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.

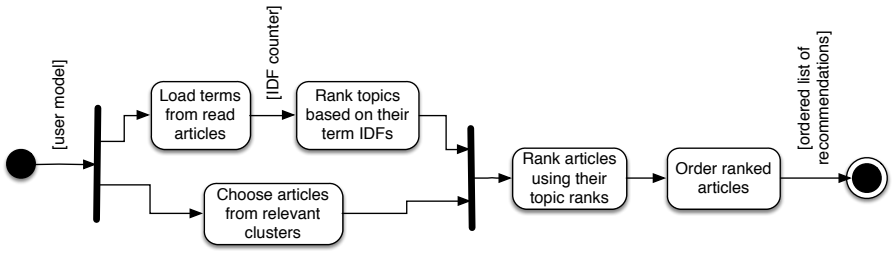

Fig. 1. Overview of the method. It first chooses relevant articles and ranks the topics using the user model. Then it ranks the relevant articles based on their topics and orders them by their rank.

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]

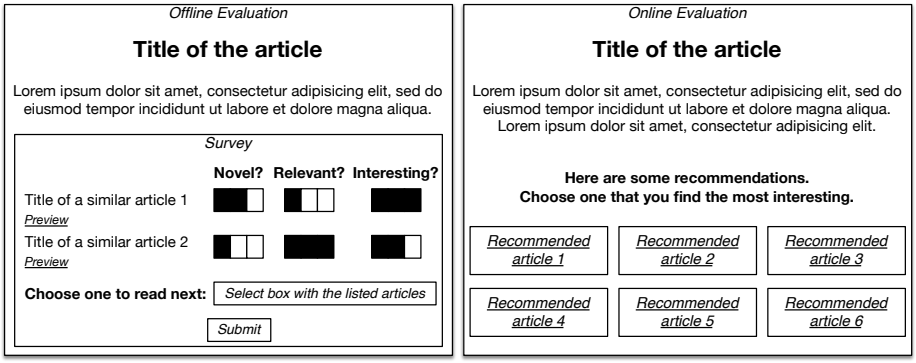

Fig. 2. Wireframes of the user interface used in the offline evaluation on the left and the online evaluation on the right. In the offline evaluation, the task was to rate the novelty, relevancy and interestingness of several articles compared to the one presented above on a given scale. The participants 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

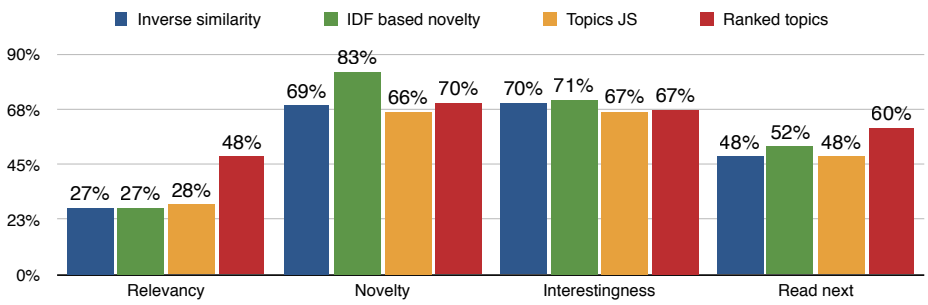

Fig. 3. Results from the offline evaluation of methods for novelty recommendation based on the data collected in our experiment.

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-

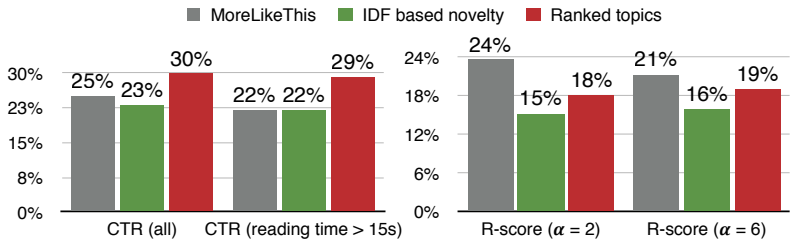

Fig. 4. CTR and R-score calculated based on the results from the online experiment.

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)