ISSN:0975-9646

Mining Suggestions for Recommender Systems

B.Nasreen[1], A. Safiya Parvin[2]

[1] *Post graduate student,*
Sathyabama University,Chennai.
[2] *Faculty,*
Sathyabama University,Chennai.

Abstract-Recommender systems are systems that suggest items to the users of the web to help them find most relevant items of interest. These recommended items could include a movie, an image, query suggestions, tag recommendation etc. Data sources are modelled as graphs and are subjected to a naive diffusion method called heat diffusion method. The heat diffusion method identifies the similarities between different nodes to suggest recommendations. This method is applied to query suggestion which is a technique used to provide related queries to the user's search query. A query-URL bipartite graph is constructed which is then mined to generate top-K recommendations for the query. In order to improve the search results, the heat values of the URL are also taken into account during the ranking of results. This technique is also extended to image recommendation.

Index Terms— Recommendation, query suggestion, image recommendation

1. INTRODUCTION

The widespread information available on the web is highly unstructured and user specific. Thus there is a need for systems that suggest items to users of the web to help them locate the most relevant item of interest. These systems typically are called recommender systems. Two different
The widespread information available on the web is highly unstructured and user specific. Thus there is a need for systems that suggest items to users of the web to help them locate the most relevant item of interest. These systems typically are called recommender systems. Two different approaches exist for recommender systems, content-based filtering and collaborative filtering [1], [13]. Recommendations based on an item to item similarity measure falls under the category of content based systems. Collaborative filtering in turn uses patterns in user item matrix to find out similarities and generate recommendations.

Recommender problems are pervasive but their success depends on effectively matching users to items in different contexts based on some utilities like click-rates, buy rates, movie ratings and so on. The utilities are not known and have to be estimated through data. Two approaches exist for recommender systems, content-based filtering and collaborative filtering [1], [13]. Recommendations based on an item to item similarity measure falls under the category of content based systems. Collaborative filtering in turn uses patterns in user item matrix to find out similarities and generate recommendations.

Recommender problems are pervasive but their success depends on effectively matching users to items in different contexts based on some utilities like click-rates, buy rates, movie ratings and so on. The utilities are not known and have to be estimated through data. The unbalanced data distribution induces significant data sparseness and makes regularization and smoothing essential. Several modern machine learning models [2] tackle the problem of cold start and data sparsity in high dimensional categorical data. Another challenge is to take into account the personalization feature. This paper puts forward a general method called heat diffusion method that can be applied to many recommendation tasks on the web to provide latent semantically related results.

2. RELATED WORK

2.1 Collaborative Filtering

Collaborative filtering algorithms usually require a user-item rating matrix. It finds out patterns within a dataset of user-item ratings. This method detects many interesting similarities between items that are not obvious given the item properties. Algorithms that directly work with the user-item rating matrix are known as memory based while algorithms that construct some model (e.g. a decision tree) are known as model based [11]. Given that we are only allowed a limited number of recommendations, it is important to try to maximize the value the user gains from the recommendations. However, on the Web, in most of the cases, rating data are always unavailable since information on the Web is less structured and more diverse. Hence, collaborative filtering algorithms cannot be directly applied to most of the recommendation tasks on the Web, like query suggestion and image recommendation

2.2 Query logs and clickthrough data

Query suggestion aims to suggest full queries from the queries typed by users previously [4] so that the integrity and the coherence among the queries are preserved. Queries are recommended based on user logs [9] and also from click through data. [3] proposes to optimize the recommendations on web portals by mining the clickthrough data. Based on [5], a context-aware query suggestion method is developed by mining clickthrough and session data. This work first extracts some concepts from the click through data by building clusters. Then, these concepts as well as the query sessions are employed to build a concept sequence suffix tree for query suggestion. Mei et al. [6] proposes a query suggestion method using hitting time on the query click bipartite graph. This method can generate semantically relevant queries to users' information needs. The main advantage of this work is that it can suggest some long tail queries (infrequent queries) to users.

3 SUGGESTIONS USING HEAT DIFFUSION

A clickthrough data consists of the following information: user id, query, URL on which the user clicked, the rank of the URL, and time, which can be represented by the tuple <u,q,l,r,t>.The relationships between queries and the URLs they correspond to, can be captured as a bipartite graph with queries as one set and URLs as other set [14]. The bipartite graph is first converted into a directed graph $G = \{V, E, W\}$ with n nodes where V is the vertex set, E is the set of all edges and W is the weight associated with each edge. The query-URL edge is weighed by normalizing the number of times the query is issued. The URL-query edge is weighed by normalizing the number of times the URL is clicked. A subgraph is constructed using depth first search of G.

Let τ_i be a flag to identify if node v_i has any outlinks. Heat matrix (H) and diagonal matrix (D) be defined as

$$H_{ij} = \begin{cases} \frac{w_{ji}}{\sum_{(j,k)\in E} w_{jk}} \\ 0, when\ i = j \\ 0, otherwise \end{cases} \quad (1)$$

$$D_{ij} = \begin{cases} \tau_i, & i = j \\ 0, otherwise \end{cases} \quad (2)$$

Let $f(0)$ be the initial heat value of the query q. Let $f(1)$ represent the heat at the node at the next unit of time t. Let R denote the randomness of nodes connecting to each other. Let γ denote the probability of a random relation existing in the graph. Let g denote the stochastic distribution vector. Set $\propto = 1$ and $\gamma = 0.85$.

The heat values is computed by,

$$f(1) = e^{\propto R} f(0) \quad (3)$$

where,

$$R = \gamma(H - D) + (1 - \gamma)g \quad (4)$$

$$g = \frac{1}{n} \quad (5)$$

The top K queries are ranked based on the largest values of heat of queries and the URLs. As reported in [6], several ranking algorithms exists which includes PageRank [7], HITS [8] etc. Ranking using similarity is proposed in SimRank [10] and ranking using affinity graph is also proposed in [12].

The diffusion between the nodes in the graph will generate heat values for both the URLs and the queries. For a given query, after the diffusion process, the heat values represent the relatedness to the original query. When search results are ranked based on the heat values of the queries and the URLs, search results can be improved to a great extent.

Experiments are done in AOL clickthrough dataset. A snapshot of the dataset format is presented in Table 1.

Anon ID	Query	Calendar time	Rank	Click URL
217	Theonering	5/18/2006 18:20	1	http://www.theonering.net
217	ask.com	3/31/2006 14:31	1	http://www.ask.com
2178	people search	1/12/2006 07:30	8	http://people.yahoo.com

Table 1. Format of AOL clickthrough dataset

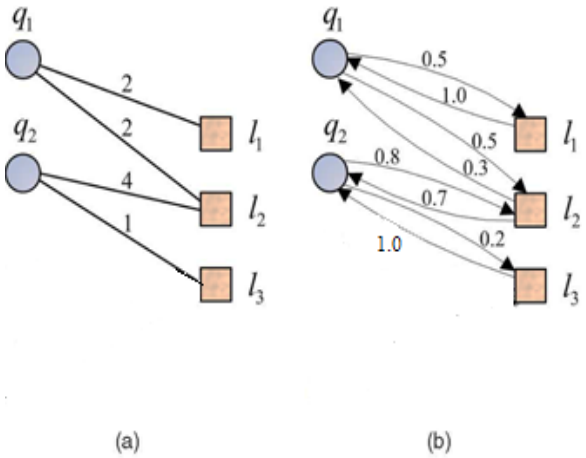

Fig 1. Construction of bipartite graph. (a)Query URL bipartite graph. (b)converted graph

After cleaning and removing duplicates, the query URL bipartite graph is constructed as shown in Fig 1(a). This undirected graph is converted into a directed graph as shown in Fig 1(b). The edges in the graph are normalized based on the weight of the edges. For example, query q_2 connects to links l_1 and l_3. Weight of edge q_2-l_2=(4/5)=0.8 After construction of the graph, query suggestion is done using the heat values of both the queries and the URLs.

4 CONCLUSION

A general framework for recommendations on web graphs is proposed. This can be applied to many recommendation tasks such as query suggestion, image recommendation, etc.

REFERENCES

[1] A. S. Das, M. Datar, A. Garg, and S. Rajaram, "Google news personalization: scalable online collaborative filtering", *Proc. of WWW*, Banff, Alberta, Canada, 2007.

[2] D. Agarwal, B.C. Chen and B. Long, "Localized Factor Models for Multi-Context Recommendation", In KDD, 2011

[3] D. Agarwal, B.-C. Chen, P. Elango and X. Wang, "Click Shaping to Optimize Multiple Objectives", In KDD, 2011.

[4] X. Wang and C. Zhai, "Learn from web search logs to organize search results", In *Proc. of SIGIR*, Amsterdam, The Netherlands, 2007.

[5] H. Cao, Daxin Jiang, Jian Pei, Qi He, Zhen Liao, Enhong Chen, Hang Li, "Context-Aware Query Suggestion by Mining Click-Through and Session Data", *KDD'08*, August 24–27, 2008, Las Vegas, Nevada, USA.

[6] Q. Mei, D. Zhou, and K. Church, "Query Suggestion Using Hitting Time," CIKM '08: Proc. 17th ACM Conf. Information and Knowledge Management, pp. 469-477, 2008.
[7] S. Brin and L. Page, "The Anatomy of a Large-Scale Hypertextual Web Search Engine," Computer Networks and ISDN Systems, vol. 30, nos. 1-7, pp. 107-117, 1998.
[8] J.M. Kleinberg, "Authoritative Sources in a Hyperlinked Environment," ACM, vol. 46, no. 5, pp. 604-632, 1999.
[9] Cui, J.-R. Wen, J.-Y. Nie and W.-Y. Ma, "Query expansion by mining user logs", *IEEE Trans. Knowl. Data Eng.*, 15(4):829–839, 2003.
[10] G. Jeh and J. Wido, "Simrank: a measure of structural-context similarity", In *Proc. of KDD*, pages 538–543, Edmonton, Alberta, Canada, 2002.
[11] M. Deshpande and G. Karypis, "Item-based top-n recommendation", *ACM Transactions on Information Systems*, 22(1):143–177, 2004.
[12] B. Zhang, H. Li, Y. Liu, L. Ji, W. Xi, W. Fan, Z. Chen, and W.-Y. Ma, "Improving web search results using affinity graph", In *Proc. of SIGIR*, pages 504–511, Salvador, Brazil, 2005.
[13] Guy Shani, Max Chickering, Christopher Meek, "Mining recommendations from the web", *RecSys'08*, October 23–25, 2008, Lausanne, Switzerland.
[14] H. Ma, I. King, M.R. Lyu, "Mining web graphs for recommendations", *IEEE Trans. Knowl. Data Eng, 2012.*