Modeling User Search Behavior

Ricardo Baeza-Yates, Carlos Hurtado, Marcelo Mendoza and Georges Dupret *
Center for Web Research
Computer Science Department
Universidad de Chile
{*rbaeza,churtado,mmendoza,gdupret*}*@dcc.uchile.cl*

Abstract

Web usage mining is a main research area in Web mining focused on learning about Web users and their interactions with Web sites. Main challenges in Web usage mining are the application of data mining techniques to Web data in an efficient way and the discovery of non trivial user behaviour patterns. In this paper we focus the attention on search engines analyzing query log data and showing several models about how users search and how users use search engine results.

1. Introduction

Commercial search engines usually recommend numerous results associated with each user query. However, scientific literature shows that the average number of visited pages per session is less than 2 and frequently these pages are shown into the top 20 results [14, 20]. Thus, the relevant piece of the result sets consists only about very few pages compared with the size of the complete answer collection. Besides, user search behavior could be compared to TV users navigation behavior in the following sense [22]: with a little effort, Web users want to achieve fast and precise results. Based on these facts, search engines should improve the precision of their results and a new generation of high precision search engines should be designed [6, 5]. Certainly, with a few query words and short searching times associated, this task is more difficult.

We address this problem from the following point of view: only if search engine communities know how users search, they could improve technologies associated with this process. To improve the precision of the results, search engines should consider context information about users such as previous queries and previous clicked documents.

* This research was supported by Millennium Nucleus, Center for Web Research (P01-029-F), Mideplan, Chile.

Scientific literature show several works focused on this problem. Despite this fact, we have little understanding about how search engines affect their own searching process and how users interact with search engine results. This is the main idea of this work: to illustrate the users-search engines interaction as a bidirectional feedback relation. In order to do that, we understand the searching process as a complex phenomenon constituted by multiple factors such as the relative position of the pages in the answer collection, the semantic connections between query words and the time variables involved, among others.

1.1. Contributions

In this study we present a novel approach to describe and analyze the Web searching process. Firstly, we present the pre-processing technique used to work over log data, identifying sessions and also building relational tables to make structured queries over the data. Then, we describe the multiple factors that are related to this process, analyzing the information in a non aggregated way, considering query keywords, clicks, ranking and times involved in query session requests.

Keyword analysis allows to identify properties over the keyword collection that frequently are considered to make queries. We identify top query terms and top queries. Also, we show the sparseness of the keyword query space. Finally we explore the correlation between query term space and document term space.

Click analysis allows to identify the effect of the ranking over the user selections. We show number of queries per query session and number of selection per position, showing the Zipf behavior of the data.

Finally, we introduce three models of user-search engine interactions considering these variables. First, we calculate a predictive model for the number of clicks expected in a session given that the number of queries formulated is known. Second, we calculate a Markov model of transitions in a query session. Also we calculate stationary prob-

Proceedings of the Third Latin American Web Congress (LA-WEB'05)
0-7695-2471-0/05 $20.00 © 2005 IEEE

abilities for the chain. Finally, we calculate a time distribution transition model considering times involved between query formulation and document selection. Results provide enough evidence to be applied to query and document recommendation systems.

1.2. Related Work

Web usage mining papers are focused on techniques that could predict user behavior patterns in order to improve the success of Web sites in general, modifying for example, their interfaces. Some studies are focused also in applications of soft computing techniques to Web usage mining [15, 9], in order to work with vagueness and imprecision in the data. Other kind of works are focused on identify hidden relations in the data. Typical problems related are user sessions and robot sessions identification [8, 24].

Web usage mining works focused on log data could be classified into two categories: based on the server side and based on the client side. Works based on the client side retrieve data from the user using cookies or other methods, such as ad-hoc logging browser plugins. For example, Otsuka *et al.* [17] propose mining techniques to process data retrieved from the client side using panel logs. Panel logs, such as TV audience ratings, cover a broad URL space from the client side. As a main contribution, they could study global behavior on Web communities.

Web usage mining studies based on server log data present statistics analysis in order to discover rules and non trivial patterns in the data. Typically, these kind of works are mainly focused on navigation log data. The main problem is the discovery of navigation patterns. For example, in order to do that, Chen *et al.* [7] assume that users choose the following page determined by the last few pages visited, concluding how well Markov models predict user behavior. Similar studies consider longer sequences of requests to predict user behavior [18] or the use of hidden Markov models [26] introducing complex models of users behavior prediction. In [12] Deshpande and Karypis propose to select different parts of different order Markov models to reduce the state complexity and improve the prediction accuracy. In [20], Silverstein *et al.* present an analysis of the Alta Vista search engine log, focused on individual queries, duplicate queries and the correlation between query terms. As a main result, authors show that users type short queries and select very few pages in their sessions. Similar studies analyze other commercial search engine logs, such as Excite [23] and AskJeeves [21]. These works address the analysis from an aggregated point of view, i.e., present global distributions over the data. Other approaches include novel points of view for the analysis of the data. For example, Beitzel *et al.* [4] introduces hourly analysis. As a main result, authors show that query traffic differ from one topic to other considering hourly analysis, being these results useful for query disambiguation and routing. Finally, Jansen and Spink [13] address the analysis using geographic information about users. Main results of this study show that queries are short, search topics are broadening and approximately the 50% of the Web documents viewed were topically relevant.

In the recent years a few papers have tried to classify the intention of a query. Broder [5] defined three types of needs: informational, navigational and transactional. In [19] the last two were refined in 10 more classes. In [27, 16] simple attributes were used to predict the need behind the query. Our models also shed light in this problem.

Applications of Web query mining covers several subjects such as query recommendation [3], query expansion [11, 10] and document recommendation [2, 25, 28]. For a recently review on applications of query session mining see [1].

1.3. Outline

The remainder of this paper is organized as follows. In section 2 we give activity statistics for the query log files and the search engine, given also some relevant definitions. In section 3, we analyze relations between queries and sessions. In section 4, we work on keyword analysis focused on query term space features. In section 5 we analyze click data. Section 6 shows our user - search engine interaction models considering number of clicks, number of queries in a session and time variables such as time involved in query formulations and document selections. Finally, in section 7 we give some conclusions and we identify avenues of future work.

2. Preliminaries

2.1. Definitions

We will use the following definitions in this paper:

1. *Keyword*: any unbroken string that describe the query or document content. Obviously, the keyword set do not include non useful words like *stopwords*.
2. *Query*: is a set of one or more keywords that represent a user need formulated to a search engine.
3. *Click*: a document selection in the results of a query.
4. *Query instance*: a single query submitted to a search engine in a defined point of time, followed by zero or more clicks.
5. *Query session*: following the definition introduced by Silverstein *et al.* [20], a *query session* consists of a sequence of query instances by a single user made within

a small range of time and all the documents selected in this range of time. The aim of this definition is to identify the user need behind the query, identifying also the refinement query process. As an additional constraint to this definition, a query session starts with a no empty query instance. With this definition, we exclude from the query session set all the query sessions without document selections. In the following we shall call this kind of query sessions as *empty query sessions*.

2.2. Query log data pre-processing

Our log file is a list of all the requests formulated to a search engine in a period of time. Search engine requests include query formulations, document selections and navigation clicks. For the subject of this paper, we only retrieve from the log files query formulations and document selection requests. In the remainder of this study we will call this *query log data*.

Using query log data, we identify query sessions using a query session detection algorithm. Firstly, we identify user sessions using IP and browser data retrieved from each request. Then, a query session detection algorithm determine when a query instance starts a new query session considering the time gap between a document selection and the following query instance. As a threshold value, we consider 15 minutes. Thus, if a user make a query 15 minutes after the last click, he starts a new query session.

We organized the query log data in a relational schema. Main relations on the data model are *query session*, *click*, *query*, *url*, *popuq* (popularity), *queryterm* and *keyword*. Several relations has been formulated in order to answer specific queries to this paper, but they are basically views over the previous relations, thus they are not included in the data model. Figure 1 shows the data model of the query log database.

We processed and stored the query log database in an Athlon-XP at 2.26 GHz, with 1GB of RAM memory and 80 GB of hard disk space, using *MySQL 5.0* as a database engine.

2.3. The search engine and the log data

We work over the data generated by a Chilean search engine called TodoCL (www.todocl.cl). TodoCL mainly covers the .CL domain and some pages included in the .COM and .NET domain that are hosted by Chilean ISP providers. TodoCL collects over three million Chilean Web pages, and has more than 50,000 requests per day being the most important Chilean owned search engine.

The query log file used was collected over 92 days, from 20 April 2004 to 20 July 2004. The table 1 summarize some log file statistics.

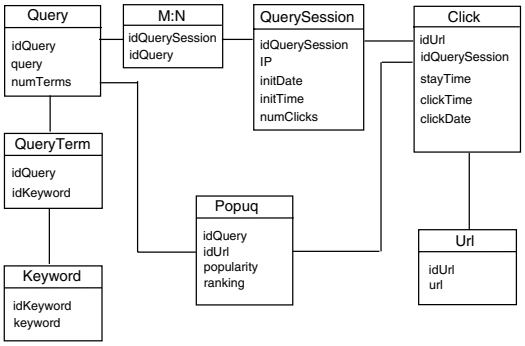

Figure 1. Data model used in the query log database.

Successful requests	4,920,463
Average successful requests per day	53,483
Successful request for pages	1,898,981
Average successful requests for pages per day	20,641
Redirected requests	380,922
Data transferred	66.96 gigabytes
Average data transferred per day	745.29 megabytes

Table 1. Statistics that summarize the contents of the log file considered in our experiments.

3. Sessions and Queries

Following definitions from section 2.1, query logs register 1,524,843 query instances, 1,480,098 query sessions, 102,865 non empty query sessions and 65,282 different queries with at least 1 related click. Also, the logs register 232,613 clicks over 122,184 different documents. The average number of queries for all sessions is 1,037 and 1,435 if we count only non empty sessions. Figure 2 shows the number of new queries registered in the logs.

Another relevant feature of the data is the occurrence distribution of the queries in the log. The figure 3 shows the log plot of the number of queries per number of occurrences.

Figure 3 shows that over the 80% of the queries are formulated only once. The most common queries are shown in table 2. They are similar to most frequent queries in other search engines.

4. Keyword Analysis

An important issue related to this study is to determine some properties over the keyword collection. For example the top query terms used to make queries. The table 3 sum-

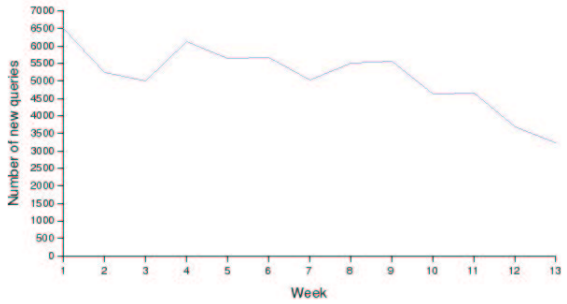

Figure 2. Number of new queries registered in the logs.

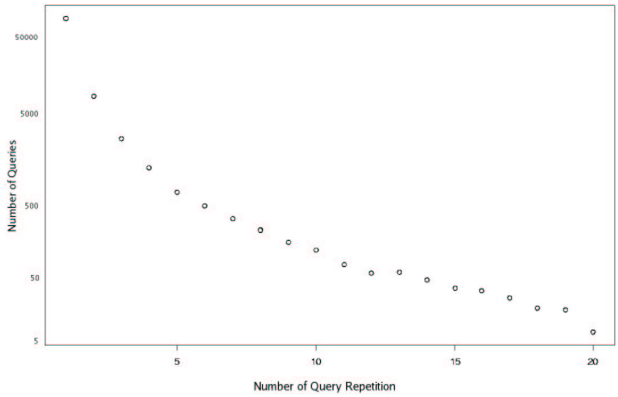

Figure 3. Log plot of number of queries v/s number of occurrences.

marize number of occurrences and normalized frequencies for the top 10 query terms in the log file.

As we can see in the table, some keywords are related to universal topics such as sales, photos or rents. Other keywords are related to domain topics (.CL) such as *Chile*, *Santiago* or *Chilean*, and are equivalent to the 40% of the keyword occurrences over the query log.

As shown in table 2, the most frequent queries do not share some popular keywords. This means that keywords as Chile, Santiago or Chilean appear in many queries, but these queries are not so popular individually. On the other hand, some specific keywords that individually are not so popular (thus, they are not included in the previous table) are popular as queries (for example, chat, Google and Yahoo queries). Finally, some keywords appear in both tables such as *cars* and *house*. These kind of keywords are good

query	occurrences
google	682
sex	600
hotmail	479
emule	342
chat	324
free sex	270
cars	261
yahoo	259
games	235
kazaa	232

Table 2. The top 10 queries.

term	occurrences	frequency
chile	63460	0,34
sale	23657	0,13
cars	10192	0,05
santiago	9139	0,05
radios	7004	0,04
used	6994	0,04
house	6073	0,03
photos	5935	0,03
rent	5352	0,03
Chilean	5113	0,03

Table 3. List of the top 10 query terms sorted by the number of query occurrences in the log file.

descriptors of common need information clusters, i.e., they can be used to pose similar queries (for example, *used car, rent car, old car* and *luxury car*) and, at the same time, have an important meaning as individual queries.

In the query term space, keywords appearing only once represent around 60% of the queries. It is important to see that the query term data follows a Zipf distribution as the log-plot of the data in figure 4 shows. Let X be the number of occurrences of query terms in the whole query log database. Then the number of queries expected for a known number of occurrences in the log is given by:

$$f(X = x) = \frac{1}{x^b}, \quad b > 0\,. \tag{1}$$

Fitting a straight line to the log-plot of the data we can estimate the value of the parameter b that is equal to the slope of the line. Our data shows that this value is 1.079.

It is important to see the data distribution of the terms that not belong to the intersection. The collection formed by

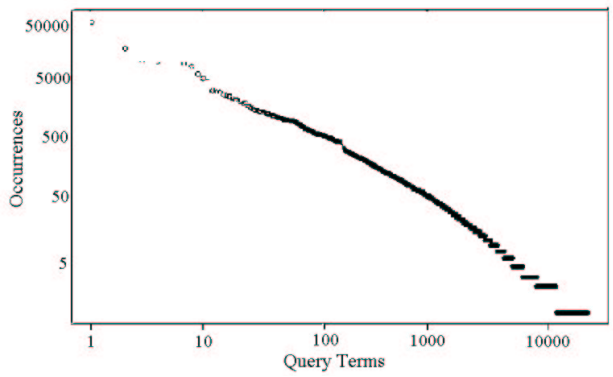

Figure 4. Log-plot of the query term collection.

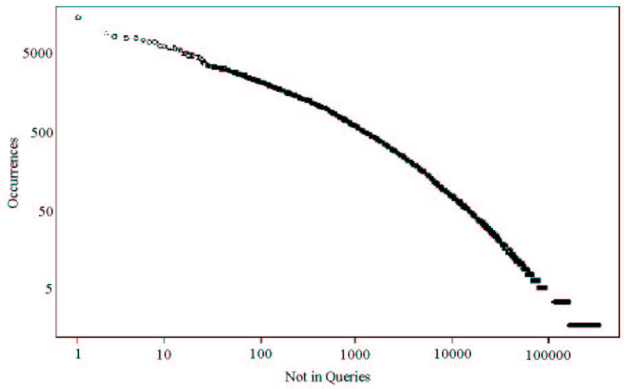

Figure 6. Log-plot of the text terms that do not appear in the query collection.

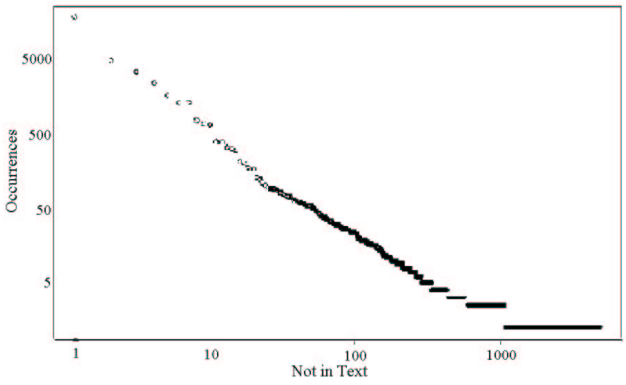

Figure 5. Log-plot of the query terms that do not appear in the text collection.

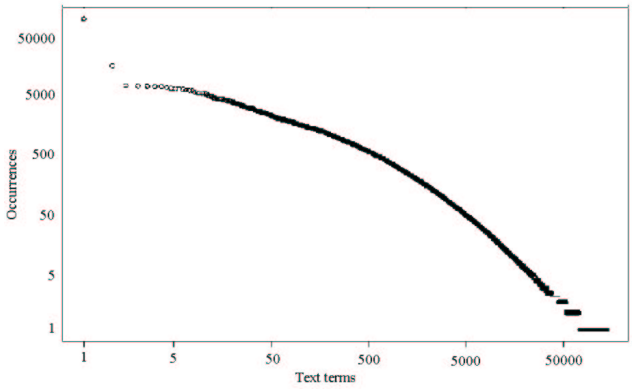

Figure 7. Log-plot of the overall text term collection.

text terms that not belong to the query vocabulary is shown in the log plot of figure 5. On the other hand, the collection formed by query terms that not belong to que text vocabulary is shown in the log plot of figure 6. Both collections show a Zipf distribution, with b values 1.231 and 1.643, respectively. For sake of completeness we include in 7 the distribution of all the terms in the text, which has $b = 1.408$.

Finally, another relevant subject of this study is to show the correlation between query terms and document terms. The query vocabulary has 27,766 terms and the text vocabulary has 359,056 terms. Common terms between both collections are only 22,692. The figure 8 shows an scattering plot of the query term and the text term collection. The plot was generated over the intersection of both collections and comparing the relative frequencies of each term calculated over each collection. As the plot shows, the Pearson correlation between both collections, 0.625, is moderated.

5. Click Analysis

An important information source that can be useful to describe user behavior in relation with their queries and their searches is the click data. Click data could be useful in order to determine popular documents associated to particular queries. Also could be useful to determine the effect of the order relation between documents (ranking) and user pref-

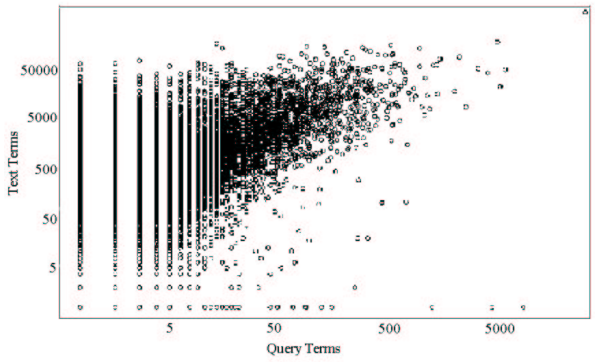

Figure 8. Scattering plot of the intersection between query term and text term collections.

erences.

Firstly, we show the number of selections per query session in the log plot in figure 9. Our data shows that over the 50% of the query sessions have only one click in their answers. On the other hand, only the 10% of the users check over five answers. The average number of clicks over all queries is 0,1525 and 1,57 without including empty sessions. The data follows a Zipf distribution where $b = 1.027$.

One important aspect for this study is to show the effect of the ranking over the user selections. Intuitively, we expect that pages that are shown in better ranking places have more clicks than pages with less score. The position effect is based on the following fact: the search engine shows their results ordered by the relevance to a query, thus, users are influenced by the position at which documents are presented (ranking bias). This phenomenon could be observed in figure 10. As we can see in the figure, the data follows a Zipf distribution where $b = 1.222$. The data shows a discontinuity in positions number ten and twenty. Our data shows that this discontinuity appears also in positions 30 and 40 and, in general, in positions that are multiple of 10. TodoCL shows ten results per page result, thus, this fact cause the discontinuity (interface bias). Finally, the 79.72% of the pages selected are shown in the first result page.

Another interesting variable involved in this process is the visit time per selected page. Intuitively, this variable is important because the time spent in a selected page indicates a preference. ¿From our data we can see that over the 75% of the visit times per page are less than 2 minutes and a half. On the other hand, the others pages show higher visit times. This fact indicate that these pages were very relevant to the their queries and obviously, this information could be used in order to improve their rankings. Besides this, an important proportion of the pages show visit times between 2 and 20 seconds. This represents the 20% of the selected pages, approximately.

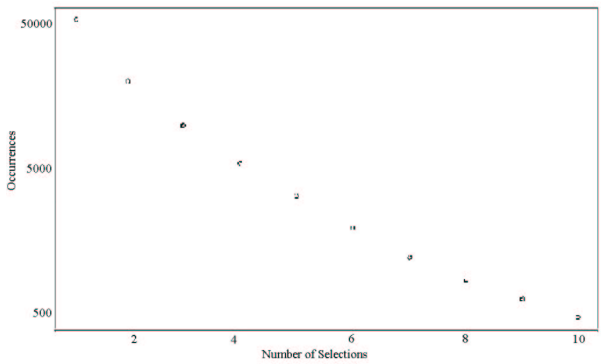

Figure 9. Log plot of number of selections per query session.

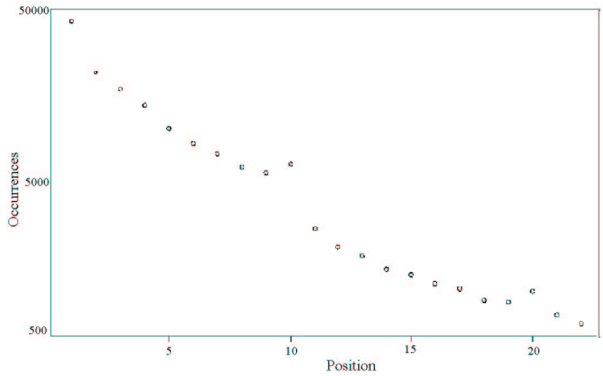

Figure 10. Log plot of number of selections per position.

6. Query Session Models

6.1. Aggregated query session model

One of the main purposes of this study is to describe user behavior patterns when a query session is started. We consider only non empty query sessions in our study, thus query instances with at least one page selected in their answers. We have focused this analysis considering two independent variables: number of queries formulated and number of pages selected in a query session.

In a first approach, we calculated a predictive model for the number of clicks in a query session when the total amount of queries formulated is known. Let $X = x$ be the random variable that models the event of formulating x queries in a query session and let $Y = y$ be the random variable that models the event of selecting y pages in a query session. This first approach models the probability of selecting y pages given that the user has formulated x queries using conditional probability $P(Y = y \mid X = x) = \frac{P(Y=y, X=x)}{P(X=x)}$. In order to do that, we consider the total amount of occurrences of the events in the query session log files. Thus, the data is depicted in an aggregated way, i.e. this first approach considers a predictive model for the total amount of queries and clicks in a query session. The figure 11 shows the first model.

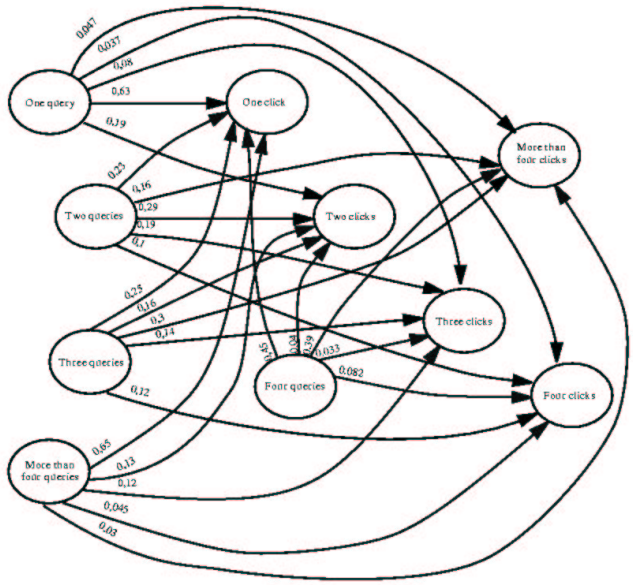

Figure 11. Predictive model for the number of clicks in a query session with a known number of queries formulated.

As we can see, if a user formulates only one query, in general he will select only one page. Probably, this fact is caused by a precise query, i.e., the user has the need defined at the begin of the query session. As a consequence of the quality of the search engine, when the query is very precise, the user finds the page in few trials. We will say that this kind of queries are of good quality, because they enable users to find their pages quickly. If the session has two, three or four queries, probably the session will have many clicks. In general, this kind of sessions are associated to users that do a more exhaustive search pattern, and, as a consequence, they have less defined information needs at the begin of the session. Finally, if the session has more than four queries, probably users will select only one result. In general, these sessions show less specific queries at the begin than at the end. So, the user had the need to refine the query in order to find the page that he finally selects. This kind of sessions are related to users that do not formulate good quality queries.

6.2. Markovian query session model

Focused in transitions between frequent states in a query session, in this second approach we build a finite state model. Using the query log data, we have calculated probabilities between states making a Markovian query session model. In order to do that, we have considered the following states:

- *M-th query formulation*: m-th query formulated in the query session.
- *N-th document selection*: n-th document selected (n-th click) for the m-th query formulated in the query session.

States labeled as n-th document selection are related to a given query formulated. States labeled as n-th query formulation are related to a previous document selection, except the first query that starts the query session. Let X_i be the random variable that model the number of queries formulated after i transitions in a query session and let Y_i be the random variable that model the number of document selected for the last query formulated in the session after i transitions. Thus, states in a query session model could be modeled as a pair $(X_i = x, Y_i = y)$ representing that the user has selected y documents after formulating the x-th query.

In order to calculate it, we consider the data in the query log files in a non aggregated way, as the opposite of the first approach. Thus, the events are distributed over the time. As a consequence, the second model is a discrete-state, continuous-time Markov Model, being the probability of be over a defined state $(X_i = x, Y_i = y)$ determined by the Markov chain of the $i - 1$ previous states.

We have considered the same events as in the first approach but focused on two kind of transitions: the first one considers the probability of doing the m-th query given that in the $m - 1$-th query users has selected n pages. The second one considers the probability of doing the n-th click given that the user has formulated the m-th query. We calculate these as follows:

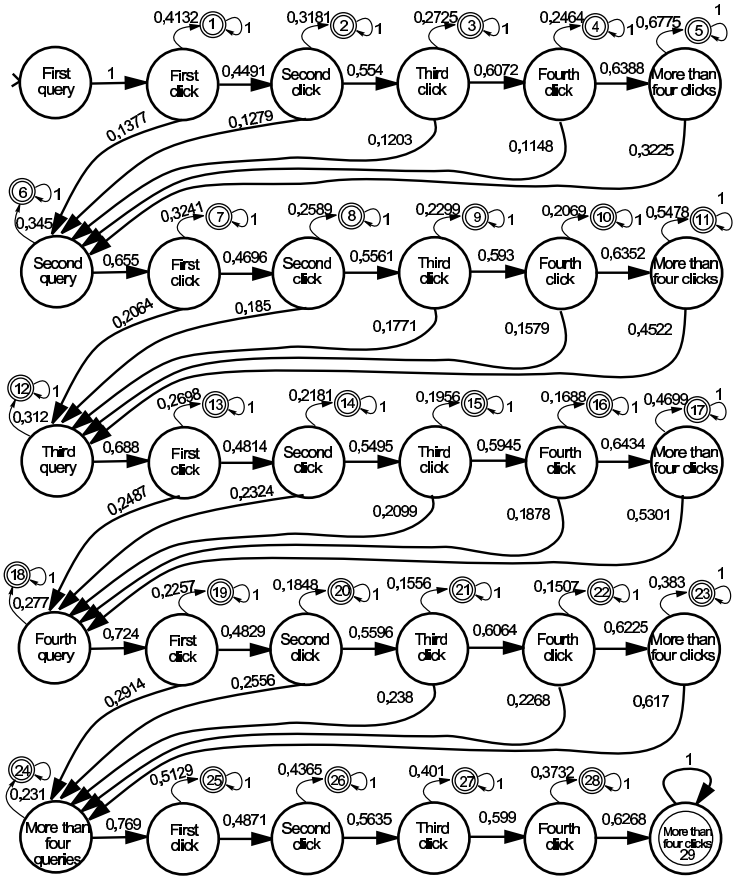

Figure 12. Markovian transition query session model.

$$P(X_i = m \mid X_{i-1} = m-1, Y_{i-1} = n) = \frac{P(X_i=m, Y_i=n)}{P(X_{i-1}=m-1, Y_{i-1}=n)},$$

$$P(Y_i = n \mid X_{i-1} = m, Y_{i-1} = n-1) = \frac{P(X_i=m, Y_i=n)}{P(X_{i-1}=m, Y_{i-1}=n-1)}.$$

The initial event in query sessions is to formulate the first query and to make the first click. After k steps, the probability of be over a determined state could be calculated using the Markovian transition matrix, M, of k-th order, M^k, and the initial distribution vector $P^{(0)}$ formed by the probabilities of being at determined states at the beginning of the process. The set of probabilities of be over a state in the chain compounds the probability vector of k order given by $P^{(k)} = M^k \times P^{(0)}$. After a large number of transitions, the $P^{(k)}$ vector converges to the $\vec{v}$ fixed point vector of the process, the stationary probabilities of the chain, that satisfy the fixed point property $\vec{v} = M \times \vec{v}$. Of course, the fixed point vector has positive values only for final states.

Using our model, we verify that after 10 steps the $P^{(k)}$ vector converge to the fixed point vector. As an example of the convergence, in the following matrix the i-th row represents the probabilities after $i+1$ steps. Each column represents final states, considering only states from 1 to 8, because the probability of be over all the other states is close to zero. We show only the first five steps and we start from $k = 2$ (for $k = 1$ the vector is null and only the transitive state $(X_1 = 1, Y_1 = 1)$ is attainable):

$$\begin{pmatrix} 0,41 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0,41 & 0,14 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0,41 & 0,14 & 0,07 & 0,05 & 0,04 & 0,07 & 0,03 & 0 \\ 0,41 & 0,14 & 0,07 & 0,04 & 0,05 & 0,08 & 0,04 & 0,01 \\ 0,41 & 0,14 & 0,07 & 0,04 & 0,05 & 0,08 & 0,05 & 0,02 \end{pmatrix}$$

As we can see, at the begin, only the first states are possible but after few steps, probabilities are propagated over the complete chain. Finally, the fixed point vector for all the final states is:

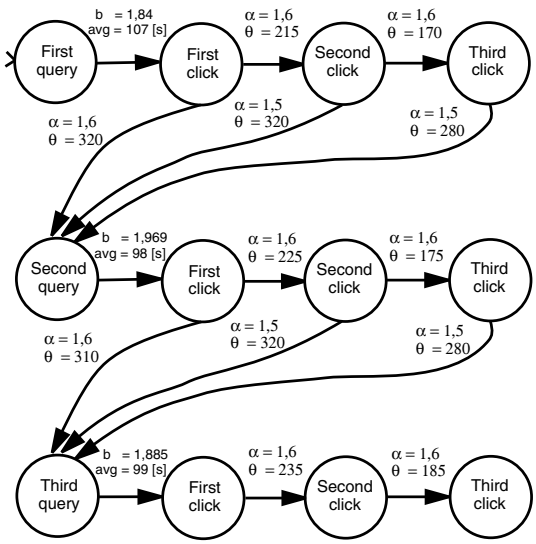

Figure 13. Time distribution query session model.

$$\begin{aligned}\vec{v} \;=\; & (0,41; 0,14; 0,07; 0,04; 0,05;\\ & 0,08; 0,05; 0,02; 0,01; 0,005; 0,01\\ & 0,02; 0,01; 0,005; 0,002; 0,002; 0,005\\ & 0,015; 0,007; 0,006; 0,003; 0,001; 0,004\\ & 0,007; 0,005; 0,001; 0,001; 0; 0,01).\end{aligned}$$

In figure 12 we show the second model. Final states are depicted with concentric circles.

6.3. Time distribution query session model

In a third model we address the problem of determine the time distribution between state transitions. Each transition time follows a distribution that could be determined using the log data. In order to do that, we measure the time between events in the query logs considering two kinds of transitions: the time distribution in query sessions between the query formulation and the following document selections and the time distribution between the clicks and the following query formulation.

For transitions between clicks and the followings query formulations, the data shows that users need to take higher times in order to formulate their queries. Intuitively, a query formulation time is distributed around an expected value and for higher times the probability density follows an exponential distribution. To calculate time distribution for this kind of transitions, we use the two parameter Weibull density function. Let t be the random variable that models the time involved in a query formulation. Then the probability density function of t is given by $f(t;\alpha,\theta) = \frac{\alpha}{\theta^\alpha} t^{\alpha-1} e^{-(\frac{t}{\theta})^\alpha}, \quad t > 0, \alpha, \theta > 0$. In figure 13 we show the α, θ parameters values for each transition.

To - From	First click	Second click	Third click
Second click	192	201	210
Third click	151	156	165
Second query	286	288	252
Third query	277	288	252

Table 4. Expected values (in seconds) for the Weibull distributions involved in the query session model.

Transitions between queries and clicks are highly correlated with the search engine interface effect. As we saw in section 5, the search engine interface produces a bias in the searching process caused by the relative position of the documents in the ranking. Intuitively, we can guess that times involved in the searching process are also correlated with the ranking. As we can see in the data, this assumption is true for the first click and these kind of transitions follows Zipf distributions. In figure 13 we show the b parameter value for each transition considered. We can see that values are independent of the query order. However, for transitions to second or higher order clicks, time distribution follows a Weibull as in the previous case. Intuitively this is a consequence of the searching time involved that changes the expected value from zero to higher values. As is well known, the expected value of a Weibull random variable is given by $\theta \times \Gamma\left(1 + \frac{1}{\alpha}\right)$. The expected values for each transition are given in table 4.

Times involved in query formulation are higher if the preceding states are document selections of low order. It is important to see that all these times are biased to the previous viewing time involved. In order to unbias the results, we must subtract the expected values for the viewing document times to the expected values of query formulations. However expected values for Zipf distributions are close to zero, thus the subtraction do not affect final results.

7. Conclusion

We have proposed models to describe user behavior in query sessions using query log data. The models proposed are simple and provides enough evidence to be applied to Web search systems. Our results show that Web users formulate short queries, select few pages and an important proportion of them refine their initial query in order to retrieve more relevant documents. Query log data, in general, show Zipf distributions such as for example clicks over ranking positions, query frequencies, query term frequencies and number of selections per query session. Moreover query

space is very sparse showing for example that in our data the 80% of the queries are formulated only once.

Currently, we are working on the use of this knowledge in query recommendation systems and document ranking algorithms. Using data mining techniques such as clustering we can identify semantic connections between queries solving the problem of the sparseness in the data. Then, using no sparse data we can build query recommendation systems based on similarity measures between queries [3]. Besides this, if we consider documents associated to each cluster we can build document recommendation systems [2].

Future work includes studies about features of information needs dynamics, for example how user needs changes in time and how this knowledge could be applied to improve the precision in Web search engines results.

References

[1] R. A. Baeza-Yates. Applications of web query mining. In *ECIR 2005, Santiago de Compostela, Spain, March 21-23*, volume 3408 of *Lecture Notes in Computer Science*, pages 7–22. Springer, 2005.

[2] R. A. Baeza-Yates, C. A. Hurtado, and M. Mendoza. Query clustering for boosting web page ranking. In *AWIC 2004, Cancun, Mexico, May 16-19*, volume 3034 of *Lecture Notes in Computer Science*, pages 164–175. Springer, 2004.

[3] R. A. Baeza-Yates, C. A. Hurtado, and M. Mendoza. Query recommendation using query logs in search engines. In *EDBT 2004 Workshops, Heraklion, Crete, Greece, March 14-18*, volume 3268 of *Lecture Notes in Computer Science*, pages 588–596. Springer, 2004.

[4] S. M. Beitzel, E. C. Jensen, A. Chowdhury, D. Grossman, and O. Frieder. Hourly analysis of a very large topically categorized web query log. In *SIGIR'04*, pages 321–328, New York, NY, USA, 2004. ACM Press.

[5] A. Z. Broder. A taxonomy of web search. *SIGIR Forum*, 36(2):3–10, 2002.

[6] A. Z. Broder and A. C. Ciccolo. Towards the next generation of enterprise search technology. *IBM Systems Journal*, 43(3):451–454, 2004.

[7] Z. Chen, F. Lin, H. Liu, W.-Y. Ma, and L. Wenyin. User intention modelling in web applications using data mining. *World Wide Web*, 5(2):181–192, 2002.

[8] R. Cooley, B. Mobasher, and J. Srivastava. Data preparation for mining world wide web browsing patterns. *Knowledge and Information Systems*, 1(1):5–32, 1999.

[9] F. Crestani and G. Pasi. Handling vagueness, subjectivity, and imprecision in information access: an introduction to the special issue. *Inf. Process. Manage.*, 39(2):161–165, 2003.

[10] H. Cui, J. Wen, J. Nie, and W. Ma. Probabilistic query expansion using query logs. In *Proceedings of the 11th WWW conference*, pages 325–332. ACM Press, 2002.

[11] H. Cui, J. Wen, J. Nie, and W. Ma. Query expansion by mining user logs. *IEEE Transaction on Knowledge and Data Engineering*, 15(4):829–839, July/August 2003.

[12] M. Deshpande and G. Karypis. Selective markov models for predicting web-page accesses. In *1st SIAM Data Mining Conference, April 5-7, Chicago, USA*, 2001.

[13] B. J. Jansen and A. Spink. An analysis of web searching by european alltheweb.com users. *Information Processing and Management*, 41(2):361–381, 2005.

[14] M. Jansen, A. Spink, J. Bateman, and T. Saracevic. Real life information retrieval: a study of user queries on the web. *ACM SIGIR Forum, 32(1):5-17*, 1998.

[15] M. Lalmas. *Information Retrieval: Uncertainty and Logics.* Kluwer Academic Publishers, Norwell, MA, USA, 1998.

[16] U. Lee, Z. Liu, and J. Cho. Automatic identification of user goals in web search. In *Fourteenth International World Wide Web Conference, May 10-14, Chiba, Japan*, pages 391–400. ACM Press, 2005.

[17] S. Otsuka, M. Toyoda, J. Hirai, and M. Kitsuregawa. Extracting user behavior by web communities technology on global web logs. In *DEXA*, pages 957–988, 2004.

[18] J. Pitkow, H. Schtze, T. Cass, R. Cooley, D. Turnbull, A. Edmonds, E. Adar, and T. Breuel. Personalized search. *Commun. ACM*, 45(9):50–55, 2002.

[19] D. Rose and D. Levinson. Understanding user goals in web search. In *Thirteenth International World Wide Web Conference, May 17-22, 2004, New York, USA*, pages 13–19. ACM Press, 2004.

[20] C. Silverstein, H. Marais, M. Henzinger, and M. Moricz. Analysis of a very large web search engine query log. *SIGIR Forum*, 33(1):6–12, 1999.

[21] A. Spink and O. Gunar. E-commerce web queries: Excite and ask jeeves study. *First Monday*, 6(7), 2001.

[22] A. Spink, D. Wolfram, B. Jansen, and T. Saracevic. Searching the web: the public and their queries. *Journal of the American Society of Information Science and Technology*, 52(3):226–234, 2001.

[23] J. L. Xu and A. Spink. Web research: The excite study. In *Proceedings of WebNet 2000, San Antonio, Texas, USA, October 30 - November 4*, pages 581–585. AACE, 2000.

[24] G. Xue, H. Zeng, Z. Chen, W. Ma, and C. Lu. Log mining to improve the performance of site search. In *WISE Workshops 2002*.

[25] G.-R. Xue, H.-J. Zeng, Z. Chen, Y. Yu, W.-Y. Ma, W. Xi, and W. Fan. Optimizing web search using web click-through data. In *CIKM '04*, pages 118–126, New York, NY, USA, 2004. ACM Press.

[26] A. Ypma and T. Heskes. Automatic categorization of web pages and user clustering with mixtures of hidden markov models. In *WEBKDD 2002, Edmonton, Canada,July 23*, volume 2703 of *Lecture Notes in Computer Science*, pages 35–49. Springer, 2003.

[27] H.-J. Zeng, Q.-C. He, Z. Chen, W.-Y. Ma, and J. Ma. Learning to cluster web search results. In *SIGIR '04*, pages 210–217, New York, NY, USA, 2004. ACM Press.

[28] D. Zhang and Y. Dong. A novel web usage mining approach for search engines. *Computer Networks Elsevier*, pages 303–310, April 2002.

0-7695-2471-0/05 $20.00 © 2005 IEEE