International Journal of Psychosocial Rehabilitation, Vol. 24, Issue 08, 2020
ISSN: 1475-7192

INTEREST CHANGES IN MULTI USER ENVIRONMENT FOR EFFICIENT HYBRID TAG RECOMMENDER SYSTEMS

[1]Fahad Iqbal T, [2]R.Gnanajeyaraman

ABSTRACT--*The modern world spends their maximum time in surfing internet and they perform their most activities through the internet. Adaptations of search performance predictors from the Information Retrieval field, and propose new predictors based on theories and models from Information Theory and Social Graph Theory. We show the instantiation of information-theoretical performance prediction methods on both rating and access log data, and the application of social-based predictors to social network structures. Recommendation performance prediction is a relevant problem per se, because of its potential application to many uses. Thus, we primarily evaluate the quality of the proposed solutions in terms of the correlation between the predicted and the observed performance on test data. This assessment requires a clear recommender evaluation methodology against which the predictions can be contrasted. Given that the evaluation of recommender systems is an open area to a significant extent, the thesis addresses the evaluation methodology as a part of the researched problem. We analyse how the variations in the evaluation procedure may alter the apparent behaviour of performance predictors, and we propose approaches to avoid misleading observations. In addition to the stand-alone assessment of the proposed predictors, we re-search the use of the predictive capability in the context of one of its common applications, namely the dynamic adjustment of recommendation methods and components. We research approaches where the combination leans towards the algorithm or the component that is predicted to perform best in each case, aiming to enhance the performance of the resulting dynamic configuration. The thesis reports positive empirical evidence confirming both a significant predictive power for the proposed methods in different experiments, and consistent improvements in the performance of dynamic recommenders employing the proposed predictors.*

Keywords--*Web Search, Web Mining, Web Inference Model, User Interest, Multi User Environment, HYBRID TAG RECOMMENDER*

I. INTRODUCTION

The web search is the most common activity performed by all the web users in their day to day process. Whatever they want to knew about they just approaches some search engines. The input query is being submitted to the search engine and the search engine return set of results according to the input query submitted. There are different type of search engines available in the market like textual search engines, image search engine and ontology search engine. We consider only the textual search engines and different search engines uses various meta data about the web url's. The web is a wast medium which is countless because every day new web sites are

[1] Research Scholar, Department of Innovative Informatics, Saveetha School of Engineering,Saveetha Institute of Medical and Technical Services, Chennai, fahad32in@gmail.com,

[2] Asso. Prof/ CSE, SBM College of Engg & Tech, Dindigul, r.gnanajeyaraman@gmail.com

Received: 21 Dec 2019 | Revised: 18 Jan 2020 | Accepted: 05 Feb 2020

launched with different domain names. The search engine maintains information about the web pages using the crawling process based on the meta data it has the user will be returned with different results.

On the other hand web mining is the process of extracting required information from the web or the web log data available. Sometimes the user may be intended to knew about the market strategy of a particular product and in order to perform such a task, it is necessary to keep track of market price of the different products at all the times. So that whenever the input query like the market strategy comes in the framework can identify or predict what will be the price of the product in the next time window. The web mining is not only for this kind of task but can be adapted for many things but we consider about the web search and the user interest prediction and identifying the change of user interest. How this can be performed is, by keep track of what the web user is visiting and what the actions performed by the user and how long the user visits the web page and other details[10]. By having these information as web log, some useful information can be performed to mine required information from the web log data.

The web inference model is one which infers set of information from the available web log data, here we discuss about the web search and with the details of search details of any single or multiple user, the framework can infer some useful information like what the people are surfing about and what the users are looking about in any point of time. For example, in a period of world cup cricket, we can see there are more click stream about visiting cricket pages than other news channels. Similarly there are many issues being visited at different time according to the changes in society and the events in society.

In real time, the user may search about many things in a day and all those things are considered as the topic and interest of the user[14]. From the identified interest, only few of them will be appear in every day or the user may be visit about the topic in each day search. Similarly the interest of a person may be changing one, because the user may be interested in using mobile phones some days and his interest will be changing to using i-phones or PDA or anything. This change of interest is most important and will be used to perform various market strategies which will be useful for marketing organizations and so on. So identifying the change of interest is very much important should be optimized in market analysis.

II. RELATED WORKS

Collaborative filtering (CF) techniques match people with similar preferences, or items with similar choice patterns from users, in order to make recommendations. Unlike CBF, CF methods aim to predict the utility of items for a particular user ac-cording to the items previously evaluated by other like minded users. These methods have the interesting property that no item descriptions are needed to provide rec-ommendations, since the methods merely exploit information about past ratings. Compared to CBF approaches, CF also has the salient advantage that a user may benefit from other people‟s experience, thereby being exposed to potentially novel recommendations beyond her own experience (Adomavicius and Tuzhilin, 2005).

In this section we focus on those CF techniques based on explicit numeric rat-ings, which are the most common in the literature. For additional references, see (Desrosiers and Karypis, 2011; Koren and Bell, 2011) and (Adomavicius and Tuz-hilin, 2005). Most of our discussion nonetheless applies to log-based recommenders alike.

In fact, as we shall show in the next section, most of the rating-based tech-niques can be used when no ratings are available (although the equivalence intro-duces additional assumptions).

In general, CF approaches are commonly classified into two main categories: model-based and memory-based. Model-based approaches build statistical models of user/item rating patterns to provide automatic rating predictions. Some ap-proaches learn such models by performing some form of dimensionality reduction in order to uncover latent factors between users and items, e.g. by such techniques as Singular Value Decomposition (SVD) for matrix factorisation (Billsus and Pazzani, 1998; Koren et al., 2009), probabilistic Latent Semantic Analysis (pLSA), or Latent Dirichlet Allocation (LDA) (Hofmann, 2003; Blei et al., 2003). Other approaches use probabilistic models where the recommendation task is modelled by user and item probability distributions (Wang et al., 2006b; Wang et al., 2008a), e.g. by learning a probabilistic model with a maximum entropy estimation (Pavlov et al., 2004; Zitnick and Kanade, 2004), Bayesian networks (Breese et al., 1998), and Boltzmann machines (Salakhutdinov et al., 2007). A graph-based model that exploits positive and negative preference data is proposed in (Clements et al., 2009). Besides, other Machine Learn-ing techniques have also been proposed, such as artificial neural networks (Billsus and Pazzani, 1998) and clustering strategies (Kohrs and Merialdo, 1999; Cantador and Castells, 2006).

Memory-based approaches, on the other hand, make rating predictions based on the entire rating collection (Adomavicius and Tuzhilin, 2005; Desrosiers and Karypis, 2011). These approaches can be user- and item-based strategies. User-based strate-gies are built on the principle that a particular user‟s rating records are not equally useful to all other users as input for providing personal item suggestions (Herlocker et al., 2002). Central aspects to these algorithms are thus a) how to identify which neighbours form the best basis to generate item recommendations for the target user, and b) how to properly make use of the information provided by them. Typically, neighbourhood identification is based on selecting those users who are more similar to the target user according to a similarity metric (Desrosiers and Karypis, 2011). The similarity between two users is generally computed by a) finding a set of items that both users have interacted with, and b) examining to what degree the users displayed similar behaviors (e.g. rating, browsing and purchasing patterns) on these items. This basic approach can be complemented with alternative comparisons of virtually any user feature a system has access to, such as personal demographic and social network data. It is also common practice to set a maximum number of neighbours (or a minimum similarity threshold) to restrict the neighbourhood size either for computa-tional efficiency, or in order to avoid noisy users who are not similar enough. Once the target user‟s neighbours are selected, the more similar a neighbour is to the user, the more her preferences are taken into account as input to produce recommenda-tions. For instance, a common user-based approach consists of predicting the rele-vance of an item for the target user by a linear combination of her neighbours‟ rat-ings, weighted by the similarity between the target user and such neighbours.

In the following equations we present two versions of a user-based CF tech-nique; in the first one rating deviations from the user‟s and neighbour‟s rating means are considered (Resnick et al., 1994), whereas in the second one the raw scores given by each neighbour are used (Aggarwal et al., 1999; Shardanand and Maes, 1995).

In (Burke, 2002) a detailed taxonomy of hybrid recommender systems is pre-sented, classifying existing approaches into the following types:

 Cascade: the recommendation is performed as a sequential process in such a way that one recommender refines the recommendations given by the other.

 Feature augmentation: the output from one recommender is used as an addi-tional input feature for other recommender.

 Feature combination: the features used by different recommenders are inte-grated and combined into a single data source, which is exploited by a single recommender.

 Meta-level: the model generated by one of the recommenders is used as the input for other recommender. As stated in (Burke, 2002): "this differs from feature augmentation: in an augmentation hybrid, we use a learned model to generate features for input to a second algorithm; in a meta-level hybrid, the entire model becomes the input."

 Mixed: recommendations from several recommenders are available, and are presented together at the same time by means of certain ranking or combina-tion strategy.

 Weighted: the scores provided by the recommenders are aggregated using a linear combination or a voting scheme.

 Switching: a special case of the previous type considering binary weights, in such a way that one recommender is turned on and the others are turned off.

The use of a specific type of hybrid recommendation method depends on the fi-nal application, but, more importantly, on the type of recommenders being com-bined. Indeed, Burke (2002) presents an analysis of the possible hybrids, their limits and incompatibilities, based on a representative subset of the recommendation tech-niques available nowadays. Moreover, the author notes that some combinations turn out to be redundant because of the symmetry in the hybridisation process for some of the techniques listed above: weighted, mixed, switching, and feature combination. Incompatible combinations arise for the feature combination and meta-level tech-niques, where in some situations one of the recommenders is not able to use the model or the features generated by the other recommender.

Efficient Multiple-Click Models in Web Search [4], presents a click model which logs the clicks and then contain the submitted query, a ranked list of returned documents, whether each of them is clicked or not, and other information that might be useful. Click models learn from user clicks to help understand and incorporate users' implicit feedback. And they follow a probabilistic approach which treats user clicks as random events, and the goal is to design generative models which are able to approximate underlying probabilities of clicks with high accuracy.

SimRank: A Page Rank approach based on similarity measure [5], propose a new page rank algorithm based on similarity measure from the vector space model, called SimRank, to score web pages. Firstly, a new similarity measure used to compute the similarity of pages and apply it to partition a web database into several web social networks (WSNs). Secondly, they improve the traditional Page Rank algorithm by taking into account the relevance of page to a given query. Thirdly, we design an efficient web crawler to download the web data. And finally, experimental studies are performed to evaluate the time efficiency and scoring accuracy of SimRank with other approaches.

All the above discussed approach has the problem of false identification of interests and has low accuracy in identifying the user interest.

III. PROPOSED METHOD:

The proposed method has various stages namely User Action Handler, User Interest Prediction, Web Inference Model, Query Processor and so on. We discuss each of the functional components in detail in this section.

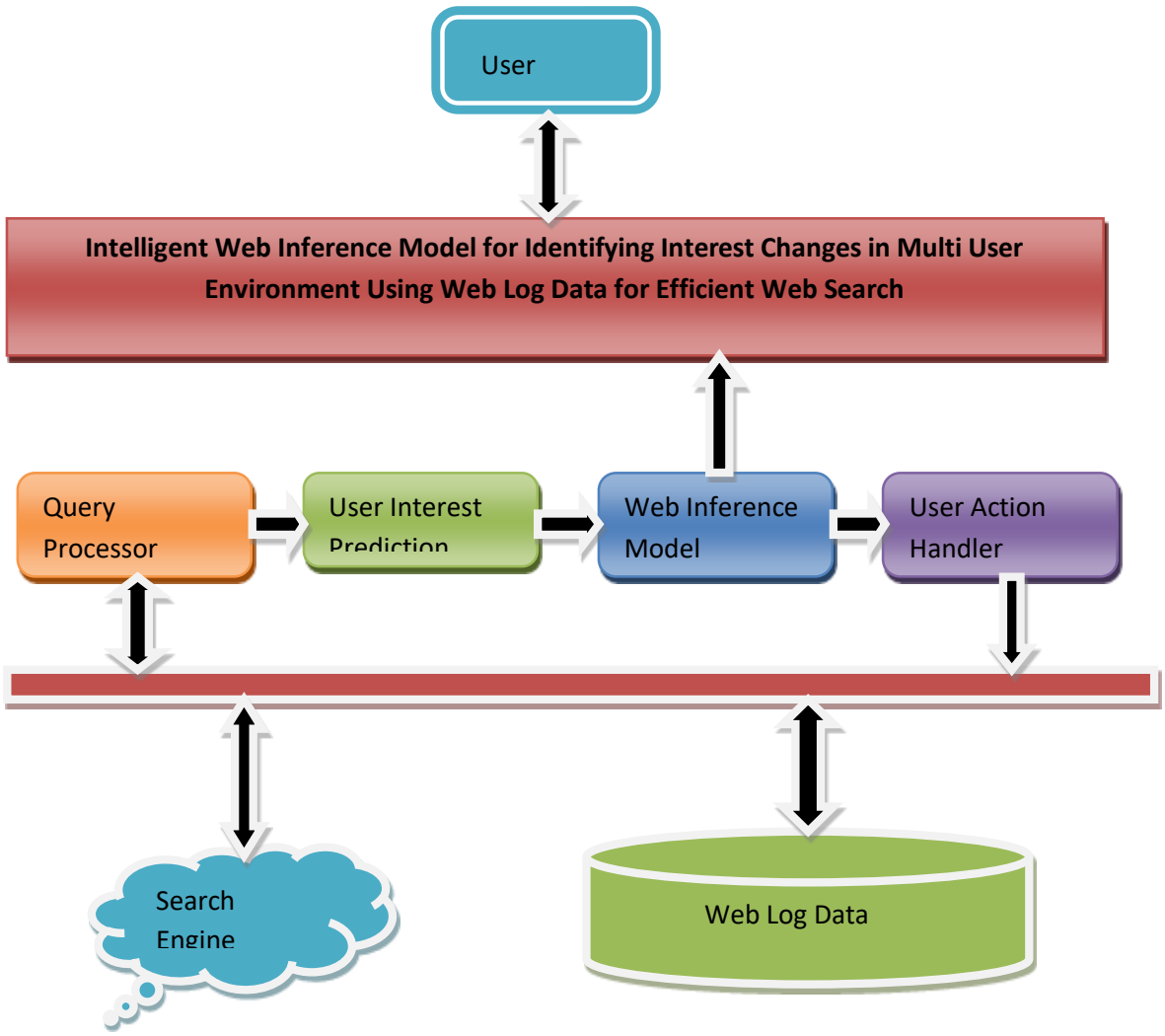


Figure 1: Proposed System Architecture.

3.1 Query Processor:

The query processor performs the operation of receiving the input query submitted by the user and submits to the standard web search engine. The query processor wait for the result to be returned by the web search engine and returns the results to the user. Between these process there are many stage of computing is performed. The

query processor invokes the search engine url by using a specific program and constructs a url with the input query. The input query is trimmed and each blank space is added with the '+' symbol to construct the query url. The constructed query url will be connected and the http result returned by the url will be received by the query processor. The received result will be handover to the rest of the components of the model to produce most effective results.

Algorithm:

Input: Search Query sq.

Output: Search Result SR.

Step1: start

Step2: read input query sq.

Step3: split sq into number of distinct words.

Query words Qw = $\int(\forall('') \in sq) \uparrow Sq(T)$

Step4: for each query word Q_i from QW

Add search url with Q_i and '+'.

QUrl = $\int_{i=1}^{size(QW)} \sum Qi + ' + '$

End

Step5: Connect to the url

Step6: SR = Receive result from the url.

Step7: stop.

3.2 User Action Handler:

This performs the monitoring of user behaviors performed on web surfing. The process monitors the query being submitted, the page being visited, the time spent on the web page, the actions being performed by the user on the web page, the previous and next page or navigation page are identified. The method identifies the topic of the web page by extracting the nouns from the page being visited. Based on the terms being visited and using the wordnet dictionary the topic of the web page is identified. The monitored and tracked user behavior information is stored to the web log which specifies the current visit history which could be used to identify the interest transition.

Algorithm:

Input: Web Page Wp,

Output: Web log Wl.

Step1: start

Step2: Extract the Url of the web page.

Wurl = Page being visited.

Step3: Compute time spent on the web page.

Visit Time Vt= (Time loaded – Time Navigated).

Step4: Identify Previous and next Urls

Purl = Previous URl

NUrl = Next URL

Step5: Identify Actions Performed Like save, copy, print, bookmark.

Ac = $\int \sum Actionsonpage.$

Step6: construct log L = User, Wurl, Purl, Nurl, Vt, Ac, Topic.

Step7: stop.

3.3User Interest Prediction:

The interest of the user will be kept on changing and in this stage we identify the current interest of the user. The user interest is identified by performing various activities on the page visited by the user. The page content visited by the user is extracted and from the page content, the unnecessary words are removed. From the remaining words, the stemming operation is performed and from then the pure nouns are identified. With the available pure nouns and word netsynsets, we collect set of related words and computes the topical similarity measure to identify the topic which represent the interest of the user.

Algorithm:

Input: Web Content Wc, WordnetWn

Output: Topic Tc.

Step1: start

Step2: Read page content Wc.

Step3: Split Wc into distinct Terms.

Ts = $\int Split(Wc,'')$

Step4: for each term T_i from Ts

If SL$\int_{i=1}^{size(Sl)} if Sl(i) == Tithen$

Ts = Ts – T_i.

End

End

Step5: for each term T_i from Ts

Perform Stemming operation.

End

Step6: for each category Ci

Load Related Terms RT.

Compute Topical Similarity Measure TSM.

TSM = $\int_{i=1}^{size(RT)} \frac{\sum Ts(i) \in RT}{size(RT)}$

End

Step7: Choose maximum similarity category Tc.

Step8: stop.

3.4 Web Inference Model:

The web inference model performs the change of interest occurred in multi user search history. From the web log available , for each user we identify the interest of user at different time window and the change of interest is also identified. From the identified interest, the inference model identifies the probable future interest of user and

groups the similar users with the same interest and collects the pages visited by them to provide more related results to the user.

Algorithm:

Input: Web log Wl.

Output: Result WR.

Step1: start

Step2: for each time window Ti

Identify the interest of the user Int = $\int_{i=1}^{size(Interest)} Interest \in \forall(Ti)$

End

Step3: Identify similar interested users

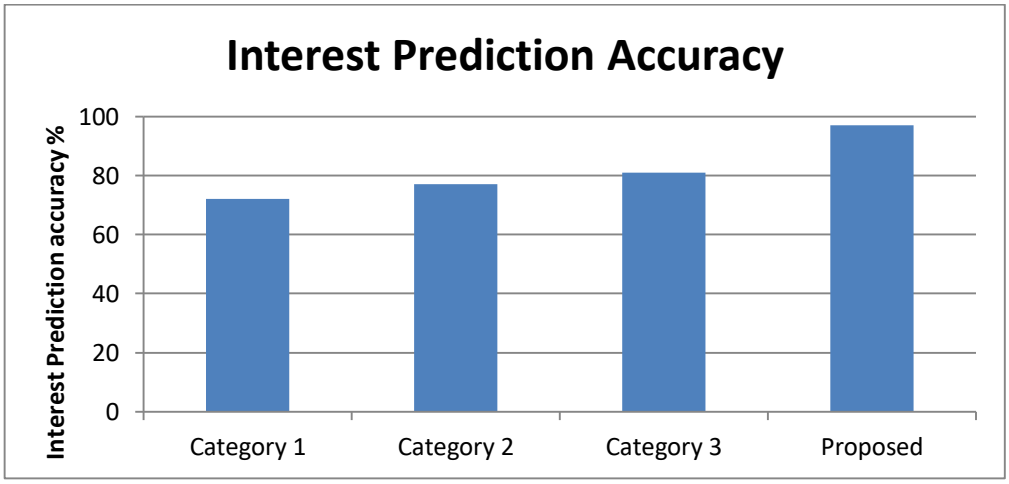
Su = $\int_{i=1}^{size(Users)} \sum Ui(Int) == Int$

Step4: Collect all the pages visited by the users SU and add to the web result.

Step5: stop.

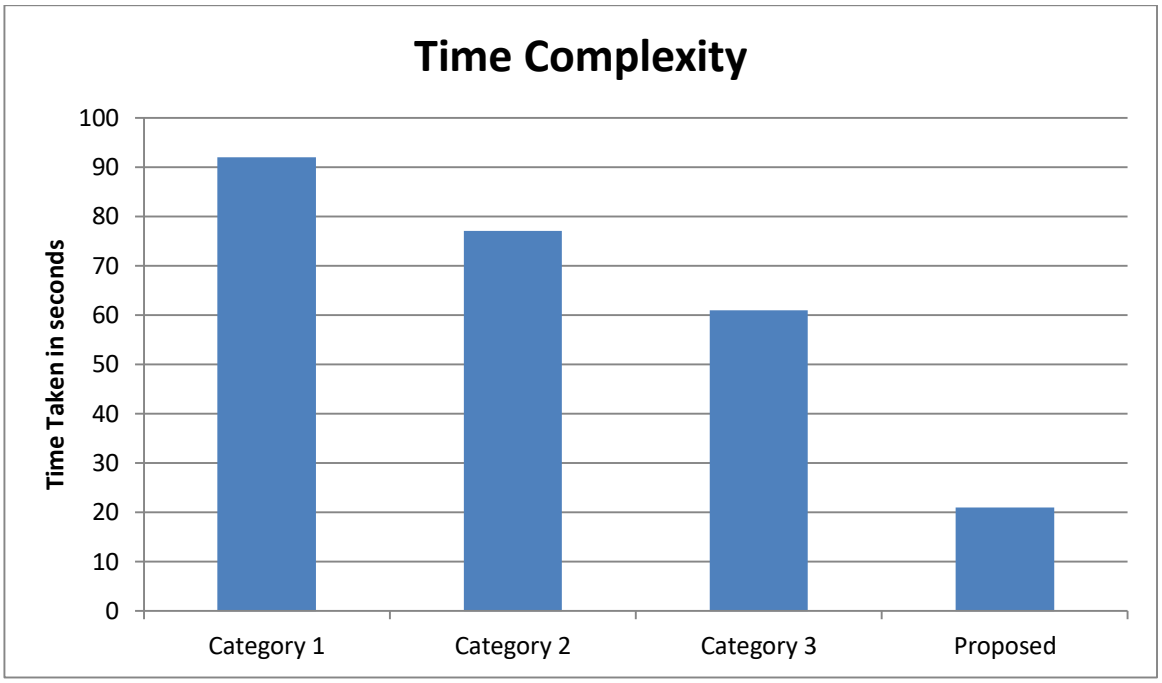
IV. RESULTS AND DISCUSSION

The proposed multi user web inference model for user interest prediction and identifying the interest changes has been implemented and tested for its effectiveness. The proposed method has produced efficient results in all the factors of web mining.



Graph 1: Comparison of interest prediction accuracy.

The Graph1 shows the comparison of interest prediction accuracy produced by different methods and it shows clearly that the method has produced higher accuracy in interest prediction.



Graph2: Comparison of time complexity of different methods.

The Graph2. Shows the comparison of time complexity produced by different methods and it shows clearly that the proposed method has produced less time complexity than others.

V. CONCLUSION

the adaptation of performance predictors from Information Retrieval – mainly the query clarity predictor, which captures the ambiguity of a query with respect to a given document collection. We have defined several language models according to various probability spaces to capture different aspects of the users and items involved in rec-ommendation tasks. In this context, we have proposed and evaluated novel ap-proaches drawing from Information Theory and Social Graph Theory for different recommender input spaces, using information-theoretic properties of the user‟s prefer-ences and graph metrics such as PageRank over the user‟s social network.

Moreover, since we aimed to predict the performance of a particular recom-mender system, we required a clear recommender evaluation methodology against which performance predictions can be constrasted. Hence, in this thesis we ad-dressed the evaluation methodology as part of the problem, where we have identified statistical biases in the recommendation evaluation – namely the sparsity and popu-larity biases – which may distort the performance assessments, and therefore may confound the apparent power of performance prediction methods. We have analysed in depth the effect of such biases, and have proposed two experimental designs that are able to neutralise the popularity bias: a percentile-based approach and a uniform-test approach. The systematic analysis of the evaluation methodologies and the new proposed variants have enabled a more complete and precise assessment of the ef-fectiveness of our performance prediction methods.

On the other hand, we have exploited the proposed performance prediction methods in two applications where they are used to dynamically weight different components of a recommender system, namely the dynamic adjustment of weighted hybrid recommendations, and the dynamic weighting of neighbours‟ preferences in user-based collaborative filtering. Through a series of empirical experiments on sev-eral datasets and experimental

designs, we have found a correspondence between the predictive power of our performance predictors and performance enhancements in the two tested applications. The proposed method has produced efficient results in time complexity and has achieved more interest prediction accuracy.

REFERENCES

1. Adams, R. A. (2007). Music Recommendation using Collaborative Filtering with Similarity Fusion. Master‟s thesis, Department of Computer Science, The University of York.
2. Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of re-commender systems: a survey of the state-of-the-art and possible extensions. IEEE Transactions on Knowledge and Data Engineering, 17(6):734–749.
3. Adomavicius, G., Tuzhilin, A., Berkovsky, S., De Luca, E. W., and Said, A. (2010). Context-awareness in recommender systems: research workshop and movie recommendation challenge. In Proceedings of the fourth ACM conference on Recommender systems, RecSys „10, pages 385–386, New York, NY, USA. ACM.
4. Agrawal, R., Gollapudi, S., Halverson, A., and Ieong, S. (2009). Diversifying search results. In Proceedings of the Second ACM International Conference on Web Search and Data Mining, WSDM „09, pages 5–14, New York, NY, USA. ACM.
5. Aizawa, A. (2003). An information-theoretic perspective of tf-idf measures. In-formation Processing & Management, 39(1):45–65.
6. Alvarez, M. M., Yahyaei, S., and Roelleke, T. (2012). Semi-automatic document classification: Exploiting document difficulty. In Baeza Yates, R. A., de Vries, A. P., Zaragoza, H., Cambazoglu, B. B., Murdock, V., Lempel, R., Silvestri, F., Baeza Yates, R. A., de Vries, A. P., Zaragoza, H., Cambazoglu, B. B., Murdock, V., Lempel, R., and Silvestri, F., editors, ECIR, volume 7224 of Lecture Notes in Computer Science, pages 468–471. Springer.
7. Bao, X., Bergman, L., and Thompson, R. (2009). Stacking recommendation en-gines with additional meta-features. In Proceedings of the third ACM conference on Recom-mender systems, RecSys „09, pages 109–116, New York, NY, USA. ACM.
8. Barbieri, N., Costa, G., Manco, G., and Ortale, R. (2011). Modeling item selec-tion and relevance for accurate recommendations: a bayesian approach. In Proceedings of the fifth ACM conference on Recommender systems, RecSys „11, pages 21–28, New York, NY, USA. ACM.
9. Samy Bengio, Jason Weston, and David Grangier. 2010. Label embedding trees for large multi-class tasks. In International Conference on Neural Information Processing Systems. 163–171.
10. Dr.M G Gireeshan.”X Ray Viewer Smart System”,International Journal Of Pharmacy & Technology [Issn: 0975-766x], Ijpt| July-2015 | Vol. 7 | Issue No.1 | 8412-8414
11. Alina Beygelzimer, John Langford, and Pradeep Ravikumar. 2007. Multiclass classification with filter trees. Gynecologic Oncology 105, 2 (2007), 312–320.
12. Pablo Castells, SaÃžl Vargas, and Jun Wang. 2011. Novelty and Diversity Metrics for Recommender Systems: Choice, Discovery and Relevance. In Proceedings of International Workshop on Diversity in Document Retrieval (DDR (2011), 29–37.

Received: 21 Dec 2019 | Revised: 18 Jan 2020 | Accepted: 05 Feb 2020

13. Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishi Aradhye, Glen Anderson, Greg Corrado, Wei Chai, Mustafa Ispir, et al. 2016. Wide & deep learning for recommender systems. In Proceedings of the 1st Workshop on Deep Learning for Recommender Systems. ACM, 7–10.
14. Gireeshan M.G.,"Air quality index prediction using meteorological data using featured based weighted xgboost"International Journal of Innovative Technology and Exploring Engineering,ISSN: 2278-3075, Volume-8, Issue-11S, September 2019
15. Paul Covington, Jay Adams, and Emre Sargin. 2016. Deep Neural Networks for YouTube Recommendations. In ACM Conference on Recommender Systems. 191–198.
16. Robin Devooght and Hugues Bersini. 2016. Collaborative Filtering wih Recurrent Neural Networks. (2016).
17. Zeno Gantner, Steffen Rendle, Christoph Freudenthaler, and Lars SchmidtThieme.2011. MyMediaLite: A free recommender system library. In Proceedings of the fifth ACM conference on Recommender systems. ACM, 305–308.
18. F. Maxwell Harper and Joseph A. Konstan. 2015. The MovieLens Datasets: History and Context. ACM. 19 pages. [10] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In Proceedings of the 26th International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, 173–182.
19. Xiangnan He, Hanwang Zhang, Min-Yen Kan, and Tat-Seng Chua. 2016. Fast matrix factorization for online recommendation with implicit feedback. In Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval. ACM, 549–558.
20. Shuai Zhang, Lina Yao, and Aixin Sun. 2017. Deep Learning based Recommender System: A Survey and New Perspectives. (2017).
21. Guorui Zhou, Chengru Song, Xiaoqiang Zhu, Xiao Ma, Yanghui Yan, Xingya Dai, Han Zhu, Junqi Jin, Han Li, and Kun Gai. 2017. Deep Interest Network for Click-Through Rate Prediction. (2017).

Received: 21 Dec 2019 | Revised: 18 Jan 2020 | Accepted: 05 Feb 2020