Improving Recommendations Through Re-Ranking Of Results

S.Ashwini

M.Tech, Computer Science Engineering,
MLRIT, Hyderabad, Andhra Pradesh, India

T.Chandra Sekhara Reddy

Associate Professor, Department of CSE
MLRIT, Hyderabad, Andhra Pradesh, India

Abstract*—World Wide Web has become a good source for any kind of information. People of all walks of life depend on it for their information retrieval through queries. However users get huge number of records or recommendations. It causes problem to users as they do not see the intended results immediately. Instead they have to spend some time browsing for intended content. This kind of user experience is not good. For this reason there is a need for improving recommender systems. Ranking is one of the techniques for improving search results. Many ranking techniques came into existence. The recommender systems are using them. In this paper we propose a re-ranking algorithm that improves the ranked results that will give rich user experience. We build a prototype to show the efficiency of the proposed approach. The empirical results revealed that the application is very useful.

Index Terms – Recommender systems, ranking, re-ranking, recommendations

I.INTRODUCTION

The information super highway has become a reliable source for information retrieval. It has given an opportunity to researcher for mining such content and get useful information. Many data mining techniques have been used to discover knowledge from the WWW. The discovered knowledge helps in making well informed decisions. The underlying patterns can be discovered through web mining. Such patterns give bui8sness intelligence that leads to effective decisions. For this reason, from researchers' point of view, the WWW is a goldmine. To help users online, many recommender systems are available. The recommender systems use ranking models to provide quality information to end users. As the information retrieval systems provide plethora of records, it is useful to have recommender systems. The ranking models used in the recommender systems include RankBoost [4], SVM [5], [6], ListNet [2] and LambdaRank [1]. The performance of search engines and recommender systems is improved using ranking algorithms.

Volumes of information are available over WWW which can be obtained through search engines and recommender systems. Many search engines like Ask, Yahoo, Google, and Bing and so on are used to obtain information. Generally users visits first two pages only [7], [8]. This indicates that users are interested to view only first two pages. When huge number of pages is retrieved they are not willing to move to all pages to find the required information. Search engines generally use Page Ranking algorithm in order to show the best results first. These activities come under the discipline of web mining. The web mining is of three types again. They are known as web structure mining, web usage mining and web content mining [9]. Out of them web mining is widely used for information retrieval. Web structure mining focuses on the structural summary of web documents. The web usage mining is responsible to find patters pertaining to user navigation.

In this paper we mine web graphs for recommendations. The results of web mining that is recommendations are used further for re-ranking process. The re-ranking helps in improving the results so as to make them more useful to end users. We also build a prototype that shows the efficiency of the proposed algorithm. The empirical results revealed that the application is very useful. The remainder of the paper is structured as follows. Section II reviews relevant literature. Section III provides information about the proposed algorithm. Section IV presents experimental results while section V concludes the paper.

II.RELATED WORK

Web mining is a popular mining domain for many years. This helps researchers to discover useful information from WWW. Search engines play an important role in information retrieval. The crawlers of web provide required information to end users. They also present information using some ranking mechanism. Through ranking users can get more important information at the beginning of search results.

The first page results are very important for end users. These results should be intended results. Page Ranking algorithm is used by search engines to make it [10], [11]. Therefore the results of search engines and recommender systems make use of a ranking algorithm to provide optimum results in first two pages. Such systems also maintain log containing id, query that has been issued and the results besides the clicked URLs, associated ranks, and so on. Researchers have been using such search logs for discovering latent knowledge [12], [13], [14], [10]. The results of such researches help in many real time applications. They are very useful to both search engines and also end users. These results are used by applications to improve the results [10], [11]. In [13] search results are improved automatically by using a novel approach. In [14] relationship is found between the query logs and queries to use in recommendation systems. Many such recommendation systems are available in the literature [15], [16], [17], [18]. In this paper the re-ranking framework is based on the framework presented in figure 1.

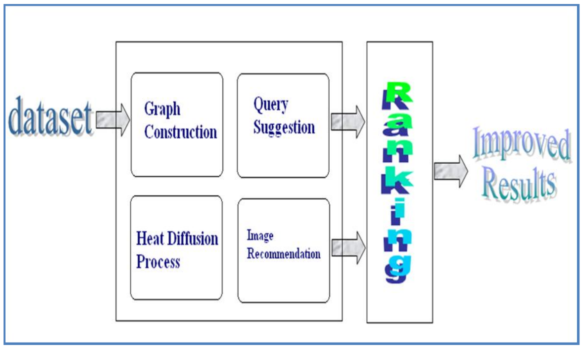


Fig. 1 – Recommender system presented in our previous work [19]

As seen in figure 1, the recommender system takes dataset as input and performs activities such as graph construction and heat diffusion for query suggestion and image recommendations. The results of image recommendations and also query suggestions are ranked in order to provide improved results. We improve this framework further in this paper. We include a re-ranking process into the framework to enhance it. This will optimize the ranked results so as to improve user satisfaction towards intended results.

Best results will appear in first page. AOL click through dataset is used for query suggestions while the Flikr dataset is used for generating image recommendations.

III.PROPOSED FRAMEWORK and ALGORITHM

The framework described in the previous section has been improved with re-ranking provision. The ranked results provided by recommender systems are used for improving the results. The optimization of results can help end users to get more intended information in the first or second page of results. This will enrich user experience. The enhanced framework is presented in figure 2.

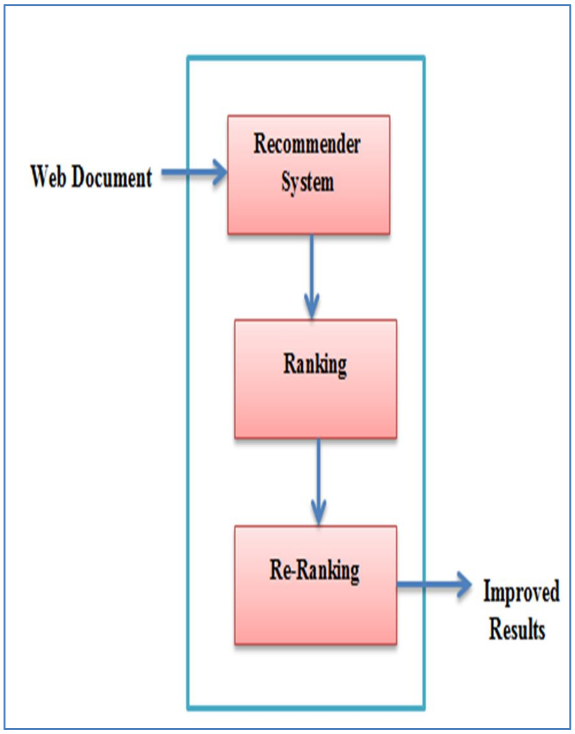
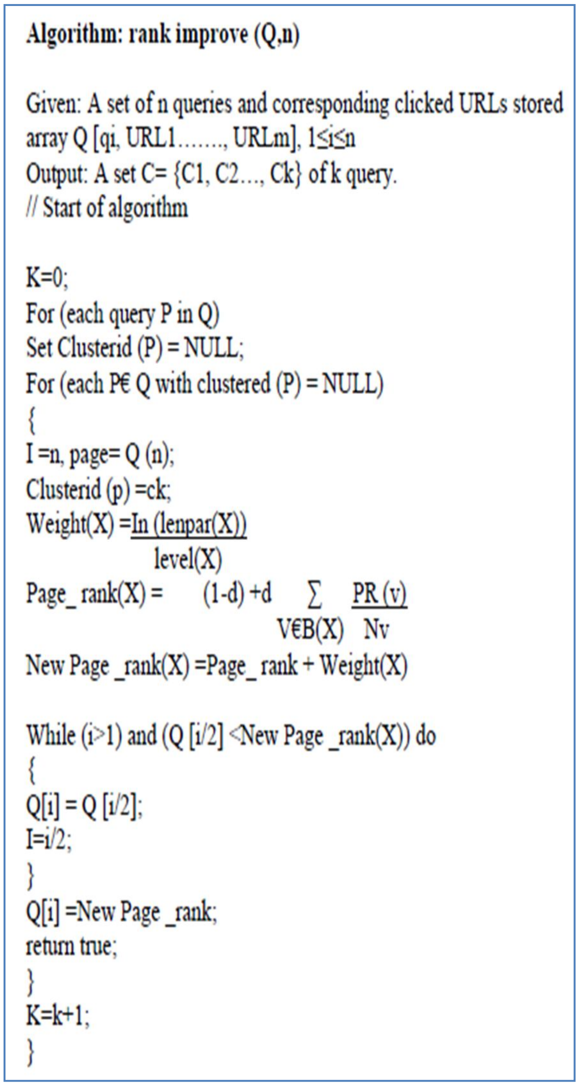


Fig. 2 – Re-ranking framework

As seen in figure 2, re-ranking part is included in the framework in order to improve the recommendations framework presented in figure 1. The framework for ranking recommendations can be found in [19]. The proposed algorithm for re-ranking is named as “Rank Improvement” algorithm which optimizes results further to make them more meaningful to end users. The algorithm used for re-ranking is as shown in figure 3.



```
Algorithm: rank improve (Q,n)

Given: A set of n queries and corresponding clicked URLs stored
array Q [qi, URL1……., URLm], 1≤i≤n
Output: A set C= {C1, C2…, Ck} of k query.
// Start of algorithm

K=0;
For (each query P in Q)
Set Clusterid (P) = NULL;
For (each P€ Q with clustered (P) = NULL)
{
I =n, page= Q (n);
Clusterid (p) =ck;
Weight(X) = $\frac{In (lenpar(X))}{level(X)}$
Page_ rank(X) = (1-d) +d $\sum_{V€B(X)} \frac{PR (v)}{Nv}$
New Page _rank(X) =Page_ rank + Weight(X)

While (i>1) and (Q [i/2] <New Page _rank(X)) do
{
Q[i] = Q [i/2];
I=i/2;
}
Q[i] =New Page _rank;
return true;
}
K=k+1;
}
```

Fig. 3 – Rank improvement algorithm

As shown in figure 3, the re-ranking algorithm is presented. The algorithm takes results of the recommender system and performs re-ranking on it.

The results of the algorithm are more optimal to the satisfaction of end users.

IV.EXPERIMENTAL RESULTS

To demonstrate the efficiency of the re-ranking algorithm, we built a prototype application. We improved the framework presented in [19] to enhance the search results. Experiments are made using our re-ranking algorithm as presented in figure 3.

The environment used to build application is Java platform, a PC with 4 GB RAM, Core 2 Dual processor running in Windows 7 operating system. The IDE used to build the application is NetBeans. After performing experiments with the system we also evaluated trough human participants. We invited 1000 end users and took feedback from them. Their feedback is recorded. The users have given their feedback based on their experience with the system. The satisfaction level of the end users is averaged for both the systems (with and without re-ranking). The results are presented in figure 4.

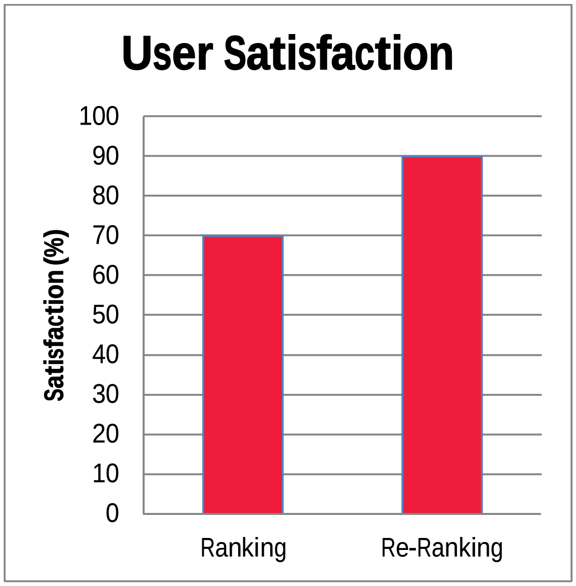


Fig. 4 – Level of satisfaction of end users for both systems

The feedback of 1000 users is presented for both recommender systems with ranking and re-ranking mechanisms. The results reveal that 70% and 90% are the average satisfaction levels with ranking and re-ranking systems respectively. From this it can be said that the recommender system with re-ranking of results has more utility that improves satisfaction level of end users. The proposed re-ranking algorithm is capable of improving level of user satisfaction.

V.CONCLUSION

In this paper we implement a re-ranking algorithm that enhances the results which have been ranked by recommender systems. This is required because the recommender systems might provide huge number of records. End users will not be able to see intended results without spending time for further surfing. The ranking algorithms used by recommender system provide results based on ranking. However, the results are more in number thus they need further improvement. We built a prototype application that shows the efficiency of the proposed re-ranking algorithm. The empirical results are encouraging.

REFERENCES

[1] C.J.C. Burges, R. Ragno, and Q.V. Le, “Learning to Rank withNonsmooth Cost Functions,” Proc. Advances in Neural InformationProcessing Systems (NIPS ’06), pp. 193-200, 2006.

[2] Z. Cao and T. Yan Liu, “Learning to Rank: From PairwiseApproach to Listwise Approach,” Proc. 24th Int’l Conf. MachineLearning (ICML ’07), pp. 129-136, 2007.

[3] C.J.C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N.Hamilton, and G. Hullender, “Learning to Rank Using GradientDescent,” Proc. 22th Int’l Conf. Machine Learning (ICML ’05), 2005.

[4] Y. Freund, R. Iyer, R.E. Schapire, Y. Singer, and G. Dietterich, “An Efficient Boosting Algorithm for Combining Preferences,”J. Machine Learning Research, vol. 4, pp. 933-969, 2003.

[5] R. Herbrich, T. Graepel, and K. Obermayer, “Large Margin RankBoundaries for Ordinal Regression,” Advances in Large MarginClassifiers, pp. 115-132, MIT Press, 2000.

[6] T. Joachims, “Optimizing Search Engines Using ClickthroughData,” Proc. Eighth ACM SIGKDD Int’l Conf. Knowledge Discoveryand Data Mining (KDD ’02), pp. 133-142, 2002.

[7] A.Spink, D.Wolfram,B.J.Jansen,T.Saracevis,”Searching the Web :The public and their queries ”.jouurnel of the amercianSocitey for information Science and technology 52(3),2001,226-234.

[8] R.Cooley, B.Mobasher and J.Srivastava, “Web mining: Information and pattern discovery on the World Wide Web,”. In 9th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 97) 1997.

[9] M.H.Dunham,Companion slides for the text ,” Data mining Introductory and advanced topics ”.Prentice Hall,2002.

[10] IsakTaka,SarahZelikovitz,AmandaSpink,”Web Search log to Identify Query Classficationterms”Proceeding of IEEE International Conference on Information Technology (ITNG’07),pp:50-57,2008.

[11] L. Page,S.Brin ,R.Motwani,T.Winograd,”The pagerankcitration ranking :Bringing order to the Web ”.Technical report ,Stanford Digital Libraries SIDL-WP-1990-0120,1999.

[12] J.Wen, J.Mie, and H.zhang,” Clustering user queries of a search engine ”.In Proc of 10th International WWW Conference .W3C,2001.

[13] Thorsten joachims,”optimizing search engine using clickthough data” Proceeding of the 8th ACM SIGKDD international conference on knowledge discovery and data mining ,2002,pp:133-142, New York.

[14] H.Ma, H.Yang,I.King,and M.R.Lyu,”learning latent semantic relations from clickthough data from query suggestion ”.InCIKM’08:Proceeding oh the 17th ACM conference on information and knowledge management ,pages 709-708,New York,ny,USA,2008,ACM.

[15] J.L. Herlocker, J.A. Konstan, L.G. Terveen, and J.T. Riedl, “Evaluating Collaborative Filtering Recommender Systems,” ACM Trans. Information Systems, vol. 22, no. 1, pp. 5-53, 2004.

[16] T. Hofmann, “Collaborative Filtering via Gaussian Probabilistic Latent Semantic Analysis,” SIGIR ’03: Proc. 26th Ann. Int’l ACM SIGIR Conf. Research and Development in Information Retrieval, pp. 259-266, 2003.

[17] T. Hofmann, “Latent Semantic Models for Collaborative Filtering,” ACM Trans. Information Systems, vol. 22, no. 1, pp. 89-115, 2004.

[18] Z. Huang, H. Chen, and D. Zeng, “Applying Associative Retrieval Techniques to Alleviate the Sparsity Problem in Collaborative Filtering,” ACM Trans. Information Systems, vol. 22, no. 1, pp. 116-142, 2004.

[19] Ch.Nagini* M.Srinivasa Rao R.V.Krishnaiah, “ A General Framework for Recommendations on the Web”. Volume 3, Issue 5, May 2013

AUTHORS

S.Ashwini, she is pursuing M.Tech (CSE) in MLRIT, Hyderabad, AP, INDIA. She has received B.Tech degree in Computer Science and Engineering.Her main research interest includes Data Mining and Databases.

T.Chandra Sekhara Reddy, M.Tech (Software Engineering). He is currently working as Associate Professor in Department of Computer Science and Engineering, MLRIT, Andhra Pradesh, India. He has more than six years of teaching experience. He has received M.Tech software Engineering degree in the year 2007. He has completed B.Tech CS & IT in the year 2005.He has received Diploma in Electronics & Communication Engineering in the year 2001.He published three Internation Journals in Wireless Sensor Networks and Stegnography.