602 users, of which the majority has been assigned to a particular household. The dataset contains information about which user rated which movie at which time. More details are shown in Table 1.

Datasets	Movies	Users	Ratings
Training	$23,974$	$171,670$	$4,536,891$
Household in training	$7,710$	602	$145,069.$
Test	811	594	4482

Table 1. The MoviePilot dataset characteristics

2.4 Million Song Dataset

The Million Song Dataset (MSD) [2] is a collection of music audio features and metadata that has created to support research into industrial-scale music information retrieval [6]. The Million Song Dataset (MSD), a freely-available collection

[5] http://www.netflixprize.com
[6] http://labrosa.ee.columbia.edu/millionsong

of meta data for one million of contemporary songs (e.g,. song titles, artists, year of publication, audio features, and much more) [7].
The Million Song Dataset is a cluster of complementary datasets contributed by the community: SecondHandSongs dataset for cover songs, musiXmatch dataset for lyrics, Last.fm dataset for song-level tags and similarity, and Taste Profile subset for user data. Comprising several complementary datasets that are linked to the same set of songs, the MSD contains extensive meta-data, audio features, and song-level, lyrics, cover songs, similar artists, and similar songs. In Lastfm dataset, songs have different tags with different degrees. The tag's degree shows how much the song is linked to a particular tag. Some of the characteristics of Millions of song Million Song Dataset are shown in Table 2.

Songs	$1,000,000$
Data	273 GB
Unique artists	$44,745$
Unique terms	$7,643$
Unique musicbrainz tags	$2,321$
Artists with at least one term	$43,943$
Asymmetric similarity relationship	$2,201,916$
Dated tracks starting from 1922	$515,576$

Table 2. The Million Song Dataset characteristics [2]

2.5 Last.fm Dataset

Last.fm dataset is one of the largest music recommender system datasets [3]. It contains $359,347$ unique users and $17,559,530$ of total lines which includes -user, artist, plays- tuples collected from Last.fm API [7]. This data was collected by Oscar Celma @ MTG/UPF, during Fall 2008 [8], and the it is available for non-commercial use.

This dataset contains user profile information as gender, age, subscription date, country, name. It also contains information about which user listened to which artist and how many times as the user name, artist id, artist name, number of plays.

The Last.fm dataset contains only the artist information that a user listened to. By looking at the number of plays we can figure out the users' most popular artists and the similarities between users' preferences. It is not possible to assess the similarities between artists or songs. So this dataset is mostly suitable for training collaborative filtering methods for artist recommendation. Since there is

[7] http://last.fm/
[8] http://ocelma.net/MusicRecommendationDataset/lastfm-360K.html

no information about the individual songs, it is not possible to recommend a song that the user has not listened to yet. The age, gender and country information can be used for group recommendations. For example, if there is an artist who is mostly listened to 22-25 year old people, it may be possible to recommend it to other users in the same age group and have not listened to it yet.

2.6 Jester Dataset

Jester is an online joke recommender system which has three different versions of publicly available collaborative filtering dataset [5]. The first version of Jester dataset contains over 4 million continuous ratings collected from $73,421$ users. There are 100 jokes in the dataset and it is collected between April 1999 - May 2003. The second version contains over 1.7 million continuous ratings of 150 jokes from $59,132$ users and it is collected between November 2006 - May 2009. Also there is an updated version of the second dataset with over $500,000$ new ratings from $79,681$ total users. [9] The ratings of Jester dataset is in range between -10.00 and $+10.00$ as a floating number. The dataset contains two files where the first one includes the item ID and the jokes, and the other one includes user ID, item ID and ratings.

2.7 Book-Crossing Dataset

Book-Crossing dataset is collected by Cai-Nicolas Ziegler [12] in 4-weeks from August to September 2004. [10] The dataset contains $278,858$ users, about $271,379$ books and $1,149,780$ both explicit and implicit ratings. In the dataset the demographic information is also provided. For the user privacy the demographic data is anonymized. The Book-Crossing dataset includes 3 tables: BX-Users (user ID, location, age), BX-Books (ISBN, book title, author, publisher, year of publication) and BX-Book-Ratings (explicit ratings from 1 to 10, implicit ratings expressed by 0).

2.8 YOW Dataset

YOW dataset is collected at the Carnegie Mellon University for the Yow-now news filtering system. Yow-now was an information filtering system that delivered news articles to users from various RSS feeds. [11] Within this project the data is collected by a one month user study which includes approximately 25 people and 7000+ feedback entries from all users. In total 383 articles rated by each user. It is collected both implicit and explicit feedback from users. Explicit feedback is collected as rating from 1 to 5 and explicit feedback is collected by tracking the user actions (mouse, keyboard and scroll activities) during the usage of the system [11].

[9] http://eigentaste.berkeley.edu/dataset/
[10] http://www.informatik.uni-freiburg.de/ cziegler/BX/
[11] http://users.soe.ucsc.edu/ yiz/papers/data/YOWStudy/

The YOW dataset contains a lot of details about the user actions while reading news. Both explicit and implicit feedbacks are available in the dataset, making this dataset well suited for collaborative filtering. Since there is no information about news content, content-based filtering is not possible with this dataset. YOW dataset is the only publicly available dataset that we could find on the news domain.

	Domain	Size			Feedback	
		Items	Users	Ratings	Explicit	Implicit
MovieLens 100k	Movie	1682 movies	943	100,000	Ratings from 1 to 5	-
MovieLens 1M	Movie	3900 movies	6040	1,000,209	Ratings from 1 to 5	-
MovieLens 10M	Movie	10682 movies	71567	10,000,054	Ratings from 1 to 5	-
Netflix (Training)	Movie	17,770 movies	480,189	100,480,507	Ratings from 1 to 5	-
MoviePilot (Training)	Movie	23,974 movies	171,670	4,536,891	Ratings	-
Last.fm	Music	186,642 artists	359,347	17,559,530	Ratings	-
Million Song (cluster of complementary datasets)	Music	1,000,000 songs	-	-	-	-
Jester v1	Joke	100 jokes	73,421	4,000,000	Ratings from -10.00 to +10.00	-
Jester v2	Joke	150 jokes	59,132	1,700,000	Ratings from -10.00 to +10.00	-
Book-Crossing	Book	271,379 books	278,858	1,149,780	Ratings from 1 to 10	✓
YOW	News	383 articles	25	7000+	Ratings from 1 to 5	Mouse, keyboard and scroll activities

Table 3. Comparison of different datasets of recommender systems and their properties.

3 SmartMedia Dataset

The specific challenges of news domain requires the usage of a special dataset for testing the news recommender system. Within our SmartMedia project [6] we are building a dataset on Norwegian news domain.

As a specific challenge to news domain, there can be hundreds of new articles every hour and it is not always possible to get enough ratings to overcome the problem of data sparsity. In SmartMedia dataset we are trying to build a dataset which is less sparse than other datasets in news domain by limiting the number

of news articles gathered from different sources. As it is stated in [8] implicit feedback is one of the challenges that both it is needed to be collected and considered the user privacy issues.

SmartMedia dataset will contain both explicit and implicit feedback from users. As the explicit feedback, we get ratings from 1-5 for each news article. Implicit feedback contains the time spent on each article, current location and timestamp. The dataset will also contain some personal information like occupation, age and gender. We developed an application to collect the data from users. We recruited 20 users with different backgrounds, occupations and within different age groups to make the dataset more homogeneous and realistic. We asked the users to read and rate the news articles which are collected from different Norwegian news sources for a period of two weeks. As a result of our data collection process we expect nearly 8500 ratings of 3000 articles.

4 Discussion

The properties of different data sets of recommender systems is given in Table 3. We have chosen to compare these data sets because most of them are very well known, publicly available and regularly used data sets in the recommender system research. Since each domain have its own specific challenges, we also wanted to compare data sets from different domains as much as possible. Recommending news articles has different challenges than recommending movies or music [8]. For example, for recommending movies, learning the users' preferences/tastes about movies can be enough. But for the news domain one may find the article important even though she does not like the topic or she may not want to read the other articles in the same topic. For the news recommendation YOW data set was the only publicly available data set with ratings which is also suitable for collaborative filtering.

Nearly all the datasets that we compare have explicit feedback from the users. So asking the user how much she liked the recommendation (rating) is the most common way to get feedbacks. Usually the ratings are integers ranging from 1 to 5 or 1 to 10. Only the Jester dataset ratings are not integers.

Most of the datasets do not have implicit feedback. Only the YOW dataset has a lot of implicit feedbacks. Since recommending news articles is different than recommending items in other domains in many aspects like recency (the people usually want to read fresh news) and quick or instantaneous changes of user interest (age, cultural level, mood or on going circumstances in the world may affect the preferences of users), the need for implicit ratings is more demanding [8]. So to develop better recommender systems in specific domains it is important to choose the suitable dataset. By the SmartMedia dataset we are aiming to have a realistic and less sparse dataset (compared to the YOW dataset) in Norwegian news domain including explicit and implicit feedbacks.

5 Conclusion

Since each domain has its own specific requirements for the recommender systems, the need for choosing the suitable dataset for developing and improving systems is quite obvious. Especially the news domain is very different in many aspects like the item churn and recency compared to the other domains.

In this paper we provided a comparison of recommender system datasets from different domains and we presented our SmartMedia news recommender system dataset which will be the first publicly available dataset in the Norwegian news domain.

References

1. G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. on Knowl. and Data Eng.*, 17(6):734–749, June 2005.
2. T. Bertin-Mahieux, D. P. Ellis, B. Whitman, and P. Lamere. The million song dataset. In *Proceedings of the 12th International Conference on Music Information Retrieval (ISMIR 2011)*, 2011.
3. O. Celma. *Music Recommendation and Discovery in the Long Tail*. Springer, 2010.
4. R. Dekker. The importance of having data-sets. 2006.
5. K. Goldberg, T. Roeder, D. Gupta, and C. Perkins. Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval*, 4(2):133–151, 2001.
6. J. A. Gulla, J. E. Ingvaldsen, A. D. Fidjestl, J. E. Nilsen, K. R. Haugen, and X. Su. Learning user profiles in mobile news recommendation. pages 183–194, 2013.
7. B. McFee, T. Bertin-Mahieux, D. P. Ellis, and G. R. Lanckriet. The million song dataset challenge. In *Proceedings of the 21st International Conference Companion on World Wide Web*, WWW '12 Companion, pages 909–916. ACM, 2012.
8. O. Ozgobek, J. A. Gulla, and R. C. Erdur. A survey on challenges and methods in news recommendation. In *In Proceedings of the 10th International Conference on Web Information System and Technologies (WEBIST 2014)*, 2014.
9. A. Said, S. Berkovsky, and E. W. De Luca. Group recommendation in context. In *Proceedings of the 2nd Challenge on Context-Aware Movie Recommendation*, CAMRa '11, pages 2–4. ACM, 2011.
10. K. Verbert, H. Drachsler, N. Manouselis, M. Wolpers, R. Vuorikari, and E. Duval. Dataset-driven research for improving recommender systems for learning. In *Proceedings of the 1st International Conference on Learning Analytics and Knowledge*, LAK '11, pages 44–53, New York, NY, USA, 2011. ACM.
11. S. R. Wolfe and Y. Zhang. Interaction and personalization of criteria in recommender systems. In *User Modeling, Adaptation, and Personalization*, pages 183–194. Springer, 2010.
12. C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen. Improving recommendation lists through topic diversification. In *Proceedings of the 14th international conference on World Wide Web*, pages 22–32. ACM, 2005.

Using a Rich Context Model for a News Recommender System for Mobile Users

Alisa Sotsenko, Marc Jansen, Marcelo Milrad

Linnaeus University, Media Technology, Växjö, Sweden
{alisa.sotsenko, marc.jansen, marcelo.milrad} @lnu.se

Abstract. Recommender systems have become an important application domain related to the development of personalized mobile services. Thus, various recommender mechanisms have been developed for filtering and delivering relevant information to mobile users. This paper presents a rich context model to provide the relevant content of news to the current context of mobile users. The proposed rich context model allows not only providing relevant news with respect to the user's current context but, at the same time, also determines a convenient representation format of news suitable for mobile devices.

1 Introduction

Nowadays, people use mobile devices in very different situations independent of time and space in order to search and to retrieve relevant information about their needs and interests. Recommendation systems have become more and more popular for mobile devices due to the use and availability of various mobile information services [1]. Here, modern mobile devices provide a profound set of sensors and, together with Internet connectivity, rich possibilities to present relevant information with respect to the users' current context. Several studies have been conducted in order to provide personalized news recommender systems [2,3]. However, a number of challenges have been identified related to the accuracy of the news with relation to the mobile user´s context and the proper format in which this content should be delivered. Most systems have limited information about the context of the mobile users and they require explicit data input about the features of the mobile device without taking into account mobile limitations (e.g. screen size, connectivity type, battery status). Work carried out by [4] shows that recommender systems can provide better quality of news and movies recommendations if additional contextual information is taken into consideration. Therefore, one of the key challenges for providing relevant news in a convenient representation format within different mobile user's environments is to conceptualize a rich context model. From this perspective, we agreed with the research efforts carried out by [5] that claim that context models should be generic and abstracted in order to reuse it in different recommendation domains.

This contribution presents a rich context model for mobile users to be applied for news recommender systems. The paper is organized as follows. Section two describes which contextual information can be used in order to describe the rich context of

mobile users. Additionally, it describes how different content of news might or might not be relevant for mobile users in a variety of contexts, also with respect to their representation format. In Section three we describe the context model for handling the rich context information. Finally, our last section concludes the paper and describes future lines of work.

2 Defining Rich Context for Mobile Users

We understand the term *rich context* as data received from different mobile sensors and how this data can be enhanced by using external Web Services (e.g. Google Place API) in order to retrieve more detailed information including among others the current location (e.g. place, environmental information, etc.). Rich contexts may also include personal information like topics of interests, hobbies, profession, etc. The information about the user's interests and hobbies can be used to describe his/her topics of interests of news items. Additional contextual information could consist of the noise level in the place where the user is currently located and his/her movement status (e.g. sitting, walking, etc.) can be used to decide about the most convenient representation format related to the news content that best suit mobile devices. For instance, if the user is walking to his/her job place listening to an audio stream provided by a text to speech API reading out the news and listening to with headphones can be more convenient in comparison to reading on the mobile device screen. Furthermore, information about the platform of the mobile device allows providing relevant news information to the user's device. For instance, if users A and B share the same interests e.g. for mobile games, if user A has an Android based device while user B has an iPhone, the news about upcoming games for the Android platform will be more relevant to user A than user B.

We classify the current context of the user into three major dimensions: the *environment context* (e.g. place, noise level, date and time, etc.), his/her *personal context* (e.g. topics of interests, hobbies, profession, etc.) with information about the activity in which the user is currently involved (e.g. doing sport, working, etc.) and/or *device context* (e.g. information about device platform). All these dimensions of rich contextual information, as illustrated Table 1, can be extracted with help of mobile sensors and existing additional Web Services.

Table 1. Dimensions of rich contextual information.

Personal Context	Environment Context	Device Context
Topics of interests	Place	Platform
Hobbies	Direction	Battery Status
Country	Movement	Internet connectivity
Activity	Noise level	
Language	Date	
	Time	

For instance, the location can be gathered from GPS sensor and the current user's location could be identified by using Web Services like the Google Places API. Another example of using GPS sensor is getting local news related to the user nearest

place by using additional web service e.g. YourStreet API or Google News API. The information about the platform of the mobile device can be obtained by using e.g. Cordova API. Information about user's hobbies, age, language, country, profession can be collected from the different social network API's, e.g. through a social network login, in order to provide better recommendation results.

All the data sets described in Table 1 are used to represent the *rich context* of mobile users. The context model supports extendability of the sub-dimensions of the context information described in Table 1 and can therefore be used in different recommendation domains by considering different context parameters. The amount of relevant news related to one topic could be different in different contexts, e.g., if the user is sitting on a train, then he/she might want to read among a number of different news sources, while during a physical exercise (e.g. jogging) the user might want to get the news just from his/her favorite news site. An algorithmic and model based approach for handling rich context information of mobile users, the *Rich Context Model (RCM),* is described in the next section.

3 Description of the Proposed Rich Context Model

RCM is a context model for the handling of *rich context* information provided by mobile users. As described in our previous work on context modeling [6], we decided to use a *multi-dimensional vector space model* (MVSM) as the approach for modeling rich contexts of mobile users. The context in which some news are suitable for a particular situation needs to be calculated. For instance, this could be done by pre-executed evaluation where the users in different context have consumed different news. Afterwards, the users' context information of consumed news is stored in the MVSM. Then, each news item could be represented as a vector in the MVSM (e.g. News1, News2). Furthermore, each context dimension is in itself multidimensional in order to allow the description of an almost unlimited amount of dimensions in the RCM, e.g. environment context includes information about the place, noise level and user movement, etc.

In this model, we considered two requirements: the first one is to provide news content that is better suited to the user's current context and the second one is the representation format of the news that is most convenient to mobile users, again, according to his/her current situation. In order to identify the relevant content of news, the *similarity* is measured between two different vectors: the vector describing the *current rich context* of the user and other vectors describing the different available news items. The similarity between vectors can be calculated by e.g. Euclidian distance, Jaccard and cosine metrics. Based on our previous efforts [6], we consider the combination of *cosine* and *Jaccard* similarity metrics in order to match the *current rich context* of the user to the content and representational format of the news. Here, we differentiate Boolean data type of the contextual information e.g., if the user is currently moving (e.g. the user is siting in the library – 0; the user is running or walking in the park – 1) or outside/inside (e.g. the user is inside of the café, library or outside in the park, stadium, etc.). For this kind of Boolean data we propose to calculate the Jaccard similarity metrics to define similar user's environment context.

The cosine metrics, which we propose for non-Boolean data, defines how similar the current context of the user to another context in which some news was consumed. Since we need to use different similarity measures, for Boolean and non-Boolean data types, we end up have a value for the similarity that is a vector itself. Thus, the final step for the identification of the news items to recommend is to calculate the closest distance to the point of the *current rich context* of the user and all available news items.

The outcomes from the proposed *rich context model* could be used for organizing relevant data with other tools to provide some classification and sentiment analyses (e.g. clustering relevant topics or categorizing users by their interests in Tweeter [7]). Especially, it allows for a flexible definition of what kind of context dimensions should be considered. Furthermore, the proposed contextualized approach allows a real time recommendation of news. The mobile application will collect the user's current context information, analyze it and recommend relevant news accordingly in real-time. Hence, the pre-executed evaluation of news should be performed before the usage of the news recommendation application.

4 Conclusions and future work

In this paper we have presented an approach for providing news recommendations based on the current context of mobile users and the format in which the news items can be represented. The proposed rich context model supports the adaptation of relevant information delivered to users of mobile devices. Our future research will be focused on the evaluation of the proposed approach in a number of practical scenarios.

References

1. Woerndl, W., Brocco, M., Eigner, R.: Context-aware recommender systems in mobile scenarios. In: IJITWE, Volume 4. (2009) 67-85.
2. Abbar, S., Bouzeghoub, M., & Lopez, S. : Context-aware recommender systems: A service-oriented approach. In: VLDB PersDB workshop. (2009) 1-6.
3. Ilievski, I., & Roy, S. : Personalized news recommendation based on implicit feedback. In: Proceedings of the 2013 Intern. News Recommender Systems Workshop, (2013). 10-15.
4. Adomavicius, G., Sankaranarayanan, R., Sen, S., Tuzhilin, A.: Incorporating contextual information in recommender systems using a multidimensional approach. In: TOIS, Volume 23(1), (2005), 103-145.
5. Mettouris, C., Papadopoulos, G. A.: Contextual Modelling in Context-Aware Recommender Systems: a generic approach. In: WISE, (2013). 41-52.
6. Sotsenko, A., Jansen, M., & Milrad, M.: About the Contextualization of Learning Objects in Mobile Learning Settings. In: QScience Proceedings (mLearn). (2013), 67-70.
7. Hannon, J., McCarthy, K., O'Mahony, M.P., Smyth, B.: A multi-faceted user model for twitter. In: Masthoff, J., Mobasher, B., Desmarais, M.C., Nkambou, R. (eds.) UMAP. Volume 7379, (2012). 303–309.

Stories around You: Location-based Serendipitous Recommendation of News Articles

Yonata Andrelo Asikin[1] and Wolfgang Wörndl[2]

[1] BMW Group, Munich, Germany
yonata-andrelo.asikin@bmw.de
[2] Department of Informatics, Technische Universität München, Germany
woerndl@in.tum.de

Abstract. Existing studies in serendipitous recommendation mostly focus on extending the metrics of desired goals such as accuracy, novelty and serendipity with respect to the user preferences. This work aims at serendipity by exploiting the prevailing location (spatial) contexts of the recommendation. For this purpose, we propose a novel spatial context model and a number of recommendation techniques based on the model. A user study on a real news dataset shows that our approach outperforms the baseline distance-based approach and thereby improves the overall user satisfaction with the recommendation result in the absence of the user's personal information.

Key words: serendipity, location-based recommender systems

1 Introduction

Serendipity means a pleasant surprise or happy accident of discovering something good or useful while not specifically searching for it. In the research field of recommendation system, serendipity is regarded as an important objective for ensuring user satisfaction with the recommendation quality [10]. Existing approaches to recommending serendipitous contents mostly focus on extending item evaluation metrics beyond accuracy and analysing existing structure of user variables such as preferences or relations to items and other users. However, this information is not always available such as in a new system or for new user (called cold start problem) or due to the privacy concerns and the willingness of the user to provide information.

In fact, serendipity can potentially happen to a lot of people due to a certain circumstance. For instance, let *Alice* be a person who does not like country music. While walking in a village near a line of mountains with a beautiful country-side scenery, she listens to radio from her mobile device. Suddenly, the radio plays a country song and she gets really interested in the song. This can be regarded as a serendipitous experience regarding her music taste. Starting from this motivation and in order to address the above mentioned problem, we propose approaches that exploit the current context variables of the recommendation which are less sensitive regarding privacy compared to the user's personal information.

Specifically, this work focuses on location or spatial variables as context. Existing works in location-based recommendation mostly emphasize the distance between the current user location with the items' coordinates as well as the user preferences. Furthermore, the works do not consider different possible associations between an item and the tagged locations that can potentially affect or enrich the recommendation result. For instance, a news item can be associated with a city because it tells a story about a person who was born there. Therefore, we model the spatial information beyond the geographic coordinates and study the associations of the location with the news articles as a part of the prior processes. Using this spatial model and considering the prior processes enable us to build various approaches to finding serendipitous items despite the absence of user preferences. To the best of our knowledge, no previous work has studied context-based serendipitous recommendation (and in particular, location-based). In brief, the contributions of this work can be listed as follows: (1) This study presents a comprehensive spatial model for recommending news articles that goes beyond the standard geographical information; (2) We introduced location-based recommendation approaches aiming at serendipity by exploiting the spatial context; (3) We conducted a user study on a real news dataset for evaluating the approaches, in which our approach outperformed the baseline algorithm in terms of surprising and serendipity of the results.

In the remainder of this paper, Section 2 presents related studies in location-aware and serendipitous recommendation. Section 3 briefly describes our spatial model as basis for the recommendation approaches in Section 4. Section 5 discusses the evaluation of the approaches that is concluded in Section 6.

2 Related Work

This work closely relates to the research on recommendation approaches focusing on serendipity and location-aware venues and news recommendation.

Serendipitous Recommendation: The traditional collaborative filtering algorithm (like-minded-people concept) can be extended by modifying the recommendation objective or similarity metrics to introduce serendipity into the recommendation result [7]. Often with this approach, accuracy is sacrificed (significantly) for the sake of other metrics. The study conducted in [10] focuses on balancing the accuracy with other factors (novelty, diversity, and serendipity) simultaneously. Social-related variables of a user can be employed to discover surprising and useful items for the user, e.g. the interaction history [3] or social relationships and trust [5]. Other researchers also modelled and analysed the user-item relations: graph-based [9] and semantic-based [1]. None of these approaches could work without sufficient user information. Our approaches, in contrast, count on contexts to deliver serendipitous items generally for all users.

Location-aware Recommendation: Location-aware recommender systems (LARS) can be classified based on a taxonomy introduced in [4]: (non-)spatial ratings for (non-)spatial items. Following this taxonomy, a location-based news recommendation uses the schema of spatial ratings for non-spatial items or for spatial items if the news is geo-tagged. This work and other studies

generally assume that the items are already tagged with geographical coordinates, and emphasize the distance between the current user location with the items' coordinates as well as the user preferences. This is shown for both venue recommendation [6] and location-aware news recommendation [2][8].

3 Spatial Model for News Recommendation

Our *spatial model* represents the broad scope of spatial information of a location in three classes: *geographical information*, *physical character*, and *place identity*. The geographical information includes the geographic coordinate (latitude and longitude) as well as the location names. The physical character of a location or *landform* generally defines the character of scenery seen by human nature. Finally, the place identity concerns the meaning and significance of places for their inhabitants and users. A news article may contain geographical information, e.g. *location name* where the news was released and *geographic coordinates* (through *geotagging* which recognizes and resolves references to geographic locations in text documents). In our approach, physical character and place identity features will be mined from a news article. We call this feature extraction process *location inference* and the further associating process *location association*.

Let $\mathcal{C} = \{c^{(1)}, ..., c^{(m_c)}\}$ as the set of m_c global available news articles, where $c^{(i)} = (u, \mathcal{D})$ is a tuple containing creator $c^{(i)}.u$ and text features vector $c^{(i)}.\mathcal{D}$. All physical locations on the earth can be represented as a set of all point locations denoted by $\mathcal{L}_G \subset \mathbb{R}^2$, where a point location $l \in \mathcal{L}_G$ is a tuple of latitude and longitude. Alternatively, $\mathcal{L}_N = \{L^{(1)}, ..., L^{(m_l)}\}$ denote the set of m_l physical places where $L^{(j)} \subseteq \mathcal{L}_G$ (allowing a place to be either a point or a region) and consequently $\mathcal{L}_N \subseteq \mathcal{P}(\mathcal{L}_G)$. Since a location can physically belong to another location (e.g. a city belongs to a country), we define a containment relation $\text{cont}_D : \mathcal{L}_N \times D \rightarrow \{0, 1\}$ where $D \in \{\mathcal{L}_G, \mathcal{L}_N\}$. Based on this representation, the different spatial information classes can be developed by introducing a set of n_l global location features $\mathcal{F}_L = \{f^{(1)}, ..., f^{(n_l)}\}$. A location feature $f^{(k)}$ can be a place name (LN) (e.g. *Munich*, *Eiffel Tower*) or a low level feature that solely or together with other features defines the physical character (LPC) (e.g. *mountain*, *beach*), or the place identity (LPI) (e.g. *industrial*, *cultural*). Let $\mathcal{F}_{LN}, \mathcal{F}_{LPC}, \mathcal{F}_{LPI} \subset \mathcal{F}_L$ be the sets of features for the particular representation of LN, LPC, and LPI, respectively. The location features are gained through the geographical mapping functions $\psi_D : \mathcal{L}_N \rightarrow \mathcal{P}(D)$, where $D \in \{\mathcal{F}_{LN}, \mathcal{F}_{LPC}, \mathcal{F}_{LPI}\}$.

Through *location inference*, spatial information is extracted from a news article. During *geotagging*, words or phrases that can be place names (called *toponym*) are firstly found in the article (this searching step is called *toponym recognition*). Afterwards, each toponym will be assigned to the right geographic coordinate (called *toponym resolution*). Formally, location inference is used to extract a set of features in $\mathcal{F}_L$ from $\mathcal{C}$. For LPC and LPI, the inference functions are denoted as $\text{inf}_{LPC} : \mathcal{C} \rightarrow \mathcal{P}(\mathcal{F}_{LPC})$ and $\text{inf}_{LPI} : \mathcal{C} \rightarrow \mathcal{P}(\mathcal{F}_{LPI})$, respectively. Since each feature $f^{(k)} \in \mathcal{F}_{LN}$ (a toponym) still has to be disambiguated to an exact $L \in \mathcal{L}_N$, the location inference is defined differently for

LN. The location inference function for LN is defined as the composition of the toponym recognition and toponym resolution functions: $\inf_{LN} = \inf_{rec} \circ \inf_{res}$ where $\inf_{rec} : \mathcal{C} \rightarrow \mathcal{P}(\mathcal{F}_{LN})$ and $\inf_{res} : \mathcal{P}(\mathcal{F}_{LN}) \rightarrow \mathcal{P}(\mathcal{L}_N)$.

People can draw a myriad of associations between news and locations. For instance, a news article can tell the history of a place and therefore, an association called *telling history* is built between the article and the place. The news articles combined with the respectively inferred locations form a set of *localized* recommendable items $\mathcal{X} = \{X^{(1)}, ..., X^{(m_x)}\}$ where $m_x \leq m_c$ is the total number of items. The tuple in $c^{(i)}$ is extended for $X^{(i)}$ resulting in $X^{(i)} = (u, \mathcal{D}, F_L, L_N)$ where $F_L \subset \mathcal{F}_L$ and $L_N \subset \mathcal{L}_N$ are the inferred location features and geographic coordinates, respectively. The associations between an item and the inferred locations can be built by means of a function $\text{association}_I : \mathcal{X} \times \mathcal{P}(\mathcal{L}_N) \rightarrow \mathcal{P}(\mathcal{A}_i)$ where $\mathcal{A}_I$ is the global set of possible associations between X and L.

4 Algorithms for Serendipitous Recommendation

For the sake of completeness, we defined a set of m_u users (either the consumer or creator of an item) as $\mathcal{U} = \{U^{(1)}, ..., U^{(m_u)}\}$. The items with inferred and associated locations together with user and location information provide building blocks for the context-aware news recommendation schema: $R : \mathcal{U} \times \mathcal{X} \times \mathcal{L}_N \rightarrow \mathbb{R}$. Given a current location L of a user u, a recommender approach suggests an item X based on L by exploiting the spatial information contained in both X and L. A baseline approach can simply be based on the distance between both of them, e.g. news near you (analogously to places near you). This method, called **Nearest Distance (ND)**, suggests a single item $X^{(i)}$ that contains $L^{(j)} \in X^{(i)}.L_N$ with smallest distance to L. To show how different utilizations of spatial model can affect the recommendation quality and in particular achieve serendipity, we propose a number of approaches below.

Geographical Hierarchy (GH) uses geographical hierarchy information of a location L and considers its parent-locations. Formally, GH looks for items with an inferred location $L^{(i)}$ where $\text{cont}_{L_N}(L^{(i)}, L) = 1$ and picks one of them randomly. Low serendipity is expected to be seen from the recommended items, since the news articles picked by this approach can be very general and well-known in a larger area of the location.

Event Association (EA) suggests the next located item from L with the association *describing event at location* with L. In this study, we define a set of associations $\mathcal{A}_I = \{\text{describing location, describing event at location}\}$. The associations are defined in this work simply by classifying based on the existence of certain keywords. Formally, we assume that if an item $X^{(i)}$ with inferred location L belongs to the class *describing location*, then the $\text{association}_i(X^{(i)}, L) = \{\text{describing location}\}$. By picking a news with a less-typical association, this approach may retrieve a more serendipitous item.

Place Identity (PI) and Combination (ND+PI): this method suggests an item with a topic that is not usual at that particular location (based on the place identity). Given current location L, the place identity is defined as $\psi_{\mathcal{F}_{LPI}}(L)$. Here, the place identity is defined as a set of topics that are often discussed at

L. Therefore, the approach will retrieve items whose topics have low similarity to the place identity, i.e. news that are not usual at L. By introducing this diversity, the serendipity is expected to be induced by this approach. Since there can also be multiple retrieved items, we can pick one item randomly (PI) or pick the nearest one (ND+PI).

5 Evaluation: Stories around You

To show how the approaches recommend serendipitous items, an online user study based on real crowd-sourced news dataset was performed. The dataset originates from an online crowd-sourced idea finding portal Jaring-Ide[1]). Specifically, it consists in a set of text articles which are ideas generated for an idea contest called *My Indonesian Moment* which is a contest about a (tourism) moment that someone experienced in a location in Indonesia. After filtering out inappropriate ideas (e.g. no text content), we get $m_c = 1869$ from 1914 ideas.

The dataset is not tagged with any spatial information and therefore, location inference (and association) are necessary. However due to the nature of the data (mixed languages, informal writing, etc.), automatic toponym recognition technique did not perform well. Therefore, we compiled a set of sub-strings of the texts that represent the correct location context of the articles. This resulted in 5293 toponyms that still have to be resolved. For the toponym resolution on $c^{(i)}$, we use gazetteer from GeoNames[2]. The inference $\inf_{LN}$ resolved the total of 4297 toponyms that corresponds to 1818 resolved items (97.27% of all available items). This forms a set of recommendable items $\mathcal{X}$ with $m_x = 1818$. Since no ground truth for disambiguated (resolved) locations (with latitude and longitude coordinates) is available, we have to relate the performance of this technique with the appropriateness evaluation of the recommendations.

5.1 Model of Place Identity

For modelling the place identity used by PI and ND+PI, we first performed items clustering based on the inferred locations of the items with *leader-follower* method (distance threshold = 200 km). This results in 57 clusters over all items with maximal distance of a cluster member to centroid is about 230 km. Next, for each cluster, we want to define the common topics in that cluster. For this purpose, we created a vector over all terms in the whole dataset for each item using TF-IDF (*term frequency-inverse document frequency*). We defined the central topics in a cluster by computing the mean centroid of the term vectors in each cluster. Next, the similarity of each cluster item with the centroid is computed with the *cosine similarity*. Table 1 shows an example of cluster resulting from the approach described above. The cluster consists of 53 items and the average of similarity computations to the centroid is 0.341. To recommend an item in a given location L, PI first looks for the nearest cluster with the smallest distance between its centroid and L. Next, the average of item similarity with the centroid (the place identity) is computed and an item with a lower similarity than

[1] http://www.jaring-ide.com/
[2] http://www.geonames.org/

Table 1: The topic extraction of a cluster showing 4 (of 53) example members.

Centroid topics ($avg = 0.341$)**:** Aceh, tsunami, fish*, fisherman*, beach*		
Items	**Sim**	**Topics**
Above *avg*	0.665	Aceh, tsunami, Province*, island*, hit*
Above *avg*	0.615	Aceh, fishing*, fish*, sun*, region*
Below *avg*	0.329	dance performance*, colonialism*, Dance*, Aceh, allowed*
Below *avg*	0.089	art*, element*, festive*, epoch*, Dance*

*word translated from Bahasa to ease the observation

the average similarity (hence, not similar to the usual topics) is picked (items labelled as *Below avg* in Table 1).

5.2 User Study

We performed a user study using a web application that shows suggested stories (news articles) based on a current location. The assumed current location is generated randomly from a set of about 300 regencies and cities in Indonesia. In every recommendation session, four stories are suggested by four approaches: ND (as a baseline algorithm), GH, EA, and PI. In every recommendation session (on a web page), the order of these stories is shuffled and hence the user can not find out which item is recommended by which technique. For each suggested story the user is asked to submit evaluations in three categories: ***appropriate, like, surprising***. The category *appropriate* is the measure of how suitable the story is with the given location (since the toponym resolution was performed without ground truth as in a real-life application). Next, user can assess the quality of the story in the category *like*. Finally, the category *surprising* defines the metric of how unusual the topic of the story in the area of the given location is. The evaluation is submitted in form of a 5-scale rating (from disagree to agree).

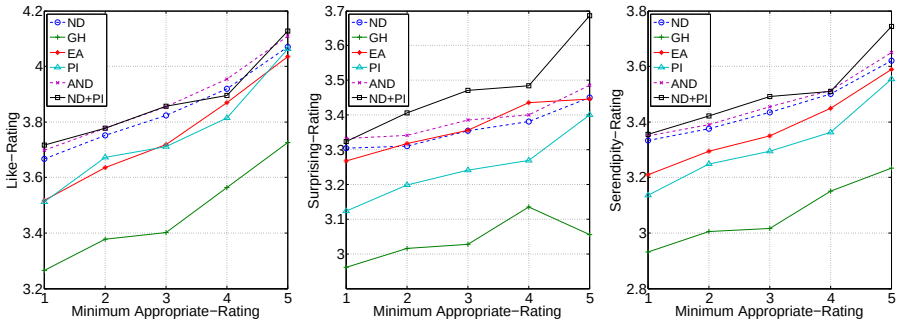
In this user study, 44 users with general knowledge about locations in Indonesia were asked to assess the recommended articles in each given location (165 locations were randomly given across the experiment). The result comprises 827 ratings distributed over stories that were recommended by the approaches (ND: 207, GH: 204, EA: 205, PI: 211) on 232 recommendation pages (which means that some pages did not receive complete ratings for all 4 stories). In addition to the online elicited ratings, we defined *serendipity-rating* as **serendipity = (like + surprising) / 2** (since serendipity involves unexpected (surprising) but pleasant (liked) aspects). We also run offline recommendation on the already rated stories with **ND+PI** and **AND** (Absolute Nearest Distance) as another baseline. This is done because: (1) GH, EA, and PI do not have real objective functions (partially random); (2) not all stories were rated completely on every recommendation page. For every page with missing ratings for ND, AND recommends other rated items with the nearest distance.

The summary of the evaluation results is partly presented in Table 2. The table presents the average of ratings for each technique and each rating category with appropriate-rating = 5 (assumed to be recommended appropriately). The result from ND can for instance be regarded as the parameter for the overall

Table 2: Results of ratings in the experiment *Stories around You*.

	ND	**GH**	**EA**	**PI**	**AND**	**ND+PI**
#appropriate $\geq$ 5	100	73	83	65	111	86
like-rating	4.070	3.726	4.036	4.062	4.108	**4.128**
new-rating	3.450	3.055	3.446	3.400	3.486	**3.686**
serendipity-rating	3.620	3.233	3.590	3.554	3.649	**3.744**

appropriateness of the recommendation: 160 out of 232 items (about 68.9%) were evaluated with rating ≥ 4. Aside from the fact that the inference may have been wrong at the first place, there may be 3 other causes for an inappropriate recommendation: (1) not enough news articles to recommend at the location; (2) a nearest item is from another adjacent regency or even another adjacent province (since no shared-parent check); (3) the participants think the location is not central to the story even though it is inferred correctly.



(a) Like (b) Surprising (c) Serendipity

Fig. 1: Ratings based on the appropriateness range

The overall comparison and the development of the like-, surprising- and serendipity-ratings of the approaches along the ranges of appropriate-rating is illustrated in Figure 1. As can be seen in this figure, our approach ND+PI can perform as well as both of the baseline approaches ND and AND in term of the like-rating (Figure 1a). In terms of both surprising- and serendipity-rating (Figure 1b and 1c), the approach outperforms the baseline approaches in almost all value ranges of appropriate-ratings. PI and EA, in contrast, did not perform well in both surprising- and serendipity-rating as expected originally. We argue that this is caused by the random nature of these approaches as well as the availability of the data (e.g. not enough data with the desired association near the location). Another important insight is to see how the items recommended by GH were seen as less-favoured (even with appropriate-rating = 5), and expectedly less-surprising for the users since the recommended news articles would be more general. This shows the effectiveness of our location inference approach to assign the locations to the correct geographical hierarchy level.

6 Conclusion

We presented approaches for recommending news article by using spatial variables as the main factor of relevance. The aim of these approaches is to deliver serendipitous recommendation and improve the user satisfaction in absence of user preferences. A user study showed that the approaches can find items that are in general more serendipitous (surprising but still favoured) than the ones retrieved by the baseline (distance-based) algorithm. This study can motivate further investigations of context-based serendipitous recommendation by using more complex spatial model (e.g. based on LDA instead of TF-IDF) and location associations, as well as the integration of user preferences where applicable.

References

1. N. Auray. Folksonomy: The new way to serendipity. *Communications & Strategies, No. 65, 2007*, 2007.
2. J. Bao, M. Mokbel, and C.-Y. Chow. Geofeed: A location aware news feed system. In *IEEE 28th International Conference on Data Engineering (ICDE)*, pages 54–65, 2012.
3. Y.-S. Chiu, K.-H. Lin, and J.-S. Chen. A social network-based serendipity recommender system. In *Intelligent Signal Processing and Communications Systems (ISPACS), 2011 International Symposium on*, pages 1 –5, dec. 2011.
4. J. J. Levandoski, M. Sarwat, A. Eldawy, and M. F. Mokbel. Lars: A location-aware recommender system. In *Proceedings of the 2012 IEEE 28th International Conference on Data Engineering*, ICDE '12, pages 450–461, Washington, DC, USA, 2012. IEEE Computer Society.
5. D. Mican, L. Mocean, and N. Tomai. Building a social recommender system by harvesting social relationships and trust scores between users. In W. Abramowicz, J. Domingue, and K. Wecel, editors, *Business Information Systems Workshops*, Lecture Notes in Business Information Processing, pages 1–12. Springer Berlin Heidelberg, 2012.
6. A. Noulas, S. Scellato, N. Lathia, and C. Mascolo. A random walk around the city: New venue recommendation in location-based social networks. In *Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom)*, pages 144–153, 2012.
7. M. Schedl, D. Hauger, and D. Schnitzer. A model for serendipitous music retrieval. In *Proceedings of the 2nd Workshop on Context-awareness in Retrieval and Recommendation*, CaRR '12, pages 10–13, New York, NY, USA, 2012. ACM.
8. W. Xu, C.-Y. Chow, M. L. Yiu, Q. Li, and C. K. Poon. Mobifeed: a location-aware news feed system for mobile users. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems*, SIGSPATIAL '12, pages 538–541, New York, NY, USA, 2012. ACM.
9. H. Yin, B. Cui, J. Li, J. Yao, and C. Chen. Challenging the long tail recommendation. *PVLDB*, 5(9):896–907, 2012.
10. Y. C. Zhang, D. O. Séaghdha, D. Quercia, and T. Jambor. Auralist: introducing serendipity into music recommendation. In *Proceedings of the fifth ACM international conference on Web search and data mining*, WSDM '12, pages 13–22, New York, NY, USA, 2012. ACM.

Method for Novelty Recommendation Using Topic Modelling

Matúš Tomlein and Jozef Tvarožek

Institute of Informatics and Software Engineering, Faculty of Informatics and Information Technologies, Slovak University of Technology, Ilkovičova 3, 842 16 Bratislava 4, Slovakia

Abstract. Content-based filtering methods fall short in situations where there are many similar items to recommend from, for instance when recommending articles from multiple news portals. To deal with this problem, we can consider the novelty of recommendations. Detecting novelty is usually implemented as finding the most dissimilar articles. We propose a method that uses topic modelling to find the novelty of articles. Our method ranks topics by their importance and novelty to the user and recommends articles according to their topics. We evaluate our method and compare it to other approaches to novelty recommendation and also to a method that doesn't take novelty into account. The results show that our method was more successful than the other approaches to novelty detection in recommending relevant articles that the users were interested in. It also showed a better click-through rate than the method that didn't incorporate novelty, although the order of its recommendations was less optimal.

Keywords: news, novelty, recommendation, topic model

1 Introduction

Redundant articles that cover similar information but present them in a different way are common on the Web. Since there are numerous news portals covering a relatively small number of events, such a situation is inevitable.

Content-based recommender systems, or adaptive information filtering systems, are mostly designed to recommend articles based on their similarity or relevancy to what the users previously read [9]. While this might not be an issue if the articles are recommended from a single source, recommending from multiple news portals based solely on the relevancy of articles can overwhelm the users with redundant information.

To deal with this problem, we have taken the novelty of individual articles into account. Novelty is defined with respect to the end-user as the proportion of known and unknown information [8]. Our goal is to maximize the novelty of the recommendations to the user while keeping them relevant to their interests.

There are various approaches to novelty detection. Many of them treat novelty as a measure of similarity. They look for articles that are least similar to the

ones the user previously read [4]. This is often not an accurate representation of novelty. In our work, we propose a method that detects the novelty of articles using topic modelling. We calculate the novelty of articles based on the novelty of their topics.

We evaluate our method in two experiments. First we compare it to other common approaches to novelty detection in an offline experiment. Then we apply it along with a method for content-based recommendation and another method for novelty recommendation in online recommendation and evaluate the results.

2 Related work

Three TREC Novelty track workshops focused on novelty detection. In each workshop, a manually created data set was used that contained sentences rated by their novelty and relevancy [6].

There were also attempts to create news recommender systems that applied novelty detection methods to provide an interface for users to find articles with novel information [2, 1]. They applied various difference metrics for novelty detection, like inverse cosine similarity, Kullback-Leibler divergence, density of previously unseen named entities, quantifiers and quotes.

The use of topic models in novelty detection mainly focused on application in research articles. It showed promising results in comparison to other approaches [4]. It also recognized the importance of ranking the significance of topics using weighted topic coverage [7].

Novelty can also be approached using collaborative filtering [8]. Instead of looking for the least similar articles, we can look for the least popular items. Novelty can also be introduced by considering the recommendations of dissimilar users in addition to similar users. However, in this paper we will focus on content-based approaches.

3 Method for novelty recommendation based on topic modeling

Our goal is to design and evaluate a method for news article recommendation that recommends articles based on their novelty to the reader. It is important to ensure that the recommended articles are relevant to the interests of the users, i.e. to what they previously read about. To achieve this, we perform the recommendation in two steps:

1. Create a cluster of similar articles
2. Recommend novel articles within the cluster

To create the cluster of similar articles, we use the Carrot[2] for the Elasticsearch server. We create a search query using the title from the last read article and from the clusters of results, we use the one that contains the article.

The overview of the method is shown in Figure 1.

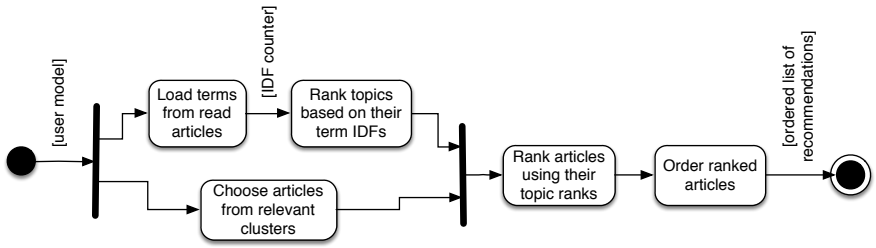


Fig. 1. Overview of the method. It first chooses relevant articles and ranks the topics using the user model. Then it ranks the relevant articles based on their topics and orders them by their rank.

Topic modeling Our method uses topic modeling in order to calculate the novelty and relevancy of articles. Topics are sets of relevant words with a probabilistic degree of distribution with them [4]. We use the Latent Dirichlet Allocation algorithm for topic modeling. The reason why we think topic modeling can be useful in novelty recommendation is that it provides a way to work with the information in articles on a higher level of abstraction. It allows us to work with information using topics as opposed to using keywords.

Our hypothesis is that topic modeling is a better approach to detecting relevant novel information than using an inverse similarity or divergence measure.

User model The main purpose of our user model is to store information about the articles the user read. It contains the following information:

- List of read articles
- List of topics of the read articles along with their probabilities retrieved from the topic model

Topic ranking Topics retrieved from LDA have various qualities. While many represent a coherent group of connected terms, frequently we find topics without any significant value. These less important topics can have an impact on the performance of our method and so it is useful to give them a lesser importance when considering their contribution. To address the novelty of topics, we want to give a lesser importance to topics that group information the user already read about. To meet this goal, we employ topic ranking. We give each topic a numeric rank that represents its importance and novelty to the user. In contrast with weighted topic coverage used in [7], we rank topics according to their terms, not their presence in the topic model and calculate their rank using the users reading history.

We use an algorithm inspired by the method proposed in [3] that calculates the novelty of an article based on the Inverse Document Frequency (IDF) of its

terms. We use the average IDF of the 100 best terms of a topic to calculate its rank. This number should be set according to the properties of the topic model, it should be lower if there are many topics covering a smaller number of events and larger if there are less topics covering more events. We found that in our topic model the first 100 terms were usually consistent within topics. We calculate the IDF against the corpus of articles the user read. The rank of a topic is calculated using the Formula 1, where T is the collection of terms and their probabilities in the topic, t is a term, w is the weight of the term and *idf* is the function for computing the IDF of a term.

$$TR(t) = 1 - \frac{\sum_{t,w \in t} idf(t) * w}{|T|} \tag{1}$$

By using the read articles as the corpus for calculating IDF, we both ensure that a lesser importance is given to topics containing terms that are frequent in other articles and that a higher rank is given to topics containing novel terms that the user didn't read about.

The novelty rank of an article is calculated using the Formula 2, where the function *topics* returns a list of topics of the article with their probabilities from the topic model.

$$AR(a) = \frac{\sum_{t,w \in topics(a)} TR(t) * w}{\sum_{t,w \in topics(a)} w} \tag{2}$$

4 Evaluation

A common and effective way to evaluate a novelty detection method is to use a preprocessed data set of article and sentence novelty comparisons created by users [6]. We took this approach to compare our method with common approaches to novelty detection offline.

We also wanted to evaluate our method in online recommendation to see what real users think about its recommendations. We compared it to a method for content-based recommendation used in production systems and another method for novelty recommendation.

4.1 Offline evaluation

The goal of this study was to find out the advantages and disadvantages of our method compared to different approaches to novelty detection. We collected explicit comparisons of articles and using the comparisons, we evaluated the following methods offline (for each method, a short explanation is given on how the novelty of an article is calculated):

- *Inverse similarity* — average of the minimum inverse cosine similarity of each sentence in the article compared to the sentences in the read articles [4]

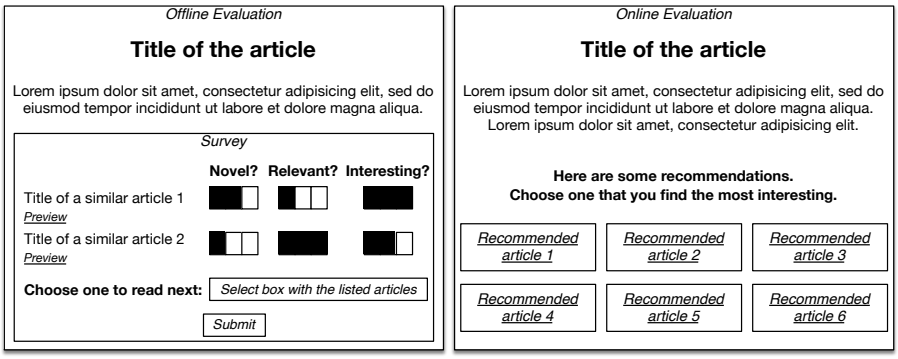


Fig. 2. Wireframes of the user interface used in the offline evaluation on the left and the online evaluation on the right. In the offline evaluation, the task was to rate the novelty, relevancy and interestingness of several articles compared to the one presented above on a given scale. The participants were also asked to choose one of the listed articles that they would most like to read next. In the online evaluation, the task was to choose one of the recommended articles that the participant found the most interesting.

- *IDF based novelty* — average IDF of the terms of the article, terms from the read articles are used as the corpus to calculate the IDF against [3]
- *Topics JS* — Jensen-Shannon divergence of the topic distribution of the article compared to the topic distribution of the read articles [4]
- *Ranked topics* — our method described in section 3

5 subjects (university students) took part in assessing the data. They compared 152 pairs of articles. The articles being compared were retrieved from 11 well-known tech blogs.

The user interface for comparing artcles is shown in Figure 2. It showed an article at the top and a feedback form at the bottom. The form consisted of 4–10 other articles that were related to the article above. The task of the participants was to compare the listed articles to the one above based on their novelty, relevancy and how interesting they were, on a scale of 3. We also asked them to choose one article that they would like to read next.

Results The study showed that the perception of what is novel information and what is not is very subjective. The participants used different scales for rating the novelty and relevancy of the articles, some of them rarely using the option *"A lot of new information"*. We also received feedback that the rating of interestingness was unclear as it could have been influenced by various factors.

The evaluation of the first part of the study went as follows. If the user rated article A as more novel (relevant, interesting) than article B, we tested if a given method also ranked article A higher than article B. To evaluate the choice of one

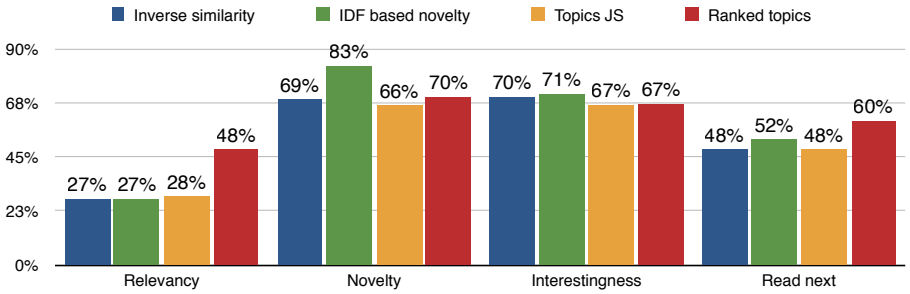


Fig. 3. Results from the offline evaluation of methods for novelty recommendation based on the data collected in our experiment.

article that the user picked to read next, we considered an algorithm successful if it listed the chosen article among the first 3 recommendations.

The results are shown in Figure 3. As the chart shows, our method ranked by far the highest in the relevancy of its recommendations. This means that it recommended articles that were relevant to the ones the participants read. It was also the most successful in recommending articles that the users chose to read next. We think that these are useful properties that other methods for novelty recommendation lack.

The *IDF based novelty* scored the highest in novelty, which means that it recommended articles containing the most novel information compared to the read article. However, the recommendations were less relevant to the read article, which is also the case for *Inverse similarity* and *Topics JS*.

The methods *Inverse similarity* and *Topics JS*, which both look for the most dissimilar articles, showed similar results. It is interesting that although *Topics JS* makes use of a topic model, it didn't make a significant difference. Our method, also based on topic modelling, showed better results than *Topics JS*, which might be thanks to ranking topics by their importance and novelty.

4.2 Online evaluation

We implemented a news reading portal — a website showing an article and 6 recommendations below it. The goal of the experiment was to compare our method with a method for content-based recommendation used in production systems and a method for novelty recommendation using online recommendation to users. We used the following methods to recommend articles:

- *MoreLikeThis* — constructs a search query from the top TF-IDF ranked terms from the article and executes it on the Elasticsearch search server
- *IDF based novelty* — creates a cluster of similar articles using Carrot[2] and orders them using *IDF based novelty* explained in the offline evaluation
- *Ranked topics* — our method described in Section 3

Two recommendations were chosen from each method. In case two methods recommended the same article, the next best article was used from one of them.

The user interface of the experiment is shown in Figure 2. It shows an article to be read at the top and 6 recommendations below it. The recommendations are presented in random order. When the user clicked on a recommended article, it was opened. The task of the participants of the experiment was to read the main article and choose one recommendation that they found the most interesting.

Results The experiment was carried out at a workshop of the PeWe research group at the Faculty of Informatics and Information Technologies STU. 23 students and graduates from the faculty took part in it. They read 310 articles. Each student read 13.5 articles on average with a standard deviation of 6.

We calculated the click-through rate of each method as the number of clicks on its recommendations divided by the number of their impressions ($CTR = \frac{clicks}{impressions}$). In Figure 4, we show the CTR of clicks on all articles and also clicks on articles that the users read longer than 15 seconds. In both cases, our method was the most successful. The score for *MoreLikeThis* shows that it is more successful when the reading time is not taken into account. It means that the participants often left the articles recommended by *MoreLikeThis* soon after opening them, possibly because they didn't contain enough novel information.

Based on the CTR results and using Bayesian inference, we calculated the approximate probability of the tested methods of being the best, with the following results: *MoreLikeThis*: 2%, *IDF based novelty*: 5%, *Ranked topics*: 93%.

We also calculated the R-score, which is a utility-based ranking metric that rates the order of a list of recommendations. It assumes that the value of recommendations declines exponentially down the ranked list (explained in [5]). For each user, we recreated the top 10 recommendations for each article that they read and removed the ones that were never recommended to them. We show the results in Figure 4 for different levels of the parameter α, which controls the exponential decline of the value of positions in the list [5]. Based on the results, *MoreLikeThis* had the most optimal ordering of its recommendations. Its score decreases with higher α, that is when the exponential decline is less steep.

In both cases, our method was more successful than *IDF based novelty*, which is probably thanks to having better relevancy as found in the offline evaluation. This also shows that even though our method was less successful in the novelty rating in the offline recommendation, this property is less crucial in real recommendation.

5 Conclusions

We proposed a method for recommending articles based on their novelty. It uses topic modeling and ranks topics by their novelty to the user based on the IDF of the topic terms.

We evaluated our method in two experiments in which we compared it to other novelty based methods and a method that didn't take novelty into acc-

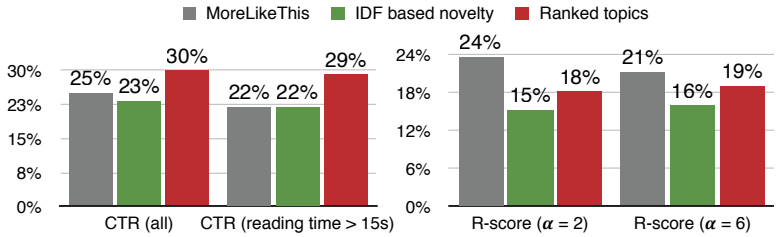


Fig. 4. CTR and R-score calculated based on the results from the online experiment.

count. We found that our method was the most successful out of the novelty based methods in recommending relevant articles that the users were interested in. It also received a higher click-through rate than the method that didn't incorporate novelty, although its ordering of the recommendations was less optimal.

We found that using topic modelling as the basis for novelty detection is a valid approach that is applicable in recommendation, particularly if the importance of individual topics is taken into account. We also think that recommendations based on novelty should be combined with recommendations that don't incorporate novelty so the users can choose to explore both similar and novel articles based on their preferences.

Acknowledgements. This work was partially supported by the Scientific Grant Agency of the Slovak Republic, grant No. VG1/0675/11 and by the Slovak Research and Development Agency under the contract No. APVV-0208-10.

References

1. Gabrilovich, E., Dumais, S., Horvitz, E.: Newsjunkie: Providing personalized newsfeeds via analysis of information novelty. In: In WWW2004. pp. 482–490 (2004)
2. Iacobelli, F., Birnbaum, L., Hammond, K.J.: Tell me more, not just more of the same. Proceedings of the 15th intl. conference on Intelligent user interfaces (2010)
3. Karkali, M., Rousseau, F., Ntoulas, A.: Efficient Online Novelty Detection in News Streams (2013)
4. Sendhilkumar, S., Nandhini, N., Mahalakshmi, G.: Novelty detection via topic modeling in research articles. airccj.org pp. 401–410 (2013)
5. Shani, G., Gunawardana, A.: Evaluating recommendation systems. In: Recommender Systems Handbook, pp. 257–297. Springer US (2011)
6. Soboroff, I., Harman, D.: Novelty Detection: The TREC Experience. In: Proceedings of the Conf. on Human Language Technology and Empirical Methods in NLP (2005)
7. Xiao, Z., Che, F., Miao, E., Lu, M.: Increasing Serendipity of Recommender System with Ranking Topic Model. Applied Math. & Information Sciences 8(4) (2014)
8. Zhang, L.: The Definition of Novelty in Recommendation System 6(3) (2013)
9. Zhang, Y., Callan, J., Minka, T.: Novelty and redundancy detection in adaptive filtering. Proceedings of the 25th intl. ACM SIGIR conference p. 81 (2002)

Building Rich User Profiles for Personalized News Recommendation

Youssef Meguebli[1], Mouna Kacimi[2], Bich-liên Doan[1], and Fabrice Popineau[1]

[1] SUPELEC Systems Sciences (E3S), Gif sur Yvette, France,
{youssef.meguebli,bich-lien.doan,fabrice.popineau}@supelec.fr
[2] Free University of Bozen-Bolzano, Italy,
mouna.kacimi@unibz.it

Abstract. Nowadays, more and more people are using online news platforms as their main source of information about daily life events. Users of such platforms have access to an increasing amount of articles of different topics, stories, and view points. Thus, a news personalization service is needed to filter the flow of available information and satisfy users needs. To this end, it is crucial to understand and build accurate profiles for both users and news articles. In this paper, we propose a new approach that exploits users comments to recommend articles. We build the profile of each user based on (1) the set of entities he talked about it in his comments, (2) and the set of aspects related to those entities. The same information is extracted from the content of each news article to create its profile. These profiles are then matched for the purpose of recommendation. We have used a collection based on real users activities in four news web sites, namely The Independent, The Telegraph, CNN and AL-Jazeera. The first results show that our approach outperforms baseline approaches achieving high accuracy.

Keywords: User modeling, Personalization, News recommendation

1 Introduction

Media platforms, like CNN [1] and Al-Jazeera [2], deliver the latest breaking news on various topics about everyday events. The rich content of such platforms and their easy access make them a leading information source for Internet users. Typically, besides reading news articles, media platforms offer the possibility for users to write their comments, express their opinions, and engage in discussions with other users. However, before reacting to any content, users need first to find news articles of interest. This task can be challenging since, in many cases, a user may not even know what to look for. Consequently, there is a need for personalized news services that recommend articles based on user profile. The accuracy of personalized recommendation depends mainly on how well user profiles are defined. Naturally, users' comments represent a valuable

[1] http://www.cnn.com
[2] http://www.aljazeera.com/

information source since they reflect not only interesting entities for users but also more details about which entity aspects they are interested on. Therefore, several past studies have exploited, in different ways, users' comments for news recommendation [1–5,7–9,11]. Most of the approaches use tweets [2–4,7,11] and few others [1,5,9] exploit users' comments on news websites. hmueli et. al., [9] restrict user profile to a set of tags extracted from related comments. Abbar et. al., [1] build the profile of each user based on the set of entities he has commented on with their related sentiments. While the proposed approach is interesting, it does not exploit all available information in users' comments and thus it provides incomplete profiles. The reason is that a user can be interested to a specific entity when it is related to a given aspect and can be not interested on it when it concerns another aspect. For instance, we can have a user who is interested by the entity *Tunisia* when it is related to the aspect *Tourism* and be not interested when it is related to the aspect *Election*. In this paper, we propose a personalized news recommendation approach that pays particular attention to interesting aspects for each entity. To this end, we introduce a new approach that models the profile of users and articles based on a set of tuples representing entities and their aspects. The idea is to have a fine-grained description of users and articles regarding general topics together with more specific issues. The profile of a user is extracted from the set of comments he provides in the news platform, and the article profile is extracted from its content and described by a set of tuples $(entity, aspect)$. We define each profile by two main components: (1) *entities* which reflect well defined concepts such as persons, locations, organizations, objects, etc., and (2) their related *aspects* representing entity attributes or any abstract object. These profiles are then matched to recommend to each user the list of articles that match with user profile interests and the current article he is reading. We evaluate our approach using four real datasets including, The Independent [3], The Telegraph [4], CNN [5] and Al-Jazeera [6]. The experiments show that our approach outperforms baseline approaches with a large margin, in term of precision and NDCG.

2 Related Work

Exploiting user generated content in social networks to define users' interests have been extensively studied [2,4,7,10,11]. Stoyanovich et. al., [10] leverage the tagging behavior of users to derive implicit social ties which were shown to serve as good indicator of user's interests. Chen et. al., [3] exploits user Tweets to build a bag-of-words profile for each Twitter user. Abel et al., [2] build hashtag-based, entity-based, and topic-based user profiles from Tweets, and show that semantic enrichment improves the variety and the quality of profiles. Other approaches [4, 7] address the problem of extracting topics of interest in micro-

[3] http://www.independent.co.uk/
[4] http://www.telegraph.co.uk/
[5] http://www.cnn.com
[6] http://www.aljazeera.com/

blogging environments. Hong et.al., [4] train a topic model on aggregated messages to improve the quality of topic detection in Tweets. Michelson et. al., [7] use a knowledge base to disambiguate and categorize the entities in user Tweets and then develop users profiles based on frequent entity categories. Our work does not fall in the previous classes since we exploit richer and longer comments than Tweets. Thus, we relate our work to the second class of approaches [1, 5, 9] which exploit users' comments on news websites to build user profiles. Li et. al., [5] enrich the content of each news article using users' comments and use the enhanced content to improve the accuracy of recommendation. However they do not build any user profile which results in a limited accuracy. Shmueli et. al., [9] restrict user profile to a set of tags extracted from related comments using a bag-of-words model. The closest work to ours is by Abbar et. al., [1] who build the profile of each user by extracting the set of entities he has commented on and their related sentiments. While the proposed approach is interesting, it does not exploit the different aspects of entities to have a more precise profile. In our work, we model user profile as set of interests reflected by the conjunction of *entities* and *aspects*. Another line of research related to this work is recommender systems [1–3, 5, 8, 9]. Two main strategies of recommender systems have been adopted and mostly combined in previous works. First, content filtering strategy creates a profile for each user or seed article and then recommends the best matching articles based on the user profile, the seed article, or both. Second, collaborative filtering strategy relies only on past user behavior without requiring the creation of explicit profiles. In our work, we adopt a content filtering strategy to recommend news articles to users based on their profile and potentially also on the article they are currently reading.

3 Personalized News Recommendation

3.1 Problem Definition

Our goal is to propose a personalized news recommendation model tailored to users' interests. Typically, interests represent the conjunction between entities and their related aspects. Entities reflect well defined concepts such as persons, location, organizations or objects, for example *"Aalborg"*, *"UMAP"*, and *"United Nation"*. While aspects reflect some specific issues related to the list of entities such as *"illegal immigration"*, *"recommender systems"*, or *"humanity acts"*. In our setting, we identify the interests of a given user based on the comments he has posted on the news platform. Using this information, the personalized news recommendation works as follows: Given a target user who is reading a seed article, we recommend a set of news articles that (1) are similar to the seed topic article for not deviating far away from user's interests and (2) match with specific issues that interest the user profile. The idea behind is to select, first, new articles that belong to the same topic than the seed article and then choose a subset that match with user interests. Formally, we define U as the set of users of a given news platform, and A as the set of articles provided by the news platform. Each user $u_i \in U$ provides a set of comments C_i about a set of

articles A' where $A' \subset A$. We assign to each user u_i a profile P_{u_i}, extracted from the set of his comments C_i, which reflects his specific issues about what he reads in the past. Similarly, we assign to each article a_j a profile P_{a_j} extracted from its content. When user u_i is reading article a_j , we proceed as follows. First, we compute the similarity between the article profile P_{a_j} and the profiles of the set of articles A_t where $A_t \subset A$ and A_t corresponds to all the articles that were published in time interval t. In this way, we can restrict our search space to any time period specified by the user. The time interval can range from a few days to months depending on user needs. The set of articles A_t is then sorted from the most similar article to a_i to the least similar one resulting in list L_1. Second, we compute the similarity between the user profile P_{u_i} and the profiles of the articles contained in the set A_t, thus, providing another sorted list L_2 from the most similar article to user profile P_{u_i} to the least similar one. As a last step, we aggregate the two lists L_1 and L_2 to obtain the final list of sorted articles from which we recommend the topk articles to user u_i.

3.2 Modeling User and Article Profiles

Due to the fact that both user comments and articles can express different types of information, including objective and subjective ones, we model both contents in the same way using the same structure for their profiles. To this end, we start by transforming the content of each comment (article) to a set of sentences S , using OpenNLP[7]. From each sentence, we extract two main components. First, a set of *entities*, where entities represent well defined concepts such as persons, locations, organizations, objects, etc. For example, given the sentence *"Obama is wrong to give work permits to young illegal immigrants"* we extract the entity *"Obama"*. Second, we extract the set of aspects, where aspects can be entity attributes or some abstract objects. In the previous example, the set of aspects are: *"Work permit* and *"illegal immigration"*. Note that for extracting entities we have used OpenCalais[8] and for aspects we have used Zemanta [9] to process a huge corpus containing $1,101,094$ Wikipedia articles.

3.3 Profile Similarity Measure

We have adopted cosine similarity to compute the similarity between profiles. This measure has been shown to be very effective in measuring similarity and detecting novelty between news articles [6]. In a standard search problem, a news article or user profile is represented by a vector of n dimensions where a term is assigned to each dimension and the value of the dimension represents the frequency of the term in the profile. In our setting we are interested in computing similarity between profiles described by a set of tuples, for this end we modify the vector representation as follows: each profile is represented by one vector

[7] http://opennlp.apache.org/
[8] http://www.opencalais.com
[9] http://www.zemanta.com/

representing the set of tuples and the value of each dimension represents the frequency of the tuple on news article or user profile.

4 Experiments

4.1 Setup

We have crawled a dataset based on the activities of 164 users from The Independent news site. The choice of this site was based on the fact that it has a large number of active users that continuously post comments on articles of various topics. Additionally and more importantly, users of The Independent follow also other news websites including The Telegraph, CNN and Al-Jazeera, so they have access to different types of articles covering different aspects for the same entity. For each of those users, we have crawled his comments in the four news sites mentioned earlier. Additionally, we have collected all the articles commented by each user from May 2010 to December 2013. Statistics about the number of comments and articles from each news web site are shown in Table 1. To evaluate our approach, we have randomly selected 23 users. For each user we performed recommendation at different time points $t_1, t_2, ..t_n$. The reason behind time dependent evaluation is two fold: (1) to take into account profile updates since users continuously post comments bringing new information about their interests, and (2) to use data before time point t_i for recommendation and data starting from time point t_i for assessment, as described later. The time points $t_1, t_2, ..t_n$ are chosen in such a way that between t_{i-1} and t_i, there is at least m comments posted by the user. In our experiments, we have set $m = 100$ to have enough evidence that the user profile needs to be updated. This setting resulted in 189 rounds of recommendation. We have simulated the recommendation system in the following way. For each user and at each time point t_i, we build the user profile based on his comments posted before t_i. Then, we choose as a seed article the first article that the user commented after time point t_i. We choose an article commented by the user to make sure that it matches user's interests. Based on the seed article and the user profile we return a set of articles that are similar to the seed article and at the same time have similar interests as the expressed in the user profile. Figure 1 shows the distribution of articles by topic. We can see

#Comments	482, 073
#Independent articles	26, 096
#Telegraph articles	23, 154
#CNN articles	535
#Aljazeera articles	303

Table 1: Datasets Statistics

that most articles and comments concerns politics. Note that the list of the seed articles we have selected follow a very similar distribution to the overall set of

articles. To assess the effectiveness of our approach we have used an automatic evaluation to avoid the subjectivity of manual assessments. We have considered the action of commenting on an article to be an indicator that the article fits the interests of the user. Based on this assumption, we check the list of recommended articles. The one that user has commented on are considered relevant. Note that it is probable that we systematically underestimate the interest of the user . A person might well be interested in an article even though he does not comment on it.

4.2 Results

We use two baselines strategies to assess our approach. The first one is based on aspect-centric profiles for both users and articles. The aspects were generated from users' comments and news articles content using Zemanta Api as we described earlier. The second strategy is based on entity-centric profiles for both users and articles. This strategy has been proposed in [1] and it represents our second baseline. We compare the two above strategies to our contribution where we define a global profile for both users and articles. To compare the results of the different strategies, we use Precision and NDCG at k ($P@k$ and $NDCG@k$). The $P@k$ is the fraction of recommended articles that are relevant to the user considering only the top-k results. It is given by:

$$P@k = \frac{|Relevant\_Articles \cap topk\_Articles\_Results|}{k}$$

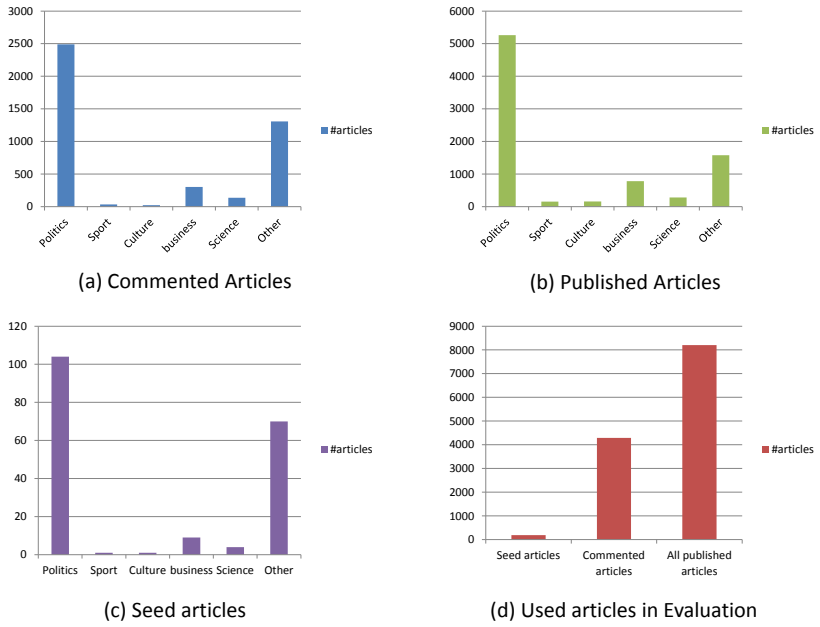


Fig. 1: Statistics about categories of used articles in Evaluation

Additionally, we compute $NDCG$ to measure the usefulness (gain) of recommended articles based on their (geometrically weighted) positions in the result list. It is computed as follows:

$$NDCG(E,k) = \frac{1}{|E|}\sum_{j=1}^{|E|} Z_{kj} \sum_{i=1}^{k} \frac{2^{rel(j,i)}-1}{log_2(1+i)}$$

where Z_{kj} is a normalization factor calculated to make $NDCG$ at k equal to 1 in case of perfect ranking, and $rel(j,i)$ is the relevance score of a news article at rank i. In our setting, relevance scores $rel(j,i)$ have two different values: 1(relevant) if the news article was commented by the user u, and 0(not relevant) if the news article was not commented by the user u. The precision and $NDCG$ results for the three strategies are shown in Table 2. We can clearly see that

	P@5	**P@10**	**NDCG @5**	**NDCG @10**
Aspect-centric Profile	0.396	0.392	0.734	0.689
Entity-centric Profile [1]	0.412	0.409	0.806	0.768
Global Profile	**0.52**	**0.507**	**0.855**	**0.797**

Table 2: Precision and NDCG values for all users

our approach of using global profile outperforms the baseline approach with a gain of 10% in terms of precision and 5% in term of ranking at NDCG@5. We also observe that using only aspects to build user and article profiles performs worst. The reason is that most of the news articles do not address certain aspects without relating them to some entities. Thus, disregarding entities leads to poor results. Moreover, when viewpoints are expressed about entities, they usually refer to certain aspects of those entities. Thus, using only entities to build profiles penalizes the performance. Consequently the combination of both entities and aspects give the best results. Note that real precision values must be higher than the one presented here. The reason is that comments can tell us if a user is interested in an article or not but their absence does not mean the opposite.

5 Conclusion and Future Works

In this paper, we have proposed a personalized news recommendation approach that takes into account fined-grained users interests. Existing approaches used only tags and entities to model interests which does not contain complete information. Thus, we have proposed a new model for user and article profiles based on entities and their related aspects. We have performed experiments based on four news websites, namely The Independent, The Telegraph, CNN and Al-Jjazeera. The results show that using both entities and aspects in the profile outperforms both entity-centric and aspect-centric approach with a minimum

precision gain of 10% and 5% in term of ranking at $NDCG@5$. This work represent a first attempt for a personalized news recommendation based on user and article viewpoints. As future works, we plan to test our model with larger set of users. It is also very promising to explore diversification techniques to improve our model by recommending articles outside of the current scope of the user profile.

References

1. S. Abbar, S. Amer-Yahia, P. Indyk, and S. Mahabadi. Real-time recommendation of diverse related articles. In *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 1–12, Republic and Canton of Geneva, Switzerland, 2013. International World Wide Web Conferences Steering Committee.
2. F. Abel, Q. Gao, G.-J. Houben, and K. Tao. Analyzing user modeling on twitter for personalized news recommendations. In *Proceedings of the 19th International Conference on User Modeling, Adaption, and Personalization*, UMAP'11, pages 1–12, Berlin, Heidelberg, 2011. Springer-Verlag.
3. J. Chen, R. Nairn, L. Nelson, M. Bernstein, and E. Chi. Short and tweet: Experiments on recommending content from information streams. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 1185–1194, New York, NY, USA, 2010. ACM.
4. L. Hong and B. D. Davison. Empirical study of topic modeling in twitter. In *Proceedings of the First Workshop on Social Media Analytics*, SOMA '10, pages 80–88, New York, NY, USA, 2010. ACM.
5. Q. Li, J. Wang, Y. P. Chen, and Z. Lin. User comments for news recommendation in forum-based social media. *Inf. Sci.*, 180(24):4929–4939, Dec. 2010.
6. Y. Lv, T. Moon, P. Kolari, Z. Zheng, X. Wang, and Y. Chang. Learning to model relatedness for news recommendation. In *Proceedings of the 20th International Conference on World Wide Web*, WWW '11, pages 57–66, New York, NY, USA, 2011. ACM.
7. M. Michelson and S. A. Macskassy. Discovering users' topics of interest on twitter: A first look. In *Proceedings of the Fourth Workshop on Analytics for Noisy Unstructured Text Data*, AND '10, pages 73–80, New York, NY, USA, 2010. ACM.
8. O. Phelan, K. McCarthy, and B. Smyth. Using twitter to recommend real-time topical news. In *Proceedings of the Third ACM Conference on Recommender Systems*, RecSys '09, pages 385–388, New York, NY, USA, 2009. ACM.
9. E. Shmueli, A. Kagian, Y. Koren, and R. Lempel. Care to comment?: Recommendations for commenting on news stories. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 429–438, New York, NY, USA, 2012. ACM.
10. J. Stoyanovich, S. Amer-yahia, C. Marlow, and C. Yu. Leveraging tagging to model user interests in del.icio.us. In *In AAAI SIP*, 2008.
11. J. Weng, E.-P. Lim, J. Jiang, and Q. He. Twitterrank: Finding topic-sensitive influential twitterers. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, WSDM '10, pages 261–270, New York, NY, USA, 2010. ACM.