Asian Journal of Applied Science and Technology (AJAST)
Volume 1, Issue 6, Pages 71-73, July 2017

User Preference Suggestion to Learn Behavioral Patterns

S.Sasireka[1] and C.Mariyammal[2]

[1]*Assistant Professor, Department of Computer Science and Engineering, Bannari Amman Institute of Technology, Sathyamangalam, Tamilnadu, India.*
[2]*Assistant Professor, Department of Computer Science and Engineering, Bannari Amman Institute of Technology, Sathyamangalam, Tamilnadu, India.*

Article Received: 03 July 2017 *Article Accepted: 21 July 2017* *Article Published: 23 July 2017*

ABSTRACT

Web Usage Mining is a part of Web Mining, a part of Data Mining. Apart from the concept of extracting meaningful and valuable information from large volume of data, Web Usage mining involves mining the usage characteristics of the users of Web Applications. This extracted information paves the ways in improvement of the application, checking of fraudulent elements etc. for ranking the user community, we use various classification and clustering algorithms such as Bayesian algorithm, click through streaming, k- means , fuzzy c-means. Based on these algorithms, we organize user search histories and learning of user behavioral patterns in social media, ranking can be performed.

1. INTRODUCTION

Various studies on query logs (e.g., Yahoo's [1] and AltaVista's [2]) reveal that only about 20% of queries are navigational. The rest are informational or transactional in nature. This is because users now pursue much broader informational and task-oriented goals such as arranging for future travel, managing their finances, or planning their purchase decisions. One important step towards enabling services and features that can help users during their complex search quests online is the capability to identify and group related queries together. Recently, some of the major search engines have introduced a new "Search History" feature, which allows users to track their online searches by recording their queries and clicks. This history includes a sequence of the queries displayed in reverse chronological order together with their corresponding clicks. Due to the advancement in computing and communication technologies especially in social networking sites like you tube, face book etc. enables people to get together and share information in innovative ways. This includes collaboration, communication and collective intelligence aiming in user interaction. This behavior analysis and patterns are to predict user behavior in heterogeneous (distinctive relations) social network. Distinctive connection limits the effectiveness of the prediction and provides an efficient approach to user behavior clustering in social network website; latent social dimensions are extracted based on network topology to capture the potential affiliations of users in the social network and the extracted social dimensions represent how each actor is involved in diverse affiliations and can be treated as features of actors for subsequent discriminative learning. The proposed (dictionary learning) algorithm which we organize user search histories and learning of user behavioral patterns in social media, hence ranking can be performed.

2. METHODOLOGY

1. Search logger
2. Update checker
3. Bucket creation for personalization
4. Behavioral future analysis
5. Learning algorithms
6. Pattern discovery
7. Ranking

2.1 Search Logger

In this module we develop a Meta search engine as the backend search engines to ensure a broad topical coverage of the search results. The Meta search engine collects click through data from the users and performs personalized ranking of the search results based on the learnt profiles of the users. Only top 100 results are returned to the user for every query posted by the user. The users are given the tasks to find results that are relevant to their interests. The clicked results are stored in the click through database and are treated as positive samples in training. The click through data, the extracted content concepts, and the extracted location concepts are used to create OMF profiles.

2.2 Update checker

The Updater checks at specific time intervals for unsettled data collections. For each such collection, the Updater performs a number of database updating tasks:

1) It adds the user key and his metadata on the Users-Attributes table or replaces already existing records for that specific user.
2) It checks for not yet discovered URLs and adds them to the Distinct URLs table.
3) It parses the URLs, their titles and semantic tags and adds possible new keywords to the Inverted Index table. The Inverted Index table maps each word with a unique number, a word-ID that is used as an identifier for all URLs that are related to this word.
4) It adds all distinct connections between word-IDs, URLs and users to the Bucket tables. In Buckets, we actually store a detailed decomposition of how words, URLs and users interact with each other. For each word, URL and user connection, a value is also stored that is later used from the Users Rank module to order pages. Currently, Searches gives higher values to keywords that are found to URL titles (semantic tags). The system uses a number of different Buckets to keep them reasonable in size.

2.3 Bucket creation for personalization

For each query posed, a dictionary is created to have a semantic similarity for each query that has been posed by the user with assigned user id. Once the similarity is found, only the related pages get stored in the dictionary in order to provide memory optimization. Finally based on the threshold, the pages get aligned by the means of relativity. Moreover the related URLs clicked by the user get stored in the respective search logger.

2.4. Behavioral future analysis

To predict the behavioral pattern of the users in social networking site, the numbers of users in the particular social media followed by their attributes are taken into account. Behavioral features like network bandwidth, message count, pair behavior. The attributes of the users can be,

1. The contact network between the users in the social media;
2. The number of shared friends between two users in the social media
3. The number of shared subscriptions between two users;
4. The number of shared subscribers between two users;
5. The number of shared favorite videos.

2.5 Learning algorithm

Two stages of algorithm,
1) Sparse coding
2) Dictionary update

2.6. Pattern discovery

Pattern discovery can be done using forward subset select based regression for the input matrix. The matrix can be obtained by greedy pursuit and convex relaxation. First is Greedy pursuit involves matching and orthogonal pursuit by selecting one atom per iteration. In Contrast, a stage wise orthogonal matching pursuit is proposed for selection of more than one atom per iteration. Second is convex relaxation which is slow compared to former one. The one approach used is basis pursuit to measure the sparseness. Another is least angle regression to overcome the computational complexity. To improve the performance further, the least angle regression and stage wise orthogonal pursuit are combined which emerge as new approach called stage wise least angle regression. This new approach is faster than the previous approach (stage wise orthogonal pursuit)

2.7 Ranking

When a user poses a query, the Search Query Analyzer parses and analyzes it. For each non-trivial word, the Analyzer finds the relative bucket and word-ID from the inverted index table. The Users Rank module produces an ordered list of the relative URLs based on their aggregated values. When we work with multiple word queries, we internally get a result set for each word inside the query. The result sets are then merged by multiplying their original single word score and by giving a disproportional benefit to results that are presented in multiple set pairs. This way single word matches are not excluded from the results but better matches combined with a good Users Rank are more probable to occur at the first spots. It is possible to augment the search query with restrictions about the participants.

3. TECHNIQUES USED

3.1 Classification

3.1.1 Support Vector Machine

Support vector machines are supervised technique used for classification and regression. We can apply linear classification method to nonlinear data. It separates the data into two categories. Future prediction is based on the target variable. Data need to be separated should be binary, if not it uses binary assessments on the data. Comparing to other classification methods, high accuracy of results are produced using SVM.

3.1.2 Bayesian classification

Bayesian classification deals with statistical in addition to supervised technique. The most usual form is in the terms of random variables. By determining the probabilities, the diagnostic and predictive problems. This classification mainly predicts the future events using the present events. In this system, ranking can be performed by using

1) Sim jaccard- number of common words to the total number of words.
2) Word count- number of similar words (text based similarity).
3) Link count- number of user clicked links.

3.1.3 Query Group

Query group is an ordered list of queries with the corresponding set of URLs that users have been clicked while posting their respective queries. This can be calculated by recording the set of all existing queries and comparing the current query with the existing set. Finally the relevance between the queries can be easily identified and plays a role in ranking.

3.1.4 Dynamic Query Grouping

In this technique, we are treating each user history as a singleton query group; these groups are then merged in an iterative fashion. Singleton query is just recording the query along with their clicks. For ranking these groups are then matched with the existing group to find out the relevance.

3.1.5 Query relevance

In the above mentioned methods, we are grouping the related queries but for ranking, a perfect relevant measure is needed to have an efficient ranking. This query relevance measure is used to establish a suitable relevance measure. This measuring is based on closely related queries and similar.

3.1.6 Query Reformulation and Fusion Graph

While posting a single query too many results gets retrieved. Hence ranking should be done to have an effective retrieval and to minimize the time in searching results. Hence each tuple is taken and a specific score is applied to it, in order to rank the page. These attribute value scores are combined according to the attribute weights to get a final ranking score for each tuple. Tuples with the top ranking scores are presented to the user first. Our ranking method is domain independent and requires no user feedback. Experimental results demonstrate that this ranking method can effectively capture a user's preferences.

3.1.7 Click through streaming

Search engine stores the click through data in triplet form (q, r, and c) where q-query
R-Ranking presented to the user
c- Clicked links or URLs.
Once the user posts the query, each query gets assigned with a unique user id. This gets stored in the query log along with query word and the presented ranking after an effective relevance measure Two Events based on link(before creating dataset) level0 and level 1.

Steps in Click through Streaming
1) Collecting Dataset
2) Dataset (Query id, Query, Time of Query, Item Rank and Clicked URL)

3.1.8 Sparse Representation

Sparse representation is used to represent the structure of the network in a compact way. This provides a feature extraction, pattern classification. The other method used for this techniques are support vector machine, relevance vector machine where it relates classification and ranking. The characteristics of SVM is
1) High dimensional input space.
2) Few irrelevant features.
3) Document vector are sparse.
4) Text categorization problems are linearly separable.

3.1.9 Dictionary Update

Dictionary update stage uses an approximated singular value decomposition stage (ASVD). ASVD computes matrix approximation, matrix rank, and null space of a matrix.

4. RESULTS AND DISCUSSION

User behavior information is analyzed using query reformulation and click graphs. In this paper, how such information can be used effectively for the task of organizing user search histories into query groups. More specifically, it combines the two graphs into a query fusion graph. The further show that the approach that is based on probabilistic random walks over the query fusion graph outperforms time-based and keyword similarity based approaches. Specifically in social networking site, we are analyzing the behavior of each user in social media and grouping the related user based on behavior and finally ranking the most shared, favorite videos.

5. CONCLUSION

We have proposed a new, bottom-up approach to study the web dynamics based on user's feedback. We are using peer to peer, bottom-up search engine that can provide search results by combining the user's preference regarding web pages. We have also discussed extensions to our approach that can provide personalized results. Finally, we have described how our approach can be integrated with Page Rank, providing an alternative version of Page Rank that combines two authorities: the link analysis and the users' preference. It is well known that actors in a network demonstrate correlated behaviors. In this work, the aim to predict the outcome of collective behavior given a social network and the behavioral information of some actors. In particular, explore scalable learning of collective behavior when millions of actors are involved in the network. Our approach follows a social-dimension based learning framework. Hence the edge-centric clustering extracts social dimensions to trace out the user behavior which is necessary for ranking based on user interest

REFERENCES

[1] A. Fuxman, P. Tsaparas, K. Achan, and R. Agrawal, "Using the wisdom of the crowds for keyword generation," in *WWW*, 2008.

[2] D. J. Hand, H. Mannila, and P. Smyth, *Principles of Data Mining*. Cambridge, MA: MIT Press, 2001.

[3] K. Avrachenkov, N. Litvak, D. Nemirovsky, and N. Osipova, "Monte carlo methods in PageRank computation: When one iteration is sufficient," *SIAM Jthenal on Numerical Analysis*, vol. 45, no. 2, pp. 890–904, 2007.

[4] R. Jones and K. L. Klinkner, "Beyond the session timeout: Automatic hierarchical segmentation of search topics in query logs," *in CIKM*, 2008.

[5] P. Boldi, F. Bonchi, C. Castillo, D. Donato, A. Gionis, and S. Vigna, "The query-flow graph: Model and applications," in *CIKM*, 2008.

[6] L. Tang and H. Liu, "Toward predicting collective behavior via social dimension extraction," *IEEE Intelligent Systems*, vol. 25, pages 19–25, 2010.

[7] "Relational learning via latent social dimensions," in KDD '09: *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. New York, NY, USA: ACM, 2009*, pages 817–826.

[8] M. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical Review E (Statistical, Nonlinear, and Soft Matter Physics)*, vol. 74, no. 3, 2006.

[9] L. Tang and H. Liu, "Scalable learning of collective behavior based on sparse social dimensions," in CIKM '09: *Proceeding of the 18th ACM conference on Information and knowledge management. New York, NY, USA: ACM*, 2009, pages 1107–1116.

[10] H. W. Lauw, J. C. Shafer, R. Agrawal, and A. Ntoulas, "Homophily in the digital world: A LiveJournal case study," *IEEE Internet Computing*, vol. 14, pages 15–23, 2010.