ISSN: 2278 – 7798
International Journal of Science, Engineering and Technology Research (IJSETR)
Volume 5, Issue 9, September 2016

Implementing Hierarchical structure for Subtasks of a task in Collaborative Environment

Mr Yogesh M.Here, Mr.H.A.Tirmare

Abstract— **Web search engine is widely used to find a required data, among the huge amount of information in a minimum amount of time. Now it is possible to find an advisor on the certain topic using web surfing data. Advisors are not expert on given query but they possess the desired piece of information related to given query. Hence suggesting advisor to the new learner saves the time and efficiency of new learner. We are finding expert using browsing history of each user. This browsing history has all information associated with it such as search topic, time etc. This web surfing data of each user is clustered according to the topics and then applying discriminative infinite hidden markov model in order to find advisor on given topic and find a hierarchy of sub-topics associated with given topic.**

***Index Terms*— Advisor search, Gaussian mixture model, Infinite hidden markov model, Precision, Semantic URL.**

I. Introduction

Within an information retrieval system, a single query is executed in order to fetch the all relevant data associated with it. This information retrieval system have three parts:First part is responsible for gathering information and creating information in the form of an index, the second part is responsible for formulating queries consisting of keywords and the third part is responsible for matching the query to the created indexes.

Finding Advisor [5] on given topic is a difficult task because of the vast amount of information available on the internet. Advisor is founded in order to save time and effort of another person. For example Jerry Studied about File handling in C programming using surfing web. But now Loy also wanted to learn File handling in c programming. If Loy also started learning using surfing web, then he has to learn from basics of C Programming. In order to save the time and effort of Loy, we will suggest Jerry as an advisor .Then Loy will start learning from Jerry. In this example, it is not that Jerry is an expert on given topic but he possesses some good amount of information on that. This method has one advantage that Jerry will share his experiences with Loy in learning and also suggest good learning material.

In order, to found an expert on given topic, we are using the semantic structure of the web. We will collect the browsing history of various websites visited. The Semantic structure will give the information about particular page visited on the web. For example, if I have searched on Google for abstraction in java programming then various websites will appear giving information about java abstraction. Suppose that I have selected website tutorialspoint.com, then the semantic structure of that page is like http://www.tutorialspoint.com/java/java_abstraction.htm . We can clearly see that given URL consist of java abstraction as a keyword. Hence every topic you searched on the internet has semantic URL. From the URL visited by the user, we will get knowledge about a web user.

Here we have compared three different methods of advisor search and showed that the sub-task based method of advisor search is better than the other two methods discussed in this paper.

II. Related work

To find experts within an organization is the point of study for many researchers. This aim of researchers leads to a class of search engines called expert finders. McDonald and Ackerman [6] proposed two point formula for finding an expert. First is 'who is an expert on a given topic' and the second is 'what does X know'.

Krisztian Balog, Leif Azzopardi and Maarten de Rijke [2] have proposed Expert finding system in enterprise corpora. They proposed two methods for finding an expert on given topic. In first method, the expert is founded based on the given documents associated with it. Whereas the second method finds documents associated with the given topic and the finds advisor based on that. They also showed that second method of finding an expert is better than the first one.

These expert finding system not only implemented on text documents but later on it is implemented on web information services. Xiasong Liu ,W.Bruce Croft and Matthew Koll [8] Proposed a paper to find experts in Community-based Question Answering Services. These information services that a big network of experts to answer the others questions. There are many questions are asked each day but some questions remain unanswered for a long period of time. In order to solve the given question very quickly, they have proposed a new method. In this question is put forward to the right person who have desired knowledge to solve it.

Witold Abramowicz, Elzbieta Bukowska ,Monika Kaczmarek and Monika Starzecka [7] proposed a new method to exactly retrieve expert from eXtraSpec System. They also show that how information is identified exactly and efficiently. They showed that Query Expansion method proposed by them increases precision value and has good scalability and efficiency than fully fledged semantics system.

Hui Fang and Cheng Xiang Zhai [9] proposed two methods such as Candidate generation model and Topic generation model for expert finding using Probabilistic model. Along with this they also proposed mixture model for modeling the candidate mentions and topic expansion for modeling topic document relationship. Also, they proposed email based candidate prior method which provides better estimation that candidate is an expert.

Yi Fang , Luo Si and Aditya P. Mathur used Discriminative models for expert Search rather than language-based models. They also showed that discriminative models have a low asymptotic error rate and make fewer model assumptions than their generative counterparts.

Dawit YIMAM-SEID, Alfred KOBSA [4] proposed domain analysis approach for finding an expert. They have discussed seven factors to find expertise in the given area. In first factor expertise is identified using various evidence. Evidence include various documents associated with the organization. In second factor expertise indicator is extracted, once all sources and information is gathered and identified. The third factor is about expertise models which describe individual person expertise, skills, and area of specialization. The fourth factor is about Query mechanism in which expert is retrieved based on the query is passed. Given query retrieve all information associated with such as users activities, behavior etc. The fifth factor is about Matching operations in which information in documents is matched with keywords given in the query. These keywords are very helpful to exactly extract information associated with the given query and find expert exactly. The sixth factor is about output representation in which sources are provided according to experts are mentioned. The seventh factor is about learning operations in which users preference and domain [3] knowledge as well as users feedback is gathered. In which Users are allowed to bring new experts into the system.

III. ADVISOR SEARCH

Finding Advisor in the company or in any organization with the skills known to him is very important for the completion of the project successfully. In traditional approach of advisor search database is maintained in order to store the skill and information of each individual of the company. In order to this lot of manual work is performed because all data is written to the database one after other. But in this new method of advisor search database is automatically constructed according to the user's web surfing behavior. The main difference between the traditional method of advisor search and this new method of finding the advisor is that in the traditional method, the advisor is founded on documents and this new method is applied to session generated by the user while web surfing. Using following Steps advisor is searched –

1) Each individual users browsing history and cookies information is collected
2) Each user has some sequence of sessions searched over the web
3) Combine all sessions of each user and cluster them according to specific task
4) Partition sessions from each cluster into set of sub-tasks where each subtask is significantly more subset of sessions from cluster
5) Compute the association weight between the query given by the user for which the advisor is to be searched and each subtask in order to give the correct ranking of advisors.

In order to find out the hierarchical structure [16] of each subtask present in all task, we are applying the discriminative infinite hidden markov model.For example When I will give topic "C sharp " then automatically all subtopics associated with such as abstraction,inheritance and multithreading are shown in a hierarchical structure way.This Hierarchical structure is shown for the given example in figure 1

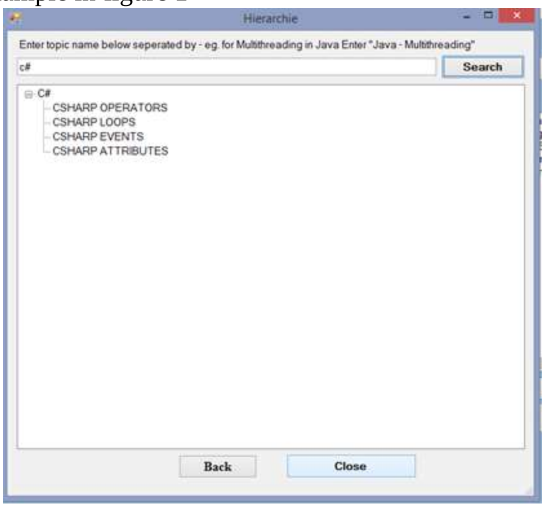

Figure -1: Hierarchical Structure for subtask in each task

We have implemented advisor search scheme on three different levels such as session-based , task-based and subtask based which is shown in Figure 2. In session based scheme advisor is searched on the given session generated by the user whereas in task-based advisor scheme advisor is searched on a particular topic from all task present in the database. Whereas last scheme of advisor search is a best among all three which gives the advisor on fine-grained level. For example, if anyone wants the advisor for topic " Java Abstraction " then we will use subtask based advisor scheme. Whereas if anyone wants the advisor on a particular subject or on a general topic such as "Java Programming" then we will use task based advisor search scheme.

ISSN: 2278 – 7798
International Journal of Science, Engineering and Technology Research (IJSETR)
Volume 5, Issue 9, September 2016

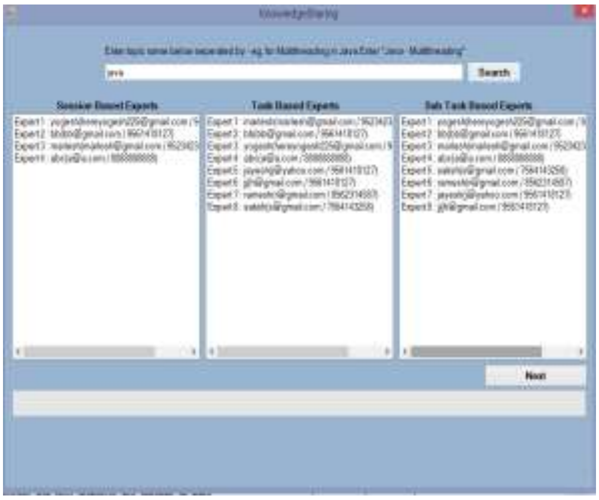

Figure -2: Comparison of Three schemes of advisor search

IV. Experimental Results

4.1 Running Time of D-iHMM

Running time for d-iHMM [1] is shown in Table 1 As the number of sessions per task increases the amount of time required to find expert based on discriminative infinite hidden markov model also increases. The Table shows as sessions increases then time also increases.

Table -1: Running Time of DiHMM

Time (s)	Number of sessions per task
2	20
4	45
6	60
8	75
10	95

Following Chart shows the graph related to running time of discriminative infinite hidden markov model.

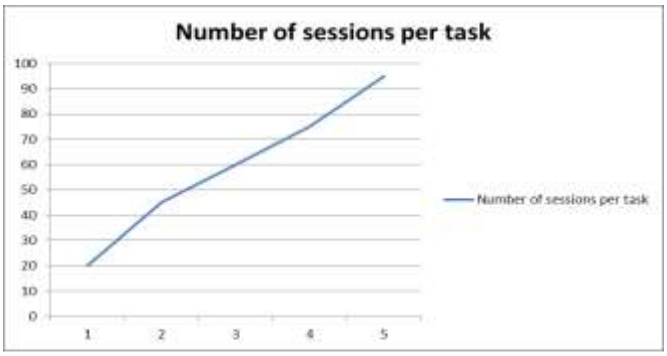

Chart -1: Running Time of Discriminative infinite hidden markov model for number of sessions in a task

4.2 Precision

In the field of information retrieval, precision is the result of relevancy that is how documents relevant to the given query:

$$\textit{Precision} = \frac{\textit{Retrieved items that are relevant}}{\textit{All Retrieved items}}$$

Table -2: Precision Values for Input Query

Chart for the entries in Table 2 is shown in figure.

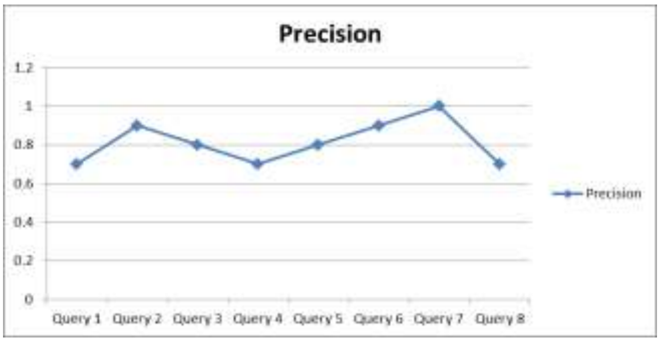

Chart -2: Precision values for input queries

4.3 Comparison of Three Advisor Search scheme

In order to compare three strategies given for advisor search, we are using normalized discount cumulative gain (NDCG). For the different level of NDCG results are shown in table 3 for dataset 1. Column chart of results shown in Table 3 are given in figure 3

Table 3: Comparison of Three advisor search schemes

Method	Level 1	Level 2	Level 3
Sub task based	0.780	0.880	0.910
Session based	0.697	0.800	0.850
Task based	0.686	0.770	0.810

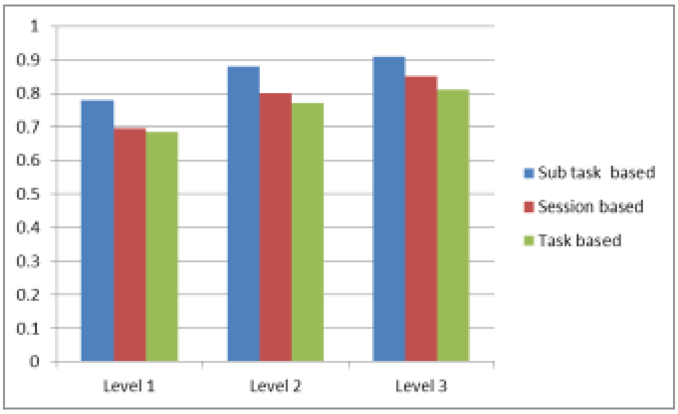

Figure -3: Comparison of Three schemes of advisor search

ISSN: 2278 – 7798
International Journal of Science, Engineering and Technology Research (IJSETR)
Volume 5, Issue 9, September 2016

4.4 Performance comparison between Discriminative infinite hidden markov model and infinite hidden markov model

On the number of experts retrieved for subtask associated with each task, comparison is done between d-iHMM and iHMM. Discriminative infinite hidden markov model shows results at subtask level than infinite hidden markov model. The following figure shows the comparison between two.

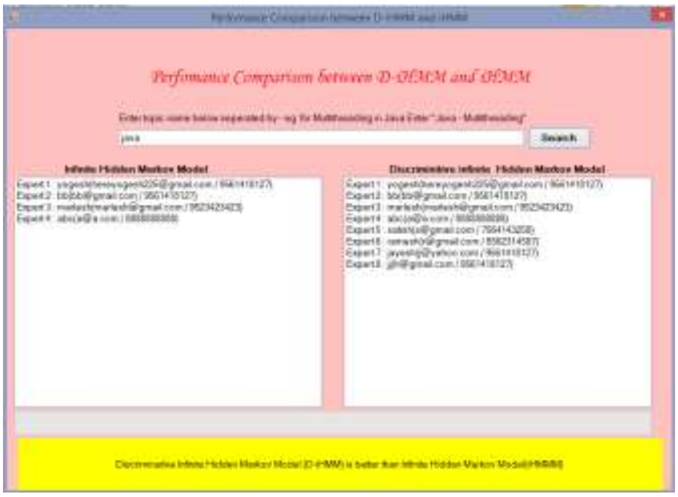

Figure -4: Performance comparison of D-iHMM and iHMM

V. Conclusion

We have shown hierarchical structure for subtask present in each task. Also, we compared three different schemes for advisor search and showed that subtask based method of advisor search is better than other two. While we are finding advisor on web surfing data that is browsing history generated by each user which leads to loss of privacy. In future, we can implement on privacy issue by using more powerful authentication.

Acknowledgment

Our sincere thanks to Department of Technology, Shivaji University, Kolhapur for supporting this work.

REFERENCES

[1] Ziyu Guan, Shengqi Yang, Huan Sun, Mudhakar Srivatsa, and Xifeng Yan "Fine-Grained Knowledge Sharing in Collaborative Environments". IEEE transactions on Knowledge and data Engineering, vol.27,No. 8,August 2015, pp.2163-2174.

[2] K. Balog, L. Azzopardi, and M. de Rijke, "Formal models for expert finding in enterprise corpora," in Proc. 29th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2006, pp. 43–50.

[3] D.Yimam. Expert finding systems for organizations: Domain analysis and the demoir approach. In ECSCW 999 Workshop: Beyond Knowledge Management: Managing Expertise, pages 276–283, New York, NY, USA, 1996. ACM Press

[4] D.Yimam-Seid and A. Kobsa. Expert finding systems for organizations. Sharing Expertise: Beyond Knowledge Management, 2003

[5] A. Mockus and J. D. Herbsleb. Expertise browser: a quantitative approach to identifying expertise. In ICSE '02: Proceedings of the 24th International Conference on Software Engineering, pages 503–512. ACM Press, 2002.

[6] D. W.McDonald and M. S. Ackerman. Expertise recommender: a flexible recommendation system and architecture.In CSCW '00: Proceedings of the 2000 ACM conference on Computer supported cooperative work, pages 231–240. ACM Press, 2000.

[7] Witold Abramowicz, Elzbieta Bukowska,Monika Kaczmarek and Monika Starzecka "Semantic enabled Efficient and Scalable Retrieval of Experts".

[8] X. Liu, W. B. Croft, and M. Koll, "Finding experts in community based question-answering services," in Proc. 14th ACM Int. Conf.Inf. Knowl. Manage., 2005, pp. 315–316.

[9] Hui Fang and ChengXiang Zhai "Probabilistic Models for Expert Finding".29th European Conference on IR Research ,ECIR 2007,Rome,Italy,April 2-5,2007,pp.418-430.

[10] Krisztian Balog, Leif Azzopardi ,Maarten de Rijke "A language modeling framework for expert finding".www.elsevier.com/locate/infoproman Information processing and management,June 2008.

[11] C. Rasmussen, "The infinite Gaussian mixture model," in Proc. Adv. Neural Inf. Process. Syst., 2000, pp. 554–560.

[12] Y. Zhao, G. Karypis, and U. Fayyad, "Hierarchical clustering algorithms for document datasets," Data Mining Knowl. Discovery,vol. 10, no. 2, pp. 141–168, 2005.

[13] R. Kumar and A. Tomkins, "A characterization of online browsing behavior," in Proc. 19th Int. Conf. World Wide Web, 2010, pp. 561–570.

[14] N. Craswell, A. de Vries, and I. Soboroff. Overview of the trec-2005 enterprise track. In *TREC-13*, 2005

[15] T. H. Davenport and L. Prusak. Working Knowledge: How Organizations Manage What They Know. Harvard Business School Press, Boston, MA, 1998.

[16] Kaingade, Rasika M., and Hemant A. Tirmare. "Personalization of Web Search based on privacy protected and auto-constructed user profile." *Advances in Computing, Communications and Informatics (ICACCI), 2015 International Conference on*. IEEE, 2015.

BIOGRAPHIES

Mr. Yogesh M.Here is a student in the master of Computer Science and Technology program at Department of Technology, Shivaji University, Kolhapur. He is interested in domains like data mining and computer security.

Mr. H. A. Tirmare is an Assistant Professor in the Computer Science and Technology department, Department of Technology, Shivaji University, Kolhapur. His domains of interest include but are not limited to computer networks, web security, operating systems, data structures.