Increasing Top-20 diversity through recommendation post-processing

Matevž Kunaver[1], Tomaž Požrl[1], Štefan Dobravec[1], Uroš Droftina[1], and Andrej Košir[1]

[1]University of Ljubljana, Faculty of Electrical Engineering, Tržaška 25, Ljubljana, Slovenia
{matevz.kunaver,tomaz.pozrl,stefan.dobravec,andrej.kosir}@fe.uni-lj.si,uros.droftina@telekom.si

Abstract. This paper presents two different methods for diversifying recommendations that were developed as part of the ESWC2014 challenge. Both methods focus on post-processing recommendations provided by the baseline recommender system and have increased the ILD at the cost of final precision (measured with F@20). The authors feel that this method has potential yet requires further development and testing.

Keywords: Recommender system, Diversity, ILD, RecSys

1 Introduction

In this paper we present results obtained from participating in ESWC2014 challenge, where we developed and tested two methods for increasing recommendation diversity while preserving user satisfaction.

The focus of recommender systems (RS) is moving from generating recommendations without any additional data about the user to generating recommendations that also consider the user's context [1][3] and personality in order to improve the recommendation results[6]. All these improvements serve to present the user with a selection of items that will be the most appropriate for the situation in which the user desires to consume the selected item.

Recommendation results can be further improved by paying attention to the diversity [4] [7] [5] [9] of recommendations presented to the user. In order to measure diversity one must have additional information available about the recommended items such as their meta-data, descriptions, technical specifications etc.. Obtaining this data can be a problem since most of systems either use their own descriptions or do not update their data regularly. This is where Linked Open-Data enabled (LOD) systems offer a significant advantage as they work with data accumulated from various sources over the internet.

1.1 Motivation and Goal

We performed this study as part of the LOD enabled RS challenge of 11th European Semantic Web Conference (ESWC-14) where we focused on task 3 of the challenge - diversity that addressed an interesting aspect of content-based RS - using diversification to avoid over-specialization. As an extra bonus the task also provided evaluation tools that enabled us to immediately measure our results and compare them with those of others.

The purpose of our study was therefore to determine whether we can increase the diversity of items presented to the user by post-processing results provided by a non-diversified RS while maintaining user satisfaction measured by the predicted rating security.

2 Materials and Methods

In this section we describe the dataset, the baseline RS used to generate recommendations, the diversification methods developed as part of the challenge and the evaluation methods used to measure the diversity of recommendations.

2.1 Dataset

We used the DBook dataset provided by the challenge enhanced with meta-data retrieved from DBpedia. Each item from the dataset was described with the DBpedia ontology, featuring 17 fields (author, year of publishing, type etc.) and Dublin Core categories featuring 7067 different values, with each item having on average 5 different categories.

2.2 Recommender System

Since our approach focused on post-processing the results provided by a content-based RS we used a RS developed as part of our previous research [8]. This RS used a rule-based approach that considered all attributes and categories available in the dataset.

We diversified the Top-20 lists using two methods that replaced items in the recommendations list. The idea was to replace some of the Top-20 items with recommendations that would increase the overall diversity of the list without having a strong negative impact on the overall accuracy of the recommender system (measured with F@20 metric).

Figure 1 shows the data flow of our recommendation process.

2.3 Diversification method

Our diversification methods focused on finding the best items to replace on the top 20 list and the best candidates to replace them with. We used two versions of this algorithm.

The first version calculated the ILD value of the top 20 list while excluding one item (effectively calculating the ILD@19 instead of ILD@20). This process was repeated until each item on the list was excluded once. This created a list of items and ILD@19 values that was then sorted in ascending order by

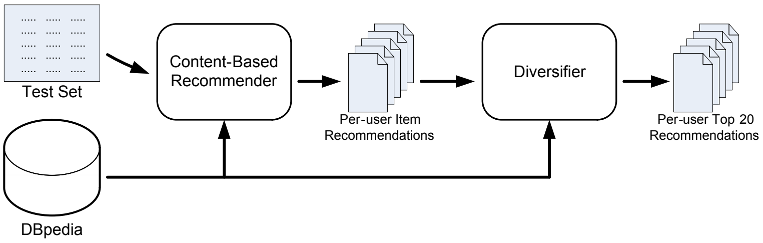
Test Set
DBpedia
Content-Based Recommender
Per-user Item Recommendations
Diversifier
Per-user Top 20 Recommendations

Fig. 1. Recommender system dataflow

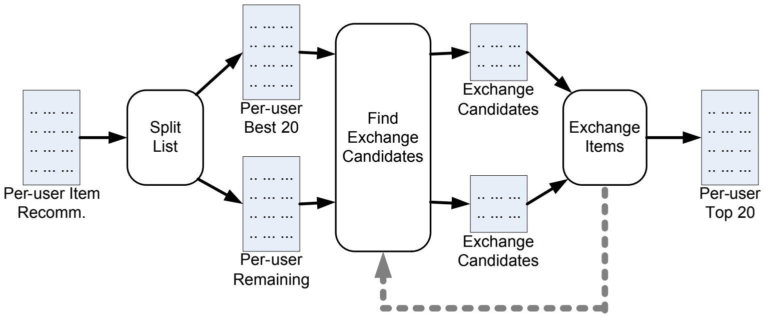

Fig. 2. Diversification method

the ILD@19 value. The idea was that the highest ILD@19 value indicated that excluded item had the smallest impact on the diversity of the list and could therefore be replaced with an item that would add more diversity to the final selection of items. The replacement candidates were selected from the remaining recommendations - in our case we considered items from 21 to 40 (if sorted by predicted ratings), since a larger number of candidates required too much processing time to be completed until the end of the challenge. We removed the item with the highest ILD@19 value and calculated the ILD@20 for each of the replacement candidates. We repeated the process by removing two items from the original list and replacing them with all the possible combinations of replacement candidates and again calculating ILD@20. This resulted with a list of 400 ILD@20 for each user from which we selected the list with the highest value as our top 20 list for each user.

The second version focused on replacing single item at a time, where a joint score in form of $a * avgPR + b * nILD$ was considered instead of pure ILD value. In this formulation $avgPR$ stands for the average prediction rating of the list and nILD for the normalized ILD value of the same list. Parameters a and b allow balancing the top20 list from more accurate / less diverse towards less accurate / more diverse. Similar to the first version the replacement candidates were calculated by excluding the worst item from the top 20 list and replacing it with the best item in the bottom list. The shuffling procedure was repeated until best top 20 list (in term of joint score) was achieved. In test, values 2/3 and 1/3 were chosen for parmeters a and b on an empirical basis.

2.4 Evaluation methods

We evaluated our methods using two different methods. We calculated ILD@20 [9] using attributes based on the equation and Java package provided as part of the challenge documentation. This evaluation was used to calculate the partial ILD (ILD@19 as described above) during the diversification process. We used 14 different attributes in the ILD calculations as these attributes were all used in the recommender system used to generate the preliminary recommendations.

Once we had the Top-20 lists for each user we also used the on-line submission system to receive the official evaluation results and to compare them with those of other research groups.

3 Results

Table 1 shows the results of our baseline recommender system, developed diversification methods and those of a random recommender for comparison.

Table 1. Evaluation results

method	P@20	R@20	F@20	ILD
random	0.0008	0.0020	0.0012	0.4853
non-diversified	0.0203	0.0644	0.0309	0.4549
diversified - version 1	0.0017	0.0047	0.0025	0.4670
diversified - version 2	0.0017	0.0050	0.0026	0.4609

4 Discussion

Table 1 shows that both our diversification approaches noticeably increased the diversity of the users Top-20 list, yet did so at the cost of precision and recall. A comparison of our results with those of a random recommender and shows that we get better F@20 in all cases while having a lower ILD value which is logical due to the random RS selection of completely random items. Unfortunately we lacked the time to preform further statistical analysis of our results.

The comparison of results shows that we increased the diversity of our top lists by 3% (for comparison - the ILD value of the random RS is 6% larger), while decreasing the F@20 value by as much as 90%. This would imply that our method focused too much on diversification and might provide better results with further parameter tweaking.

5 Conclusion and Further Work

Since there was a time limit we were unable to perform all the tests that we desired, leaving open quite a few questions. The main issues that we plan to address and present as an article to be published at a later date are:

- Determine whether the number of replaced items from the top list can be fixed or must be calculated iteratively for each user each time the RS generates recommendations.
- The number of replacement candidates to be considered.
- Perform a series of statistical tests in order to determine whether our results are really significantly different from those of a non-diversified (or random) RS.
- Determine the optimal values of parameters a and b for the second method we developed.
- Perform an A/B test to determine how the lower accuracy impacts the actual user satisfaction.

We will also apply these methods to our *Context Movie Dataset* (LDOS-CoMoDa) [2] which also features live users thus making an A/B test a possibility.

Participating in this challenge provided a very good experience since we tackled a completely new dataset and had the appropriate evaluation tools at our disposal.

6 Acknowledgments

Operation part financed by the European Union, European Social Fund.

References

1. Dey A. and Abowd G. Towards a better understanding of context and context-awareness. pages 304–307, 199.
2. Košir A., Odic A., Kunaver M., Tkalcic M., and Tasic J. F. Database for contextual personalization. *Elektrotehniški vestnik*, 78(5):270–274, 2011.
3. Odic A., Tkalcic M., and Košir A. Managing irrelevant contextual categories in a movie recommender system. *Human Decision Making in Recommender Systems (Decisions@ RecSys 13)*, page 29, 2013.
4. Smyth B. and McClave P. Similarity vs. diversity. In *Case-Based Reasoning Research and Development*, pages 347–361. Springer, 2001.
5. Jannach D., Lerche L., Gedikli F., and Bonnin G. What recommenders recommend–an analysis of accuracy, popularity, and sales diversity effects. In *User Modeling, Adaptation, and Personalization*, pages 25–37. Springer, 2013.
6. Adomavicius G. and Tuzhilin A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, 2005.
7. Adomavicius G. and Kwon Y. Improving aggregate recommendation diversity using ranking-based techniques. *Knowledge and Data Engineering, IEEE Transactions on*, 24(5):896–911, 2012.

8. T Požrl, M Kunaver, M Pogačnik, A Košir, and JF Tasič. Improving human-computer interaction in personalized tv recommender. *International Journal of Science and Technology, Transactions of Electrical Engineering*, 36(E1):19–36, 2012.
9. Cai-Nicolas Ziegler, Sean M McNee, Joseph A Konstan, and Georg Lausen. Improving recommendation lists through topic diversification. In *Proceedings of the 14th international conference on World Wide Web*, pages 22–32. ACM, 2005.