Prediction of users' next question in a QA system using Association rule mining technique

Prof. Vidyashree K.P, Amulya R, Ashrith C, Kavyashree S, H.V Tejas Taunad

Dept of Information Science and Engineering, Vidyavardhaka College of Engineering

ABSTRACT – *The online community consists of several discussion groups, help communities and forums. The Question Answering system is a useful tool in such communities. The proposed system is a Question Answering system with an extended functionality of predicting the users' next question. By prediction, we eliminate the work required for posting each question and improve the efficiency and user friendliness of the system. The prediction is done using the data mining technique of Association rule mining. In particular, we make use of the Apriori algorithm to generate association rules among the questions posted by the user. We analyze the question log of the users to obtain their history with the system and derive their area of interest. The predictions are obtained using the question log as an input to the algorithm*

1. INTRODUCTION

The QA system has become an integral part of the internet which involves large communities discussing among themselves about topics of interest, problems and their solutions. You can find questions and their answers related to any domain in the QA systems. It is a very necessary tool for customer interaction or a general discussion group. The process involves a user posting a question about his/ her area of interest on the system which will be answered by the other users who may know an answer to it. The process becomes much easier if the next question posted by the user can be predicted. It takes out the extra work required by the user in posting the question and also provides a very interactive environment to the user. The prediction can be achieved by storing the question log of the user. Each question posted by the user are maintained as records in the log. Log analysis is done on the records upto a certain period. This reveals the patterns between the questions and helps in prediction. Analysis is done using data mining techniques like Association Mining. It associates one question to another by recognizing the pattern between them. The semantics behind this are as follows. The user posts a query to the system through a Question Processing Module. All the interactions between the user and the system are stored as a session. The module identifies the stock words which are unnecessary for prediction or type identification. The stock words are removed and the query is modified. This modified query is used as a reference to obtain the type of the question and its relevant documents in the Document processing module. The documents are searched to obtain the answer. Log analysis is done on the user's records to obtain association rules and the next question is predicted. These data are then presented to the user in the Answer Processing module. Fig.1 shows the architecture of a general Question answering system

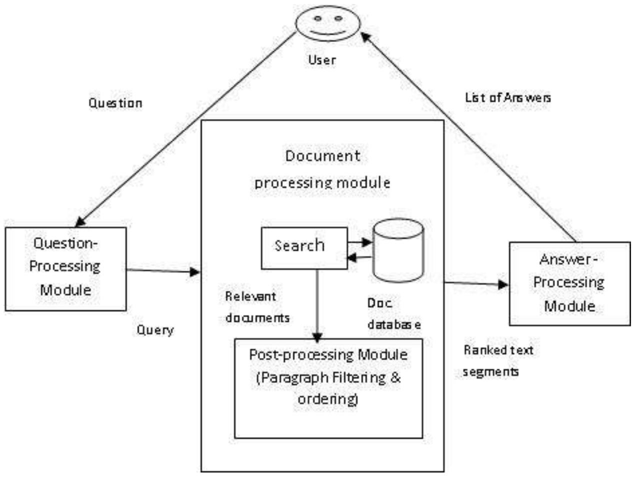

Fig.1 Architecture of Question answering system

This paper discusses about the modification of the current QA system by adding a prediction module to it to increase the interactivity and efficiency of the QA system. The prediction of next question by analyzing the question log through Association mining and Apriori Algorithm has been proposed. This paper discusses about the application of this QA system which predicts next question based on the current interactions of the user with the system. The organization of the rest of this paper is as follows. The literature review is presented along with the motivations in Section II. The modules and their interactions have been provided in Section III. The working of the system has been provided in Section IV. The experimental results have been provided in Section V. The Future enhancements have been provided in section VI and the Conclusion has been provided in Section VII

2. MOTIVATION AND RELATED WORK

Several research topics similar to ours have been used as references and a great help as knowledge sources. The primary goal is to obtain predictions of the users' next questions based on his interests. Limam, Coquil, Kosch and Brunie [1] worked on the enhancing semantic relations between the search query logs of the user using real life data from search engines. Fonseca et al. [2] research is very similar to ours. They worked on obtaining association rules between queries which are previously posted by the users. Zhang and Nasraoui [3] used the combined method of sequential search behaviour and content based similarity search to obtain query recommendations for a search engine. They use the method of graph analysis where the queries are represented as nodes in a graph and relation between the nodes is determined by the distance between each node. Cheng et al. [4] works on predicting the search pattern of the user based on the diverse browsing patterns of the user on the internet. They present three stages in their analogy – query extraction, building a model for prediction and an optimization algorithm for the model obtained in the previous stage. These diverse browsing patterns are similar to the diverse categories of questions posted by the users in our research. In Wang et al. [5] discusses the possibility of browsing the internet beyond hyperlinks. They provide a system of browsing the question logs of a user in a commercial search engine to obtain the browsing patterns and build a multi resolution topic map which can be zoomed in or zoomed out to vary the sensitivity of information. K.H Lin et al. [6] use a very similar approach to ours in predicting the users' next query. They use 3 different methods WTAL, SRPF and ACTF do encounter this issue. They determine that WTAL is the best among the methods. The paper also shows that historical data of a long time may not be suitable for prediction. Dupret and Mendoza [8] propose a system of recommending better queries for the user. They provide an improvised query as a recommendation for the user instead of providing related queries based on the user's question log. They also predict the date even when there are no matching keywords to be searched. They achieve this by ranking the data in the query log of the user. In [8], Madaan, Sharma and Dixit have used the Apriori algorithm to obtain association rules among the keywords present in the query log of the users for prediction of the next query to the user. This research is very much similar to ours in terms of working and algorithm usage. Much of the previous knowledge required has been obtained from their research. After analyzing all the mentioned resources, we have come up with our paper which aims on achieving a better result and overcoming all the disadvantages presented in the other researches. We have taken a sample question log of a user for 100 days and analyzed it offline to obtain our experimental results. Our aim is to enhance the QA system to have efficient prediction and a better interaction with the users to make it user friendly.

3. MODULES AND THEIR INTERACTIONS

The proposed system performs the QA module functionality with an addition of predicting the next question posted by the user.

The application mainly consists of 3 end modules

1. Visitor module: The visitor is a user who interacts with the system for the first time. He does not have any previous interactions with the system. He is presented with an option of registering himself to the system via a registration page to become a member. He can also view the trending questions of the system.

2. Member module: The member is a user who is registered to the system and has some or none previous interactions with the system. Each history of the user's interactions is stored as a log. The user can post his/ her query and can view the answers and questions related to the topic. The system also predicts the user's next question as soon as the answers are retrieved in the form of hyperlinks. He can also view the questions posted by other users on the system. The member can view his history with the system which can be the questions posted by him or the answers to the questions posted by others. He also has an option to change his profile details.

3. Administrator module: The administrator module consists of the administrators who handle the system. The administrators have permissions to manage users and the questions posted by them. The administrator enters keywords into the system upon which the question type is identified. These keywords also help in prediction. The administrator can view the questions posted by other users and can verify their content by activating them. The question can be viewed in the system only after the administrator activates the question. The administrator can also de-activate a question which was previously activated. The main functionalities of the administrator are managing the members and handling the QA module. The administrator also has an option the edit the details of his profile. Fig 2 shows CFD of the system

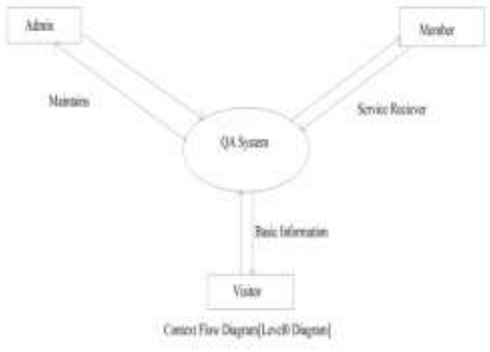

Fig. 2 CFD of the system

4. WORKING OF THE SYSTEM

A. User session extraction: Every time a member logs into the system, a session for that interaction is created. This session contains information about the questions posted in that particular interaction with the system. Sessions S = {S_1, S_2, S_3, ….., S_m} where m is the total number of sessions. In each session, there are questions Q = {Q_1, Q_2, Q_3, …., Q_n} wher n is the total number of questions in each session. The time constraint is applied on extracting sessions to obtain the latest information about the user's interactions. Each session contains the questions posted by the user in that interaction. These sessions in a particular time interval are extracted to obtain data about the area of interest of the user

B. Question extraction: Each session contains several questions posted by the user on the system. This module performs the extraction of questions from each session of the user. These questions contain both keywords and stock words necessary to obtain the answers to the question and the prediction of the next question. The questions are separated from their session IDs.

C. Question Filter module: The questions extracted from the sessions are filtered in this module. In this module, the category or domain of the question entered by the user is identified. Since the keywords are already fed by the administrator to the system for each category, the keywords present in the question are compared with the keywords entered by the administrator. A count is kept for each keyword matching to a particular category. The category containing the most keywords matched is identified as the question type. Identifying the question makes it easier to obtain the related questions and their answers. Once the question type is identified, the question is placed under that category in the database.

D. Question Pre-processor: Once the type of the question is identified, the question must be processed to obtain the required keywords to construct a query. The stock words from the sentence such as 'the' 'and' 'when' 'what' 'is' etc. are removed from the question to obtain the special words. The required keywords must be obtained from these special words. The special words are compared with the keywords stored in the database and each match from a special word to a keyword stored gives us the required keyword. Each such keyword is identified as a term in the query. If we take a query Q, it contains several terms T. Q = {T_1, T_2, T_3, ……., T_n}. The query contains the necessary data to build association rules and patterns.

E. Association rules generation: The association rule mining is a type of data mining technique which brings out the relation between data in transactions. A transaction can be anything from a purchase to a conversation. Here, we take posting a question as a transaction with the system. Association rule mining was first done among customers in a supermarket who bought daily necessities. The aim of finding relations between these transactions was to identify the customer purchase pattern and modify the sale process accordingly to increase the profits. For example, it was found that when several baskets of products purchased by the customers were analyzed in a supermarket, milk and eggs were usually bought together. In a similar way, if we apply the same logic to our system, we can consider the questions as baskets and the keywords as the items in the basket and we can generate associations among these items. There are two terminologies used in Association rule mining – Support and Confidence. Support is the measure of each item in every transaction. Confidence is the probability of the occurrence of an item with another item already being occurred. Support gives an idea about how frequently an item is used in a transaction. If the item occurs more frequently, it can be considered in the frequent itemset. A minimum support count is mentioned for an item to be considered in the frequent itemset. Any item exceeding the support count is considered as a frequent item. The support counts for combinations of these items are also calculated. Once the frequent itemset is obtained, the confidence values among the items are calculated. This is done by analyzing the occurrence of two items in combination. A minimum confidence count exists, for which any rule which exceeds this minimum count is considered as an association rule. Several such association rules are generated which reveals the pattern among the items in each transaction. Once the association rules are obtained, we can predict the next question easily by extracting all association rules associated with the keywords present in the question. To achieve this process, the Apriori Algorithm is used.

Apriori Algorithm

There are two major steps involved in the Apriori algorithm:

Step1: Finding the itemsets having minimum support (frequent itemsets, also called large itemsets). Let the transaction data is given as follows:

Query1: Student, Teach, School

Query 2: Student, School

Query 3: Teach, School, City, Game
Query 4: Baseball, Basketball, teach, school

Query 5: Basketball, Baseball, Player, Spectator

Assume the two thresholds be minsup=30% (0.3) and minconf= 80% (0.8). Some example frequent itemsets: Itemset1: {Teach, school} has sup= 0.6. Itemset 2:{Baseball, Basketball} has sup=0.4. Both of these itemsets satisfy the criterion sup>=minsup, so these are selected as frequent itemsets and are considered for the next step to generate association rules. Step 2: Generation of association rules using frequent itemsets. Using the frequent itemsets generated in the above step for generating association rules as follows: Rule 1:teach->school, sup=2/5 (0.6), conf=3/3 (1) Rule 2: school-> teach sup= 2/5 (0.6), conf=3/4 (0.7) Rule 3:baseball->basketball, sup= 2/5 (0.4), conf=2/2 (1) Rule 4:basketball->baseball, sup= 2/5 (0.4), conf=2/2 (1) So, associations rules that satisfy the criterion of conf>=minconf are Rule1, 3 and 4. The sup and conf values of the Rules above are shown in the Fig. 3

These associations generated are then given as input to the next module i.e. Next question predictor which generates the next question on the basis of the association rules that have been generated by the current module.

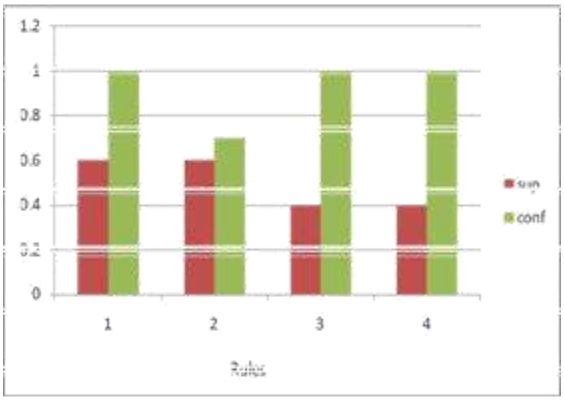

Fig. 3 Support and Confidence values

<u>*F. Next question predictor*</u>: This module considers the association rules generated in the previous module and retrieves questions according to those rules. The questions retrieved are of the same category as the question posted by the user. The keywords present in the questions posted by the user and the keywords present in the questions predicted by the system have strong associations between them. For example, if the user posts a question as “What are shares?” the predicted question may be “What are shares in the stock market?” or “What is cricket?” and the predicted question may be “Who is the best batsman?”. They keywords in the questions like shares and stock market or cricket and batsman are strongly related. Once the associations are identified, the category belonging to the keywords is identified and the questions related to that

category containing the keyword are retrieved. These questions are then presented in the form of a hyperlink to the user. The hyperlink redirects the user to the answers of the respective question.

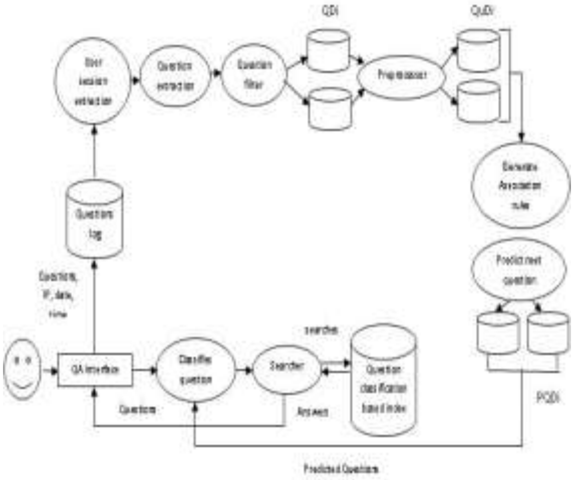

Fig. 4 Working of the system

5. EXPERIMENTAL RESULTS

For the purpose of analyzing the working of the system, we have taken some sample experimental measures. C# and .NET have been used on a 64 bit windows system to create the system. Two members are created who post the questions. The question log of the users is shown in fig 5.

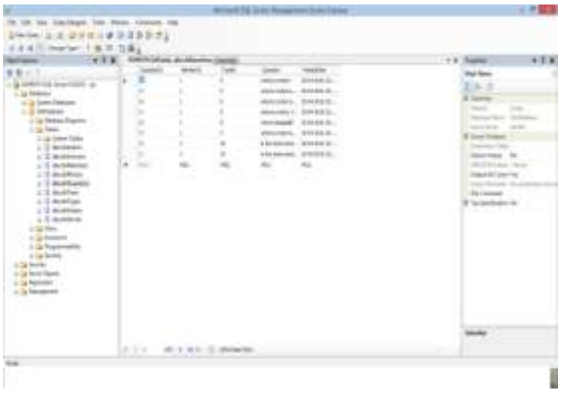

Fig. 5 Questions posted by the users

Each question belongs to either the sports or the finance category. This question answer module can be viewed by any of the members. There is also a filter to view the questions related only to a required category. Fig 6 shows the QA module of the system

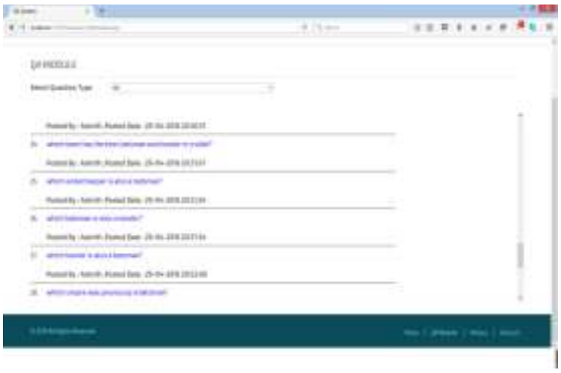

Fig. 6 QA module of the system

For the purpose of prediction, we have taken the minimum support as 50% and the minimum confidence as 80%. As soon as a query is submitted, the keywords are identified, frequent itemset is generated and association rules are obtained from this frequent itemset. In our case, we have entered the query as “Which stock purchase would incur loss”. The previous question logs contain other queries such as “Is the stock market in loss?” or “Is the stock market in profit?”. The system analyzes these logs and brings about the associations required. The association rules generated for the query input by the user using the previous question logs is shown in fig 7.

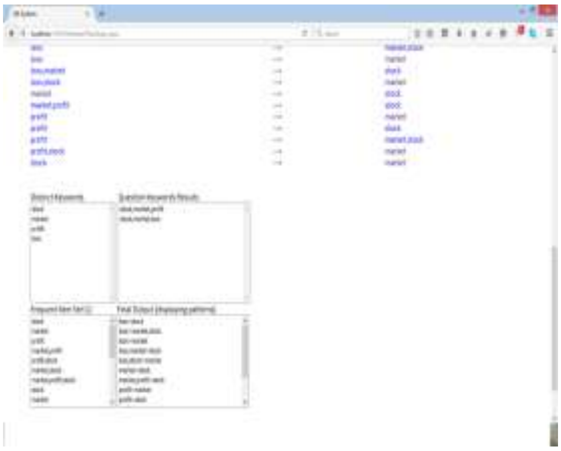

Fig. 7 The association rules generated after posting query

The experimental data show that the associations generated and the predicted questions are based on the user's previous interactions with the system. If the user has no previous interactions, the interactions of other members with the system are considered. The accuracy of the predictions increases with a larger training set for the system. It can be seen that the proposed system is very interactive, efficient and smart.

6. FUTURE ENHANCEMENTS

The proposed QA system improves the functionality, interaction and user friendliness. However, the system needs some enhancements to improve its quality. For the predictions to be accurate, a large number of data must be provided as a training set to the system. The information available also increases with the increased training set data. If more data is present, they system can obtain more association rules and many accurate predictions can be obtained. The system can also be enhanced to include more categories of questions and answers. A rating system can be added for the users to assess their quality of answers. A user with a high rating can be given the responsibility of a moderator. The process of manually entering keywords becomes very tedious as the size of the data increases. An automatic approach must be chosen to identify the question type. This reduces the workload on the administrator and also improves the efficiency of the system. As the questions and answers increase, the verification process takes a lot of time. To overcome this, the number of administrators and moderators must be chosen accordingly.

7. CONCLUSION

The proposed system is an improvement in the existing system by predicting the users' next question based on the previous interactions of the user with the system. This improves the effectiveness of the system with regard to the interaction and user friendliness of the system. The system uses the data mining technique, Association rule mining. This technique makes use of association rules generated by the Apriori algorithm to obtain a relation among transactions. The online community is always in search of a more efficient and user friendly approach to their problems. The proposed system overcomes most of the drawbacks of the existing system. The experimental results have proven to be a success showing a good performance of the proposed system.

REFERENCES

[1] L. Limam, D. Coquil, H. Kosch, and L. Brunie, “Extracting user interests from search query logs: A clustering approach,” DEXA '10 Proceedings of the 2010 Workshops on Database and Expert Systems Applications, 2010.

[2] B. M. Fonseca, P. B. Golgher, E. S. de Moura,, and N. Ziviani, “Using Association Rules to Discover Search Engines Related Queries,” in Proceedings of the First Conference on Latin American Web Congress, pp. 66–71, 2003.

[3] Z. Zhang, and O. Nasraoui, “Mining search engine query logs for query recommendation,” in Proceedings of the 15th international conference on World Wide Web, pp. 1039– 1040, 2006.

[4] Z. Cheng, B. Gao, and T. Liu, “Actively predicting diverse search intent from user browsing behaviors,” in Proceedings of the 19th international conference on World wide web, pp. 221–230, 2010.

[5] X. Wang, B. Tan, A. Shakery, and C. Zhai, “Beyond hyperlinks: organizing information footprints in search logs to support effective browsing,” in Proceeding of the 18th ACM conference on Information and knowledge management, pp. 1237–1246, 2009.

[6] K.H Lin, “Predicting Next Search Actions with Search Engine Query Logs,” Web Intelligence and Intelligent Agent Technology (WI-IAT), IEEE/WIC/ACM International Conference on (Volume: 1), 2011.

[7] G. Dupret, and M. Mendoza, “Recommending Better Queries Based on Click-Through Data,” LNCS, Springer, 2005.

[8] “A Data Mining approach to predict users’ Next Question in QA system” Rosy Madaan, A.K. Sharma, Ashutosh Dixit