ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

New approach for Effective Query Recommendation: Context Aware Fragment Based Method

Sumi Joy, Rini Simon

M.Tech Student, Dept. of CSE, VJCET, Vazhakulam, Kerala, India

Asst. Professor, Dept. of CSE, VJCET, Vazhakulam. Kerala, India

ABSTRACT: Interactive database exploration is a key task in information mining. However, users who lack SQL expertise or familiarity with the database schema face great difficulties in performing this task. To aid these users, we developed the recommender systems. Different kinds of recommender systems are there. For exploring the database different methods are available like collaborative exploration, cluster models, Search based query recommendation etc. In this project web based collaborative fragment based recommendation along with the context aware recommendation is implemented. Performance evaluation of the recommendation system is done by using the parameter recall. The recommendation is more effective in case of context aware fragment based approach.

KEYWORDS: Data mining, Database Exploration, Similarity calculation, Recall, interactive Data discovery, personalization.

I. INTRODUCTION

Data mining is the process of finding useful information from the data. Data may be quantitative or qualitative. In many applications (for example marketing or business related), the data to be handled may be very large (also known as big data). Discovering information from the huge amount of data may be difficult for the end users who lack SQL expertise. In such situations database exploration plays a major role. Database exploration tools help user to explore the database even though the underlying schema is unknown. Relevant data discovery is difficult for users and recommendation engine is a solution for such difficulty.

Different types of recommendation engine are available [2]. Collaborative filtering based recommendation engine are most popular. Recommendation engine track the users querying behaviour and use that behaviour to generate recommendations. User query is submitted in the user interface. The query execution engine will handle the query and it will provide the result. The role of recommendation engine is to store the user query and suggest similar queries to the users. Fig. 1 shows the architecture of a recommendation system [1] [3]. In Fig. 1 End users enter their query database query interface. Query interface pass this query to database engine and recommendation engine. Both database engine and recommendation engine process the user query. Database engine provide the query result to user. It also creates a query log in query log database. Recommendation engine uses the query log and user query and compute similar queries for the input user query. Similar queries are given to user as recommendation.

In context aware fragment based recommendation system recommendation engine uses the fragments [1] and context of the user query [4] to compute similar queries. The inspiration for this work is web recommendation strategy. In this work we analyses the web recommendation strategy for database recommendation. In web based recommendation system, users with similar search behavior will be identified. Subsequent queries of such a user are provided as recommendation for other user.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

The remaining of this paper is organized as follows: In Sections II and III we discuss related works and brief overview of the proposed work. We then present some implementation details in Section IV. The experimental evaluation of the work is done in section V and concludes the paper with our plans for future work in Section VI.

II. RELATED WORK

Different types of recommendation algorithms are used for the recommendation such as: Traditional Collaborative Filtering, Cluster Models and Search-Based Methods [2]. Collaborative filtering does little or no offline computation, and its online computation scales with the number of customers and catalog items [2] [9]. The algorithm is impractical on large data sets, unless it uses dimensionality reduction, sampling, or partitioning all of which reduce recommendation quality. Cluster models can perform much of the computation offline, but recommendation quality is relatively poor. To improve recommendation quality, it's possible to increase the number of segments, but this makes the online user segment classification expensive. Search-based models build keyword, category, and author indexes offline, but fail to provide recommendations with interesting, targeted titles. They also scale poorly for customers with numerous purchases and ratings. Exploring data from large database is a challenge. SnipSuggest [8] uses a query auto completion technique. When user writes a query in the query field, recommendations will be provides on the-go, context-aware assistance in the SQL composition process. Snipsuggest will suggest tables, views, predicates etc. Hive [7] is a database system, which store unstructured data. Hive database help users to store large amount of data. "you may also like" [6] describe different approaches for the recommendation like history based approach, current state approach, or external source approach. History based approach uses user search history. Current state approach use user's current behavior. External source approach use external sources to recommend recommendations. Personalized query recommendation [5] [11] generate recommendation based on the user preferences. User preference is modeled by user models and algorithms. Collaborative database exploration [9] includes two type of exploration: Tuple based exploration and fragment based exploration [1]. It Supports the efficient execution of complex queries and enables users to interactively explore the data [3]. Recommendation algorithm retrieves interesting information from the data. Tuple based exploration generate recommendations based on the witness value of a tuple in the query. Fragment based recommendation generate recommendation based on the fragment in the user submitted query. Tuple based recommendation provide efficient recommendations but fragment based recommendations increase the speed of recommendations.

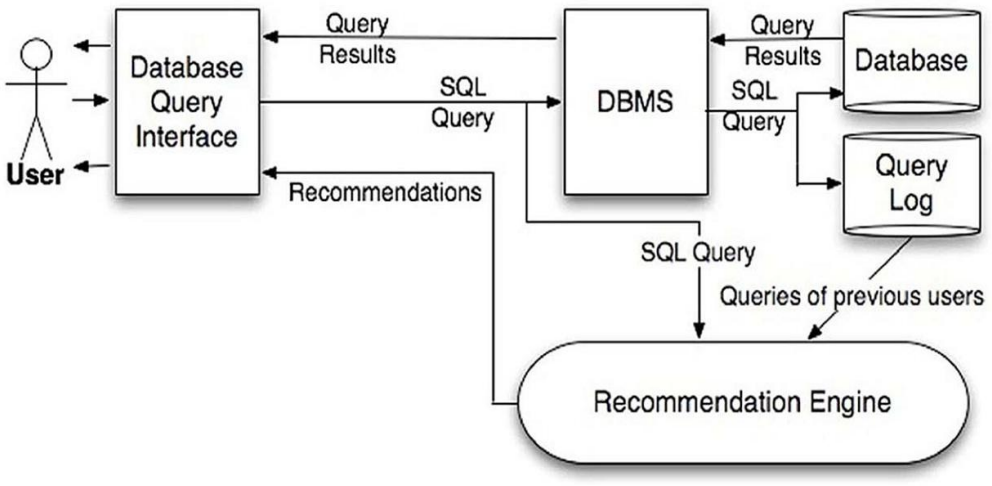

Fig. 1. Overall Architecture of Recommendation System

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

III. CONTEXT AWARE FRAGMENT BASED METHOD

Fragments are the predefined structures used in a query language [1]. SQL query language mainly contains 5 types of fragments as shown in Table I. Each fragment is identified using begin and end keywords. The context of a query is the entities and conditions used in the query. For example consider Query: Select * from employee fragments are select, from and context are *, employee. In context aware fragment based system both the context and fragments are considered. The active user's queries are forwarded to both Database and recommendation Engine. The Database processes each query and returns a set of results. The results include fragments of the user query. When each time user query is submitted query is logged in the Query Log. The Recommendation Engine combines the current user' input with history information of other users, as recorded in the Query Log, and generates a set of query recommendations that are returned to the user.

A. *Context Aware Approach:*

In traditional recommendation system user rating are used for the recommendation generation. In context aware system [4] context is also used with the user history and ratings. In context aware fragment based recommendation a web based recommendation strategy is used. In such recommendation context of the data item is also considered for the recommendation generation. Existing fragment based model searches the similarity of the user query with the stored fragments.

Table I. Fragments

Fragment Name	Begin Keyword	End Keyword
Attribute String	SELECT	FROM
Relation String	FROM	WHERE, GROUP BY, ORDER BY, end of query
Where String	WHERE	GROUP BY, ORDER BY, end of query
Group By String	GROUP BY	ORDER BY, HAVING, end of query
Having String	HAVING	ORDER BY, end of query

In Context Aware Fragment based QueRIE(CAFQueRIE) system User queries are computed by stored fragments. These computed queries are used for the fragment similarity calculation. Similar queries are provided as recommendation.

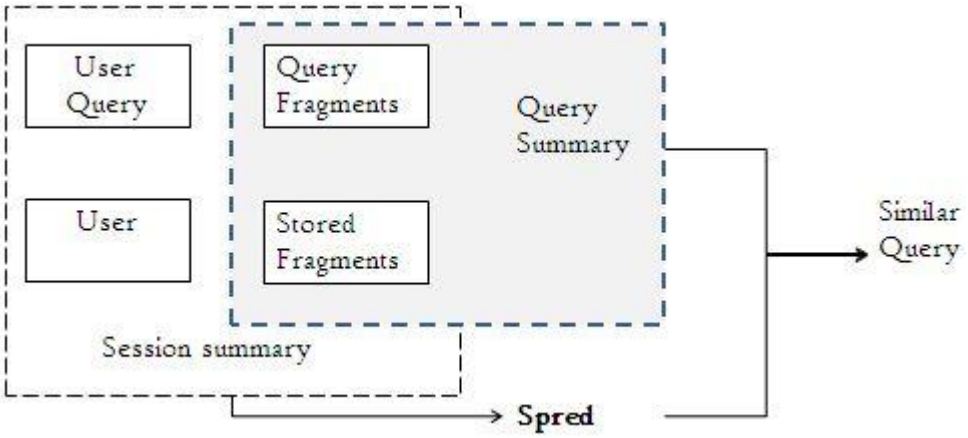

Fig. 2. Recommendation Procedure

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

Fig 2 shows the recommendation process of context aware fragment based method. User is the current user values such as user id and time stamp values. Use Query is the query inputted by the user. Current user and user query is used to compute a session summary of the user. Session summary include the fragments of current user query. Query summary of a user is computed using the query fragments of current user and previous users.

The CAFQueRIE system uses the query summary and Session summary S_{pred}[1] to compute the similar queries. Based on the S_{pred} recommendations are generated. This similar query is provided as suggestions. Algorithm for Recommendation Algorithm I is shown below:

Algorithm I: Algorithm for Recommendation

Input: User Current query Q_0, Users History Q_i
Output: Top Suggestions
1: Generate Fragments from user Query Q_0
2: Generate Fragments from users Query Q_i
3: Compute similarity of fragments generated using the equation eq.(1)
4: Construct query using the similar fragments
5: Check validity of the query
6: Add valid query to suggestion list
7: return suggestions

IV. EXPERIMENTAL RESULTS

The System is implemented in Java language. Database used is MySQL. Other than Java and MySQL Hibernate framework and JQuery is used for the implementation. The database used for the implementation and testing of CAF system is the classic model database. The system is not designed for a particular database. Any database can be used with the system. The work is mainly divided into two phases: Query Execution and Recommendation.

A. *Query Execution*

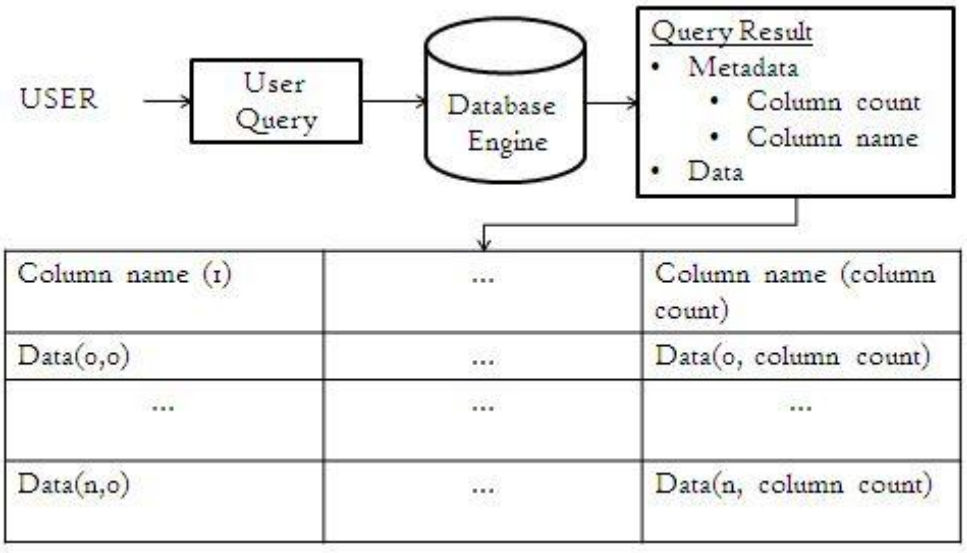

Fig. 3. Query Execution

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

In general, a query is a form of questioning, in a line of inquiry. In computing Query is a precise request for information retrieval with database and information systems. Query language, a computer language used to make queries into databases and information systems. Here we are using SQL as the query language.

User query is submitted in the query interface. Basic knowledge of the SQL query composing is an essential. Once the user submitted the query execution engine will execute the result and will return the query result. Result of query execution is displayed as a table model. Fig 3 shows the query execution. Meta data information and the result data is used to display the result in the table model.

B. *Recommendation*

1) Preprocessing: In CAFQueRIE system data preprocessing operation is performed using regular expression. Redundant and irrelevant data are preprocessed and converted in to an analyzable format. An example of preprocessing is shown below:

SELECT ID, COUNT (ASD) AS ASD FROM REGION WHERE TYPE LIKE 'TIPRIMARY' OR ID > 10.0 GROUP BY ID AND NAME HAVING COUNT (*) > 1

SELECT ID, COUNT (*) FROM REGION WHERE TYPE PATMATCH OR ID COMPARE NUM GROUP BY ID AND NAME HAVING COUNT (*) COMPARE NUM

In the above example > symbol is replaced with COMPARE LIKE 'TIPRIMARY' is replaced with PATMATCH and numeric values are replaced with NUM.

2) Fragment Generation: Table I shows the categories of fragments. From each query these fragments are separated and stored in the database. Pattern matching is done to identify the fragments.

3) Similarity Calculation: For the similarity calculation cosine similarity is used. Cosine similarity is a measure of similarity between two vectors of an inner product space that measures the cosine of the angle between them. The cosine of 0^0 is 1, and it is less than 1 for any other angle. It is thus a judgment of orientation and not magnitude: two vectors with the same orientation have a cosine similarity of 1, two vectors at 90^0 have a similarity of 0, and two vectors diametrically opposed have a similarity of -1, independent of their magnitude. Cosine similarity is particularly used in positive space, where the outcome is neatly bounded in [0, 1].The equation for cosine similarity is shown below:

$$Similarity = \cos(\theta) = \frac{A.B}{\|A\| \|B\|} \qquad \text{eq. (1)}$$

CAFQueRIE system provides Recommendations by combining the fragment based recommendation and context aware fragment recommendation, So that the relevance of the recommendation can be improved.

V. PERFORMANCE EVALUATION

QueRIE system and CAFQueRIE system models are tested with many users and feedback is collected from them. Each user gave a feedback on relevance of the recommendation. Based on the feedback both the systems are evaluated.

$$Recall = \frac{TP}{(TP + FN)} \qquad \text{eq. (2)}$$

In pattern information retrieval with binary classification, precision is the fraction of retrieved instances that are relevant, while recall is the fraction of relevant instances that are retrieved

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

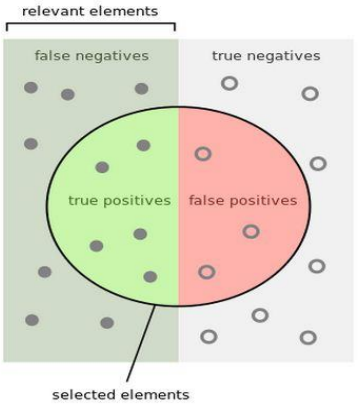

Fig. 4. Relevant elements

Fig. 5. Recall

Fig 4 shows the relevant and irrelevant items in a set. The recall is therefore based on an understanding and measure of relevance. Suppose a recommendation engine suggest Top 10 suggestions for a query. If 5 of the suggestion is relevant then its recall is 5/10 = 1/2. Recall is computed as the fraction of correct instances among all instances that actually belong to the relevant subset i.e. (Recall = Actual True Positive rate).

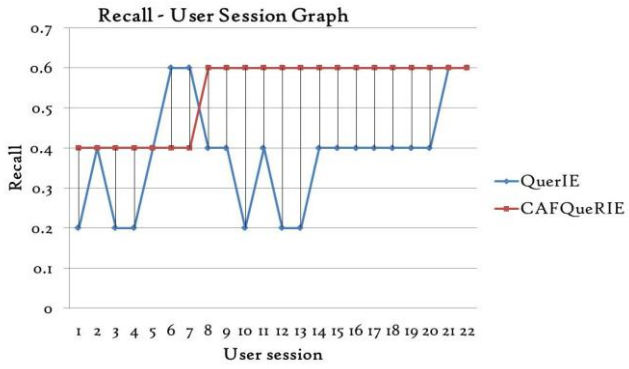

Fig. 6. Recall-User Session Graph

Fig 5 shows graphical representation of recall. Eq. (2) shows the equation for recall, where TP is true positive and FN is false negative. Fig 6 shows the performance evaluation of QueRIE and CAFQueRIE system. 20 user sessions are used to plot the Recall - User session graph. From the figure it's clear that in small number of user session CAFQueRIEsystem have better Recall compared to QueRIE system.

VI. CONCLUSION AND FUTURE WORK

Despite the availability of querying tools over large databases, users often have difficulties in understanding the underlying schema and formulating queries. Interactive Exploration of large database to aid users who lack SQL expertise or familiarity with the database schema is a challenge. For exploring the database different methods are available like collaborative exploration, cluster models, Search based query recommendation etc. Collaborative Item based recommendation provides efficient recommendation .Context aware fragment based method improves the effectiveness of the recommendation.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 11, November 2015

REFERENCES

1. Magdalini Eirinaki, Suju Abraham, Neoklis Polyzotis, and Naushin Shaikh, "QueRIE: Collaborative Database Exploration", IEEE Transactions on Knowledge and Data Engineering, Vol. 26, pp. 1778 – 1790, 2014
2. G. Linden, B. Smith and J. York "Amazon.com recommendations: Item-to-item collaborative filtering", IEEE Internet Comput., vol. 7, pp.76 -80, 2003
3. Gloria Chatzopoulou, Magdalini Eirinaki and Neoklis Polyzotis,"Query Recommendations for Interactive Database Exploration ", 21st International Conference, SSDBM, pp. 3-18 , 2009
4. Kostas Stefanidis,Evaggelia Pitoura,"A Context-Aware PreferenceDatabase System", J. PERVASIVE COMPUT. & COMM. 1 (1), 2005
5. G. Koutrika and Y. Ioannidis, "Personalized queries under a generalized preference model", Proc. 21st Conference on Data Engineering ICDE ,2005
6. K. Stefanidis, M. Drosou and E. Pitoura, ""You May Also Like" results in relational databases", Proc. 3rd Int. Workshop PersDB, 2009
7. A. Thusoo, "Hive - A petabyte scale data warehouse using hadoop", Proc. IEEE 26th ICDE, pp.996 -1005, 2010
8. N. Khoussainova, Y. Kwon, M. Balazinska and D. Suciu , "SnipSuggest: Context-aware autocompletion for SQL", PVLDB, vol. 4, pp.22 -33, 2011 .
9. Badrul Sarwar, George Karypis, Joseph Konstan, Joseph Konstan, "Item-Based Collaborative Filtering Recommendation Algorithms", WWW '01 Proceedings of the 10th international conference on World Wide Web. ACM, pp. 285-295 , 2001
10. N. Koudas, C. Li, A. K. H. Tung and R. Vernica "Relaxing join and selection queries", Proc. 33nd Int. Conf. VLDB, pp.199 -210, 2006
11. G. Koutrika , "Personalized DBMS: An elephant in disguise or a chameleon?" , IEEE Data Eng. Bull., vol. 34, pp.27 -34, 2011

BIOGRAPHY

Sumi Joy was born in Kerala, India on April 28, 1992. She did her Bachelor of Technology in Computer Science and Engineering from Mar Baseliose Institute of Technology and Science, Nellimattom, Kerala in the year 2013. Sumi is currently pursuing her Master of Technology in Computer Science and Engineering from Viswajyothi College of Engineering and Technology, Vazhakulam, Kerala. Her fields of interest lie in areas of Data mining and Big Data Analytics, etc.

Rini Simon was born in Kerala, India on July 28, 1986. She did her Master's in Computer Science and Engineering from Kumaraguru College Of Technology in the year 2010. She joined Viswajyothi College of Engineering and Technology and is working as Assistant Professor in Department of Computer Science since 2010. Her areas of interest are data mining and computer networks.