Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 3, March 2014, pg.45 – 48

REVIEW ARTICLE

Data Services For E-Tailers Leveraging Web Search Engine Assets- A Review

Shaleena K.P

Thomas George

Jyothi Engineering College Thrissur

naah.lee625@gmail.com

Abstract— ***This review paper presents a study about how the data services can be provided for an e-tailer by mining the web search engine assets. It describes in detail some of the existing approaches to mine data from large databases to the methods that can be used to mine data from the query log of a search engine. It gives a brief description about some string matching techniques. It also discusses about the techniques used behind each of the data service which is been presented.***

I. INTRODUCTION

The main idea presented in the paper 'Data services for e-tailers leveraging web search engine data assets'[1]is to make a connection between search engine data assets and mining information for e-tailer's products. These data services for e-tailers can make them able to provide some superior experiences to their customers. The superior experiences include query auto completions, identifying synonym queries etc. These superior experiences in turn help the e-tailers to attract more customers towards them. The method proposed is very helpful for small scale e-tailers as they don't have any rich data assets of their own. So it is a better idea to mine necessary information from the query log of a search engine and then providing them as data services for e-tailers.

The data services that is been developed are entity synonym data service, entity tagging data service and query to entity data service.All these data services are developed by mining data from the query log of a web search engine. So it is better to understand how these can be done by reviewing some of those previously done works. The previously done works describes the different ways in which large databases has been mined and can be utilized to mine the query logs to provide the data services that is been discussed in [1].

II. QUERY LOG MINING TECHNIQUES

DBXplorer is a system for keyword based search over relational database [2].The idea used is to retrieve relations that contains all query keywords. When a user enters a query, this returns all rows either from a single table or by joining tables. This can be done by using two steps:

1. Publish
2. Search

In Publish, the database is enabled for keyword search by first identifying a database along with set of tables by creating symbol tables which are used at search time to efficiently determine the locations of query keywords in the database. In the search step, the symbol table is looked up to identify the location of the database. Then for each enumerated join tree an SQL statement is generated so that it returns rows that contains all keywords.

In Agglomerative clustering of a search engine query log [3] a method for clustering related queries and related URLs from a collection of user transactions with a search engine is been done. The method used doesn't depend on the content of the pages, but instead uses co-occurrence information across multiple transactions. This brings together query log analysis, web page clustering and query clustering. The clustering is done through click through data. The first step is to construct a bipartite graph where the vertices on the other side corresponds to URLS and an edge between them indicates they co occurred in a click through record. To discover groups of similar queries and groups of related URLs an iterating agglomerative algorithm can be used. The algorithm merges two vertices with high similarity there by revealing similarities between vertices which aren't apparent in the original graph by executing. Iteratively it iterates until a termination condition is met.

III. INTEGRATED WEB SEARCH ENGINE ARCHITECTURE

Exploiting web search engines to search structured data bases [4] is such a work in which the existing search engine components are leveraged to enable entity search over structured databases. The main aim is to identify mentions that occur in close proximity to query keywords; then by aggregating the occurrences of entities in the top N web search results, the most relevant entities can be retrieved. For that we are integrating search engine with entity search. Along with the four main components of the web search engine- crawlers, indexers, searchers and the front end there are three modifications which are entity extraction, entity retrieval and entity ranking units.

1. Entity Extraction: This component takes tokenized text documents as input analyses it and outputs all mentions of entities that are present in the database.
2. .Entity retrieval: This retrieves the entities extracted from the document.
3. Entity ranker: This ranks the set of all entities retrieved by the entity retrieval component.

IV. STRING MATCHING TECHNIQUES

All the techniques which has been discussed above involves string matching, like it always compares the query keywords with entity names present in the databases. In many situations it is necessary to find similarities between the string entered by the user and the ones present in the database, query log etc. So, it is necessary to have some idea about string matching techniques. There are exact string matching techniques and approximate matching techniques.

Efficient String Matching [5] discusses the basic idea of string matching which is always needed to have a better understanding of[1] .It describes a simple, efficient algorithm to locate all occurrences of any of a finite number of keywords and phrases in an arbitrary exit string. The algorithm consists of two parts: first to construct from the set of keyword finite automata and second, apply the text string as an input to the pattern matching machine. The behavior of such a machine is dictated by three functions: a goto function,an output function and a failure function.

1. The goto function maps a pair consisting of a state and an input symbol into a state or the message fail.
2. The failure function maps a state to another.
3. The output function associates a set of keywords as output with every state.

An Efficient Filter for Membership Checking [6] formulates an efficient technique for finding approximate matches of an input query with some members of a probably large dictionary. This concept is really applicable mainly in shopping sites as most of them are having backend databases with product dictionaries. An approximate membership checking problem identifies substrings which approximately match with a dictionary string.

Approximate membership checking is based on the principle that if 'm' is a substring of an input string 'S' which is a candidate member, then it is said to be a true member if there exists a dictionary string 'r' such that their similarity exceeds a certain threshold. There are three main such similarity measures

1. edit similarity
2. jaccard similarity
3. weighted jaccard similarity

.For computational efficiency a filter doesn't always compare a query substring with every dictionary string it uses a pruning condition to do this. The ISH filter is able to efficiently identify the queries which cannot match with any dictionary string. Candidates that pass the filter will be verified . ISH requires multiple signatures to be matched simultaneously by using weighted signature scheme. In the online querying phase the query is applied and candidates are generated similarly. Afterwards identifying the candidates the false candidates are filtered out using a one-at-a time verification.

V. DATA SERVICES

The data services which has been discussed in [1] can be described more clearly with the help of some previous works.

Context sensitive query auto completion [7] proposes an algorithm for query auto completion which can find hits even when the users input is still very short. The main observation is that the user typically has some context, which can reveal more information about her intent. The focus in this study is on the user's recent queries ie; within the same session. It is been assumed that when the context is relevant to the intended user query, the intended query is likely to be similar to the context queries. Here the query is expanded by iteratively applying a black box query recommendation algorithm on the query, on its recommendations, on their recommendations, and so on. The nearest completion algorithm which is proposed here keeps track of a user' login session by keeping the user's recent queries in a cookie, It finds out the conditional probability that that the next query is 'q' given the context is 'C'. Nearest Completion is designed to work well when the user input has a non-empty context and this context is relevant to the query that the user is typing.

A framework for robust discovery of entity synonyms [8] discusses the idea of finding entity synonyms. Several methods which have been in use before are based on click similarity, document similarity and distributional similarity. A synonym discovery framework is proposed which can make use of the existing similarity functions .The system carry out a two way checking to ensure symmetric property of synonyms. Query context similarity to make sure that candidate synonym string is of the same class of the entity. Main insight here is that the words that appear in the context of entity names in web search queries can help us distinguish between entities of different classes.

VI. CONCLUSION

As we all know, e-tailing is becoming a huge business these days. So the idea presented to mine data services for e-tailers from web search engine[1] is really great. But the techniques used to mine these signals must be made more efficient so that the service becomes more reliable for all type of e-tailers. So in future more works can be done in order to improve the mining techniques of the query logs there by making this idea a great success.

REFERENCES

[1] Data Services for E-tailers Leveraging Web Search Engine Assets by Tao Cheng, Kaushik Chakrabarti, Surajit Chaudhuri, Vivek Narasayya, Manoj Syamala ,In ICDE Conference 2013

[2] S. Agrawal, S. Chaudhuri, and G. Das. *DBXplorer: A System for Keyword-Based Search over Relational Databases*. In IEEE ICDE Conference, 2002

[3] Beeferman.D and Berger.A, *Agglomerative clustering of a search engine query log*,KDD 2000

[4] Sanjay Agrawal et al .*Exploiting web search engines to search structured databases*. In WWW, 2009.

[5] A.V. Aho and M. J. Corasick. "Efficient string matching: An aid to bibliographic search. Commun". *ACM*,(1975),18(6):333-340

[6] Kaushik Chakrabarti, Surajit Chaudhuri, Venkatesh Ganti, and Dong Xin. *An efficient filter for approximate membership checking*. In SIGMOD, 2008

[7] Ziv Bar-Yossef and Naama Kraus. *Context-sensitive query autocompletion*. In Proceedings of WWW, 2011.

[8] Kaushik Chakrabarti, Surajit Chaudhuri, Tao Cheng, and Dong Xin. *A framework for robust discovery of entity synonyms.* In SIGKDD, 2012.

