

# Efficient Query Analyzer with Personalized Recommendations

Sini R<sup>1</sup>, Lizmol Stephen<sup>2</sup>

<sup>1</sup>PG Student, Sarabhai Institute of Science and Technology, Thiruvananthapuram, India

<sup>2</sup>Assistant Professor, Sarabhai Institute of Science and Technology, Thiruvananthapuram, India

**Abstract:** *This paper deals with the development of flexible recommender systems, to make real time computations on vast amount of data. Even if most of the research has been dedicated to improve accuracy of recommendations, scalability is a serious issue. To address these issues, a QueRIE framework is developed to provide suggestions that are likely to interest users. QueRIE blends user characteristics into queries to generate personalized recommendations to user by prioritizing and filtering queries that are available. For this purpose this paper proposes a new approach called fragment based system in which tokenization of string is focused. In this paper, data management systems must provide powerful query management capabilities, from query browsing to automatic query recommendations. Here the session of active user is represented as strings. These recorded strings are used to identify similar query strings by means of string mapping in the previously recorded sessions, which are assembled together to form a complete query for the active user. So the system uses previous information of current user and previous users. Thus this system automates the process of suggesting recommendations. Finally a comparison of fragment based approach to the previously proposed method called tuple based approach is discussed.*

**Keywords:** collaborative filtering, database exploration, meta query, personalized and recommended systems

## 1. Introduction

Relational database systems have huge impact on scientific database users. Discovery of information what they need remains a big problem for these kinds of users. People who are not thorough with query making find difficulty to explore which part of database provide useful information. Database exploration is an important task in database mining [3]. To solve this problem of providing guidance to these users, a QueRIE suggestion system is developed. Here first user queries are fragmented into components for the easiness of calculation of similarities and then reassembles them into simple, understandable meaningful queries. Thus a closed-loop approach is followed [1, 3]. Here also explains personalized query answering in the context of databases.

In the previous work to represent user queries user-based collaborative filtering is used [2]. This paper uses query fragments to represent queries including an overview of tuple based approach. To identify similar components in the previous session similar query strings are used which are reconstructed in interesting queries for active users [1, 2]. A meta query is a query that enables users to locate past queries that match with the previous queries [2]. The users can mine system log to generate recommendations to present queries that match their querying characteristics than to provide with synthetic ones.

The remaining of this paper is organized as follows: Section 2 provides a brief overview of related works of QueRIE system. In section III previous work of query recommendation system is presented, in section IV several major challenges of query recommendation system is presented. Section V discusses an overview of activity diagram of the proposed system in step by step procedure and describes an overview of the abstract framework of the

QueRIE system in tuple\_based approach and then present the fragment based approach. The experimental evaluation is presented in Section VI and followed by conclusion in section VII.

## 2. Literature Review

Even though there are recommendations addressed in the Web context, only a few related works exist in the database context. The necessity of a query recommendation framework is emphasized in “A case for a collaborative query management system” [2]. In Context-aware auto completion for SQL, SnipSuggest recommends possible additions to various clauses in the current user’s query, and not complete queries [4]. Moreover each query is treated independently of any previous one, even if they belong to the same user session. In “ReDRIVE: Result-driven database exploration through recommendations” a content-based approach is followed [5]. Web databases provide a keyword-based query interface, and suffer from the “empty-answer” and “too-many-answers” problems. In logical foundations of preference queries they show how preferences can be embedded into relational query languages [6] [7].

## 3. Previous Work

In the previous work they developed a tuple\_based approach of the framework using user-based similarities to generate recommendations. When the tuple\_based instantiation employs approximation techniques (Minhash synopses) to enable real-time calculations, the loss in precision is much greater than that of the fragment-based one [1].

Many relational mining techniques are directly applicable for query mining, especially multi relational mining since queries are likely to be stored in multiple database relations.

Related work also is the work on association rule mining. Work that may be adapted for query similarity includes the Context Distance Measure framework designed for computing the distance between two objects defined across multiple relations. In previous work they discussed the requirements for a collaborative query management system. They outline early system architecture and discuss the many research challenges associated with building such an engine [2].

#### 4. Analysis of Problem

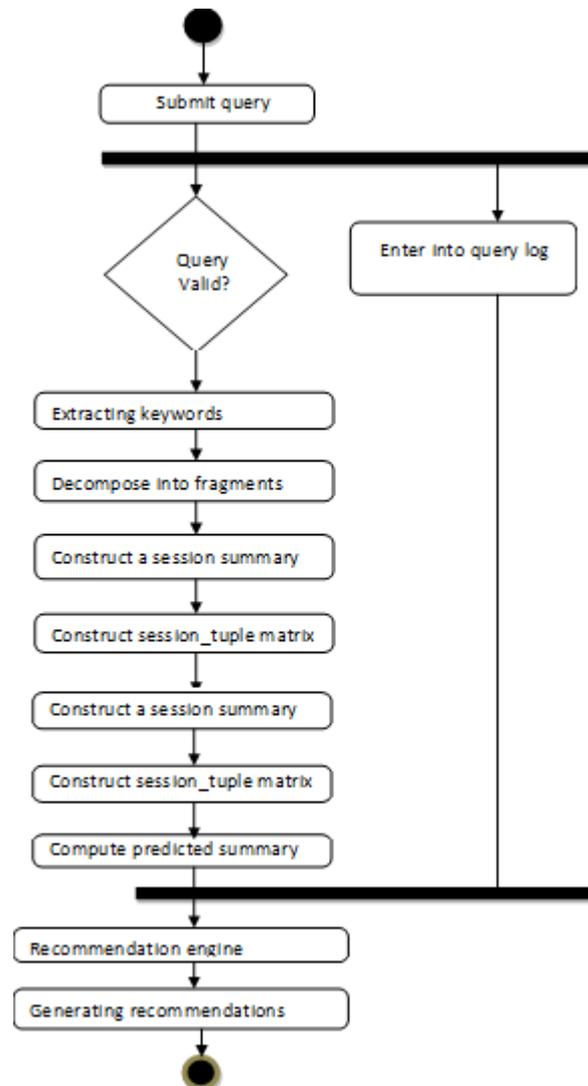
The traditional DBMS usage patterns are challenging in the area of emerging applications of large-scale scientific data management and industrial data analysis. There is a need for formulating new queries. The traditional trial- and error method is too expensive due to the increasing size of data. A human guidance is costly and unrealistic. Moreover scalability is one of the serious issues for future generation recommendation systems. For the queries posed by the user there is an absence of explicit rating mechanism. There is a problem of how to solve query-equivalence problem.

Challenges:

- Managing a collection of queries is more difficult rather than managing ordinary data. To generate personalized efficient query recommendations.
- The system must be efficient because it must provide hints and recommendations interactively, as a user types a new query.
- Should include an easy-to-use interface and effective visualization methods.

#### 5. Proposed Work

Taking into consideration the findings of previous work, where developed a tuple\_based approach of the framework using user-based similarities to generate recommendations, here decided to follow an item-based approach using query fragments to represent user sessions. The fragment-based approach can be implemented very efficiently, the space of fragments grows slowly, the summaries are very sparse and, most importantly, the fragment-to-fragment similarities can be computed offline and stored for very fast retrieval when recommendations need to be generated. The method propose a scalable design and generates meaningful recommendations on real-life traces that accounts the incremental update of similarities, thus computations on large amounts of data is feasible in real time. The dynamic nature of this work is shown below using activity diagram.



**Figure:** Activity diagram of query recommendation

#### Methodology

Objective of this paper work is an instantiation of the QueRIE framework. The system is going to describe two instantiations of the abstract recommendation framework, namely a *tuple-based* recommendation engine and a *fragment-based* recommendation engine.

##### A) Tuple\_based recommendation

Here the session summary is represented as a weighted vector, where every coordinate corresponds to a distinct database tuple. The weight of each tuple represents the importance of a given tuple *in* the session of user, and is non-zero only if tuple is a witness for at least one query in the session. The intuition is that session captures the tuples in the base tables that are touched by the queries in the session. Hence, sessions that contain equivalent queries will map to the same summary.

##### B) Fragment\_based recommendation

The fragment-based approach works similarly as that of tuple-based recommendation, except that the coordinates of the session summaries correspond to fragments of queries instead of witnesses. Here identify as fragments the

following syntactical features of the queries in the session: attribute references, table references selection predicates and join. The idea behind this approach is to recommend queries whose syntactical features match the queries of the current user.

## 6. Experimental Result Analysis

In what follows this paper first present an evaluation of the various parameters of the fragment-based approach. This small-scale experimental evaluation, using a subset of the dataset helped me to choose the default values of the system's parameters. Using these, more extensive experiments can be preceded on the entire data set. Then present these results, along with an overview of the results of the tuple based approach [1], in order to compare and discuss the trade-offs of the two instantiations.

## 7. Conclusion and Future Scope

This paper presents the QueRIE framework that aims to generate useful SQL query recommendations to users of relational databases. Taking into consideration the findings of previous work, a more generic and scalable system is developed. This is a first-cut solution to the very interesting problem of interactive database exploration. The fragment-based approach can be implemented very efficiently; the space of fragments grows slowly, the summaries are very sparse and, most importantly, the fragment-to-fragment similarities can be computed offline and stored for very fast retrieval when recommendations need to be generated. While the fragment-based approach seems as a straightforward selection, new challenges related to the formulation of session similarity, the synthesis of recommendations and their presentation arise. I would like to measure the impact the query relaxation process has in the quality of recommendations in future.

## References

- [1] Magdalini Eirinaki, Suju Abraham, Neoklis Polyzotis, and Naushin Shaikh, "QueRIE: Collaborative Database Exploration", *IEEE Transactions on knowledge and data engineering*, vol. 26, No. 7, July 2014
- [2] G. Chatzopoulou, M. Eirinaki, and N. Polyzotis, "Collaborative filtering for interactive database exploration," in *Proc. 21st Int. Conf. SSDBM*, New Orleans, LA, USA, 2009, pp. 3–18.
- [3] S. Mittal, J. S. V. Varman, G. Chatzopoulou, M. Eirinaki, and N. Polyzotis, "QueRIE: A recommender system supporting interactive database exploration," in *Proc. IEEE ICDM*, Sydney, NSW, Australia, 2010.
- [4] N. Khoussainova, Y. Kwon, M. Balazinska, and D. Suciu, "SnipSuggest: Context-aware autocompletion for SQL," *PVLDB*, vol. 4, no. 1, pp. 22–33, 2011.
- [5] M. Drosou and E. Pitoura, "ReDRIVE: Result-driven database exploration through recommendations," in *Proc. CIKM*, Glasgow, U.K., 2011, pp. 1547–1552.
- [6] G. Koutrika, "Personalized DBMS: An elephant in disguise or a chameleon?" *IEEE Data Eng. Bull.*, vol. 34, no. 2, pp. 27–34, Jun. 2011.

- [7] K. Stefanidis, G. Koutrika, and E. Pitoura, "A survey on representation, composition and application of preferences in database systems," *ACM Trans. Database Syst.*, vol. 36, no. 4, Article 19, 2011.
- [8] J. Levandoski, M. Mokbel, and M. E. Khalefa, "FlexPref: A framework for extensible preference evaluation in database systems," in *Proc. IEEE ICDE*, Long Beach, CA, USA, 2010.
- [9] J. Akbarnejad *et al.*, "SQL QueRIE recommendations," *PVLDB*, vol. 3, no. 2, pp. 1597–1600, 2010.
- [10] A. Giacometti, P. Marcel, E. Negre, and A. Soulet, "Query recommendations for OLAP discovery driven analysis," *IJDWM*, vol. 7, no. 2, pp. 1–25, 2011