Using association rules to discover related queries on search engines

Bruno M. Fonseca[1], Paulo B. Golgher[2], Edleno S. de Moura[3], Nivio Ziviani[4]
[1]Fed. University of Minas Gerais, Belo Horizonte, Brazil
[2]Akwan Information Technologies, Belo Horizonte, Brazil
[3]Fed. University of Amazonas, Manaus, Brazil
[4]Fed. University of Minas Gerais, Belo Horizonte, Brazil
[1]maciel@dcc.ufmg.br, [2]golgher@akwan.com.br, [3]edleno@dcc.fua.br, [4]nivio@dcc.ufmg.br

ABSTRACT

This work presents a method for automatic generate suggestions of related queries submitted to Web search engines. The method extracts information from the log of past submitted queries to search engines using algorithms for mining association rules. Experimental results were performed on a log containing more than 2.3 million queries submitted to a commercial searching engine giving correct suggestions in 90.5% of the top 5 suggestions presented for common queries extracted from a real log.

Keywords

related queries, data mining, query expansion

1. INTRODUCTION

The Web has become an essential source of up-to-date information that covers almost all the topics a human could be interested in. However, the task of finding relevant information related to a given topic on the Web is difficult. The information available on the Web is not structured, therefore the useful material related to any searched topic is always mixed with billions of Web pages with little or no interest. Under this scenario, Web search engines became one of the most popular service available on the Web.

Despite the recent advances on the technology of the search engines there are still many situations where the user is contemplated with non-relevant answers. One of the great challenges faced by search engines is the difficulty in uncovering an exact description of the user need, since users usually submit very short and imprecise query[9].

A popular solution to help the users in the task of specifying their information needs is to use relevance feedback techniques [3]. These techniques improve the interactivity of the system by allowing users to inform about the relevance of answers given to their initial query. The feedback information is used to refine the initial query and get a better specification of the user needs.

A form of relevance feedback that has recently become popular in many search engines is to show a list of *related queries* to the user initial query. For instance, if the user searches for "*Madonna*" in All the Web search engine[1] the following related queries are presented : "*madonna lyrics*", "*madonna music*", "*madonna mp3*" and "*madonna wedding*". The presentation of a list of related queries is an interesting feedback alternative because user can explicitly reformulates the query, removing possible ambiguities, or turning its query more specific, or just redirecting its query to another topic which is related to the initial query and that. Despite of its raising popularity among search engines, there is a lack of related work in the literature on how to get query suggestions automatically. i

The objective of this paper is to present a method for automatically generate lists of related queries. The method uses an algorithm for mining association rules from the log of past submitted queries to a search engine. In our experiments we used a log containing more than 2.3 million queries submitted to *Farejador*[2], a popular brazilian search engine. Experimental results show that our method is both precise and useful. It usually usually generates correct suggestions that retrieve relevant documents to the user. Also, we show that the same approach can be used to provide terms to the classic problem of query expansion.

This paper is organized as follows. Section 2 presents related work. Section 3 discuss the proposed method . Experimental results are presented in Section 4. Finally, Section 5 presents the conclusions.

2. RELATED WORK

Association rules are widely used to develop high quality recommendation systems in e-commerce applications available in the Web [6, 10]. In these applications the systems use user sessions stored at system logs to obtain information about the user behavior and recommend services and products.

Our work here makes the assumption that the same idea can be applied to help search engine users in the task of finding relevant information available in the Web. The idea is to find previous search pattern that matches the current user query and use this information to suggest related queries that may be useful to users.

The idea of using information available from query logs was already exploited in previous work. In [7] it is proposed a method to study the relationship between queries analyzing the relationship between terms. The method takes the top 10 answers given by a search engine for each query and use this information to study the relationship among query terms and document terms. This relation is mapped in a graph which may be navigated by users to refine their queries.

Another technique which uses search engine logs is proposed

[1]http://www.alltheweb.com

[2]http://farejador.ig.com.br

WWW2003, May 20–24, 2003, Budapest, Hungary.
ACM 1-58113-680-3/03/0005.xxx.

in [5]. In this technique, the authors suggest a method for finding relations between queries and phrases of documents. This work uses the hypothesis that the click through information available on the search engine logs represents an evidence of relation between queries and documents chosen to be visited by users. Based on this evidence, the authors establish relationships between queries and phrases that occurs in the documents chosen. These relationships are then used to expand the initial user query or to give query suggestions.

The two works presented above are based on the idea that there is relation between queries and the textual content of documents selected for these queries. This assumption is not always true when we are dealing with the Web, where many documents may contain noisily or non textual information and also new search engine are using alternative sources of information to rank documents [4].

Furthermore, in both cases the information extracted from logs about queries relationship depends also on the search engine results, which is costly and makes the methods highly dependent on the quality of the search engine used in the experiments. An example of this problem is presented in [7], where experiments show that their method give different performances when applied with two different search engines.

Our method avoids this dependence because information about queries relationship is extracted exclusively from query logs. We don't ask any information from the search engine and don't use the document contents. This new approach is possible due to the use of association rules as the base technique to extract information from logs.

Another way to guide users in the task of finding relevant information is to develop query expansion techniques [3, 8]. Some query expansion methods can be adapted to give suggestion of new queries. This strategy is different from finding related queries because the expansion methods construct artificial queries, while in our case we give actual related queries formulated by other users which had the same information need in the past. On the other hand, the related queries can also be used like a query expansion method. In [5] is suggested a method that uses the relationships extracted from search engine log files for query expansion. We will present some experiments showing how to use our method to derive an useful expansion method.

3. IDENTIFYING RELATED QUERIES

Our method for identifying related queries is divided in two phases, as shown in Figure 1. In phase 1, search engine logs are analyzed and user sessions are extracted. On the second phase, association rules are mined from the set of user sessions and related queries are identified. We now describe each phase in details.

3.1 Extracting user sessions

We call a user session s all the queries made by a user in a predefined time interval t[3]. The set of user sessions can be extracted directly from search engine web server logs. Figure 2 presents a typical log format as generated by the *Squid Proxy*, the server used in our experiments, and Figure 3 presents a real example. Although this format is specific to *Squid*, all web servers generate similar log files.

Based on the log information, the user session is defined as follows. Each user is identified by the remote_host field (its IP address) and the session is defined by the set of queries (extracted from the URL field) each user submitted, divided in t minutes intervals according to the time field.

[3]In our experiments, we used $t = 10$ minutes

Once the set of user sessions S is characterized, we can now start the phase 2 of our method, as described in the immediately bellow.

3.2 Mining Association Rules and Identifying Related Queries

Before introducing our method to discover related queries, we now briefly describe the problem of mining association rules. We use an example formalized in [1] to revise the needed concepts. This example is based on the problem of mining sales data, called basket data. Let $I = \{I_1, I_2, ..., I_m\}$ be a set o literals called items. Let T be a database of transactions. Each transaction t can be represented by a binary vector, with $t[k] = 1$ if t bought the item I_k, and $t[k] = 0$ otherwise. Let X be a subset of I. A transaction t satisfies X if for all items I_k in X, $t[k] = 1$.

This statement can be redefined for the problem of finding related queries. Here, the set $I = \{I_1, I_2, ..., I_m\}$ is a set of queries from log files and T is the the set of user sessions t. For each t there is a binary vector $t[k]$ such that $t[k] = 1$ if session t searched for query I_k, and $t[k] = 0$ otherwise. A transaction t satisfies X exactly as described in last paragraph.

By an association rule we mean the implication $X \Rightarrow Y$, where $X \subset I$, $Y \subset I$, and $X \cap Y = \emptyset$. The rule $X \Rightarrow Y$ has a confidence factor of c if $c\%$ of the transactions in T that contains X also contains Y. We will use the classical notation $X \Rightarrow Y|c$ to specify that the rule $X \Rightarrow Y$ has a confidence factor of c. The rule $X \Rightarrow Y$ has a support factor of s if $s\%$ of the transactions in T contain $X \bigcup Y$. The problem of mining association rules is to generate all association rules that have a support greater than an specified minimum support (also called *minsup*) [2].

In the problem of finding related queries, we are only interested in association like $X \Rightarrow Y$, where $X \subset I, Y \subset I, X \cap Y = \emptyset$ and X and Y have only one element. We say that a query I_x is related to I_y if the transaction $X \Rightarrow Y$ has at least a *minsup*, where $X = I_x$ and $Y = I_y$. The intuition behind our method is as follows. During a session, the user defines (roughly) his information need submitting a set of queries. If distinct queries occur simultaneously in many user sessions, these queries may be related. This is a very simplified form of using association rules, however, it provides us a formal framework for analyzing the query logs, its implementation is efficient enough to process huge log databases and, as shown at the experiments presented here, it produces good results

Now for each query I_k we have a set of transactions ($\{I_k\} \Rightarrow \{I_1\}, \{I_k\} \Rightarrow \{I_2\}, \{I_k\} \Rightarrow \{I_3\}, ..., \{I_k\} \Rightarrow \{I_m\}$) sorted by it confidence. Then, I_k related queries is the set $\{I_1, I_2, I_3, ..., I_k\}$, where $\{I_k\} \Rightarrow \{I_1\}|c > \{I_k\} \Rightarrow \{I_2\}|c > \{I_k\} \Rightarrow \{I_3\}|c > ... > \{I_k\} \Rightarrow \{I_m\}|c$. It can be easily sorted by the frequency that each query appears in the same session of I_k. This procedure is justified by the definition of confidence and because our set of association rules has only one element for each transaction.

This simple definition allow our method to compute the relation between queries in an extremely fast way, which means new association rules can be updated periodically to identify new groups of related queries. This feature is important since the topics searched on the Web are dynamic and new relations may arise every day. Notice that techniques based on more complex log analyzes or based on associations between query logs and document contents do not allow this updating.

4. EXPERIMENTAL RESULTS

We present in this section experiments to evaluate the quality of our method. All experiments were performed using a log with 2312586 queries from a popular search engine in Brazil (Farejador IG). For these experiments we have implemented our method using

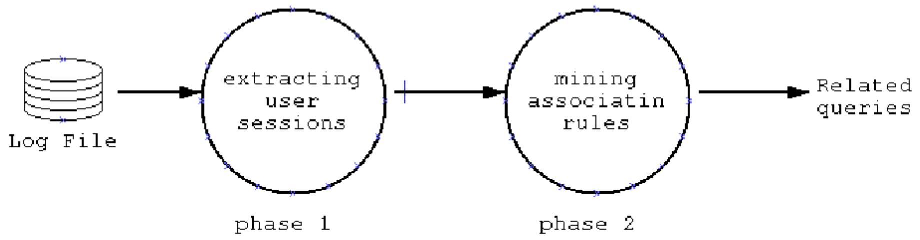

Figure 1: Identifying related queries

time elapsed remote-host code/status bytes method URL rfc931 peerstatus/peerhost type

Figure 2: Squid log format.

$minsup = 3$.

Table 1 shows related queries found by our method for the top 5 most popular queries in this period. The translation of non English words are presented in parenthesis to turn the results clearer. The query "games", has the queries "jogos", which is a translation of "games" to Portuguese, "jogos gratis", which means "free games" and the singular form of the word, which is "game". The results for the top 5 queries where quite good, despite of the wrong suggestions of "sexo" related to the query "games". This occured probably because "sexo" is a very common query submitted by users that also searched for "games".

Table 2 shows the results for an experiment using the top 95 most popular queries. For each query we evaluated the first 5, 10, 15 and 20 first suggestions given by our system. Considering the first 5 suggestions, more than 90% of the results suggested by our system where correct. The judgment about the relationship between queries was performed by five people from our laboratory. They analised each query and the suggestion provided by the program, assigning as related the suggestions they believed could be interesting for users who formulated the original query.

Suggestions per query	Right	Wrong
5	90.5%	9.5%
10	89.5%	10.5%
15	86.9%	16.1%
20	81.4%	18.6%

Table 2: Top 95 queries suggestions.

Another possible ways of checking the degree of relation between two queries is to evaluate the precision recall curve of the original query, compared against the curve for the related queries. The relevance judgment in this case is always performed considering the user needs in the original query.

Figure 4 shows a graph comparing the top 10 precision of the related queries and of the original queries. In this experiment we submitted the 20 most popular original queries to the search engine Google. For each query we also submitted the top 5 related queries obtained by our method. The answers were given to the user as results for the original query and they have assigned each item as relevant or non relevant for the original query. The results shows that the related queries bring a high concentration of relevant documents, with a result close to the original query. Therefore, the related queries can be use to find new documents which are relevant for the user who formulated the original query.

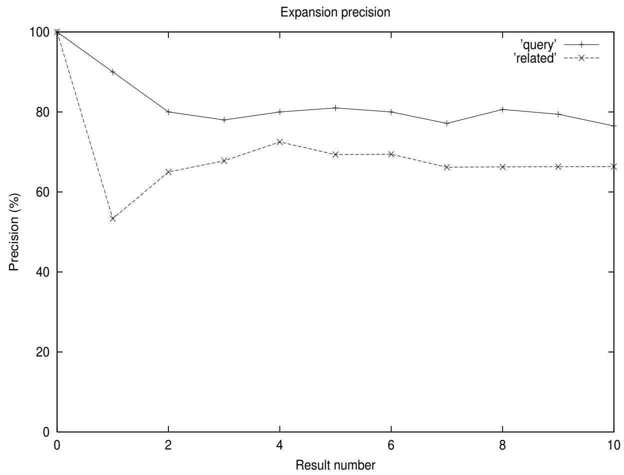

Figure 4: Precision for the related query

Table 3 shows results with queries randomly selected from the logs. This experiment was performed to check the overall performance of our method against the whole set of queries, and not only over common queries. The table shows we got only about 2 suggestions per query on the average, which was expected since we have many low frequency queries in the log. However, the quality of the suggestions presented indicate that our method still useful even for the general case, obtainning 93.45% of sucsseful suggestions.

Total of suggestions	Right suggestions	*Wrong suggestions*
214	93.45%	6.55%

Table 3: Percentage of correct related queries obtained when giving suggestions for 100 randomly selected queries.

4.1 Query Expansion

We also experimented the possibility of using our method as part of a simple query expansion technique. Query expansion methods are used to avoid the necessity of textual matches between the user query and the Web pages searched. The idea of any expansion method is to preprocess the user query in order to find new terms that are related to the user needs and then submmit a query expanded with these new terms.

1042078585.991 3713 200.226.211.142 TCP_MISS/200 25368 GET http://cluster.igbusca-cluster/query.cgi?query=+origem+da+familia+marques - DIRECT/192.168.2.12 text/html

Figure 3: Example of an entry in the query log.

Query	Suggestions
jogos (games)	games (games), sexo (sex), games jogos (games and games in portuguese), gratis jogos (free games), jogo (game)
horoscopo (horoscope)	signos (signs), astrologia (astrology), signo (sign), sexo (sex), horoscopos (horoscopes)
papel de parede (wallpaper)	de protecao de tela (screen protection), "papel de parede" ("wall paper"), de protetor de tela (screen protector), baixaki(an website about wallpapers), wallpaper
musicas (musics)	musica (music), mp3, de letras musicas (musics lyrics), radio (radio), gratis musica (free music)
concursos (concurrence)	concursos publicos (public concurrence), dirigida folha (a famous Brazilian website about public concurrence), concurso (concurrence), concurso do jornal (concurrence newspaper), concurso publico (public concurrence)
receita federal	imposto de renda (income tax), "receita federal"(Brazilian government agency responsible for tax collection), receitafederal, receita(abbreviation of the original query), ministerio da fazenda (Department of Treasury)

Table 1: Related queries examples.

We have implemented and experimented a simple query expansion method which uses the related queries as the expansion for the initial query. In this first experiment, we got the top 20 popular queries, and expanded each query with the top 20 related queries, performing an OR operation between the original query and all suggestions obtained. In this first experiment, we expanded each query with the top 20 related queries, performing an OR operation between the original query and all suggestions obtained. This simple strategy produced an interesting result, indicating that we should study this possibility more carefully in the future.

The Figure 5 shows the precision rates at the top 10 for the original and expanded queries. The superior results achieved by the expanded query indicates that our proposed method may be an useful source of information for query expansion techniques.

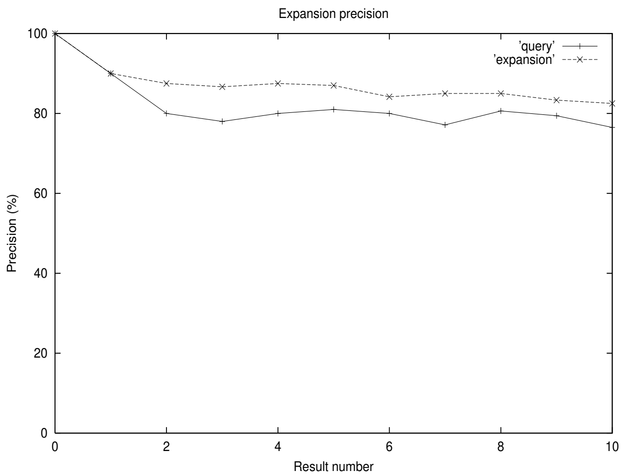

Figure 5: Precision for the expanded query

5. CONCLUSIONS AND FUTURE WORK

In this article we have shown a method for proposing related queries based on the application of association rules over search engines query logs. The method proposed is simple, has low computational cost and presents good results. The experiments presented shows the practical usefullness of the method. We have also experimented our method as an input to a simple query expansion technique. An initial experiment has shown that using just an OR operator between the original and the related queries we can obtain an improvement in the precision. As future work, we are planning to study the idea of expanding queries with our method by performing more detailed experiments and by studying the possibility of new combinations among the original and the related queries. We are also studying the possibility of combining the information extracted form the logs with information extracted from the Web documents to derive new suggestion methods.

6. REFERENCES

[1] AGRAWAL, R., IMIELINSKI, T., AND SWAMI, A. N. Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data* (Washington, D.C., 26–28 1993), P. Buneman and S. Jajodia, Eds., pp. 207–216.

[2] AGRAWAL, R., AND SRIKANT, R. Fast algorithms for mining association rules. In *Proc. 20th Int. Conf. Very Large Data Bases, VLDB* (12–15 1994), J. B. Bocca, M. Jarke, and C. Zaniolo, Eds., Morgan Kaufmann, pp. 487–499.

[3] BAEZA-YATES, R., AND RIBEIRO-NETO, B. *Modern Information Retrieval*. Addison Wesley, Essex, England, 1999. 513 pages.

[4] CALADO, P., RIBEIRO-NETO, B., ZIVIANI, N., MOURA, E., AND SILVA, I. Local versus global link information in the web. *ACM Transactions on Information Systems (TOIS) 21*, 1 (2003), 42–63.

[5] CUI, H., WEN, J.-R., NIE, J.-Y., AND MA, W.-Y. Probabilistic query expansion using query logs. In

Proceedings of the eleventh international conference on World Wide Web (2002), ACM Press, pp. 325–332.

[6] GEYER-SCHULZ, A., AND HASHLER, M. Evaluation of recommender algorithms for an internet information broker based on simple association rules and on the repeat-buying theory. In *Proceedings of WEBKDD'2002* (Edmonton, Canada, July 2002), pp. 100–114.

[7] GLANCE, N. S. Community search assistant. In *Proceedings of the 6th international conference on Intelligent user interfaces* (2001), ACM Press, pp. 91–96.

[8] J. XU, AND CROFT, W. B. Improving the effectiveness of information retrieval with the local context analysis. *ACM Transaction of Information Systems 18*, 1 (2000), 79–112.

[9] JANSEN, B. J., SPINK, A., BATEMAN, J., AND SARACEVIC, T. Real life information retrieval: a study of user queries on the web. *ACM SIGIR Forum 32*, 1 (1998), 5–17.

[10] LIN, W., ALVAREZ, S., AND RUIZ, C. Efficient adaptive-support association rule mining for recommended systems. *Data mining and knoledge discovery 6*, 1 (2002), 83–105.