Identifying the Intent of a User Query Using Support Vector Machines

Conference Paper · August 2009

DOI: 10.1007/978-3-642-03784-9_13 · Source: DBLP

2 authors:

Marcelo Mendoza
Universidad Técnica Federico Santa María
112 PUBLICATIONS 4,692 CITATIONS

SEE PROFILE

Juan Zamora
Pontificia Universidad Católica de Valparaíso
15 PUBLICATIONS 69 CITATIONS

SEE PROFILE

Identifying the intent of a user query using support vector machines

Marcelo Mendoza[1] and Juan Zamora[2]

[1] Yahoo! Research Latin America, Chile
[2] Applied Computational Intelligence Lab (INCA), Department of Informatics, Universidad Técnica Federico Santa María, Chile

Abstract. In this paper we introduce a high-precision query classification method to identify the intent of a user query given that it has been seen in the past based on informational, navigational, and transactional categorization. We propose using three vector representations of queries which, using support vector machines, allow past queries to be classified by user's intents. The queries have been represented as vectors using two factors drawn from click-through data: the time users take to review the documents they select and the popularity (quantity of preferences) of the selected documents. Experimental results show that time is the factor that yields higher precision in classification. The experiments shown in this work illustrate that the proposed classifiers can effectively identify the intent of past queries with high-precision.

1 Introduction

With advances in technology and world wide access to Internet, user needs have gone beyond simple informational needs. The web, growing in size and complexity, offers more resources and services that make it more difficult for search engines to find precise results for their users. In this sense, the attention of the web community has focused on identifying the intent of a user query. With this, the goal is for search engines to use different ranking functions depending on the type of intent detected.

To develop a coherent framework that points to the intent of a user query, also known as user goals, through the queries they make, several authors have risen to the challenge of defining web search taxonomies. In a first approximation, Broder [3], and later Rose and Levinson [15], have consolidated widely-accepted taxonomies of intents of user queries. These advances have concluded that the intents of user queries are related to the type of interaction or use the users wish to have with the selected resource.

Later, works based on these taxonomies have taken on the problem of constructing query classifiers. Using different information sources, among them, text [11, 1], click-through data [14], or combinations of both sources [12], have shown results with diverse outcomes.

1.1 Contributions

In this work we propose three vector representations of queries based on text and click-through data that allow for query classification by intent. We will consider the taxonomy proposed by Broder (informational, navigational, transactional) and for the classification process we will use techniques based on support vector machines (SVMs) that learn to classify past queries. We will show that the proposed methods achieve high-precision results for all the categories considered.

1.2 Outline

The rest of this work is organized as follows: In section 2 we review related work. In section 3 the vector representations of the queries presented in this work are introduced and the characteristics of the classification technique used are described in detail. In section 4 the experimental design is presented, along with analysis of the results obtained. Finally, conclusions are drawn and future work outlined in Section 5.

2 Related Work

When a user formulates a query in a search engine, their objective is not just to find information about a certain topic, but also to find a site or interact in some way with the suggested results (e.g. read a document, download files, purchase a book, among other interactions). Considering this, Broder [3] proposed a taxonomy of web searches consisting of three categories: *informational*, *navigational* and *transactional*. Broder understood that informational queries were those where the user's intents was to find information related to a specific topic. Navigational queries were those where the user's intents was to find a certain site. Finally, transactional queries were those where the user's intents was to complete some type of transaction on a website, that is, download a resource, or make a purchase, among others. Through an experiment based on the opinion of experts, a set of queries were classified using the proposed categories. As a result, the navigational, informational, and transactional categories were distributed in 20%, 50% and 30%, respectively. Later this taxonomy was extended by Rose and Levinson [15], who developed a framework for classification of search goals and illustrated how to use this framework to manually classify web queries from a search engine.

Once the categories were consolidated by Broder and subsequently by Rose and Levinson, the design and implementation of methods for the automatic classification of queries according to their intents became an important part of many investigations. Kang and Kim [11] proposed the construction of classifiers that characterized queries according to the distribution of query terms. To achieve this, over a set of queries classified by experts, they obtained two collections of terms frequently used in writing informational and navigational queries. By measuring mutual information between the two collections and features such as the distance between the terms in a query and the terms in the titles and snippets of the selected documents, they were able to determine if a query is for general use or if it is informational or navigational. Despite the 80% precision rate this classifier reached, the results can not be considered conclusive because of the small volume of data used (only 200 queries were extracted and labeled from the TREC collection).

Lee *et al.* [12] enumerated bias levels in click distributions as classification features. Intuitively, an informational query should have more clicks concentrated in lower-ranked items, as opposed to navigational queries which are expected to have more clicks in the highest-ranking positions and, in general, they have only one click if they are successful. Using a set of queries classified by a group of experts, they evaluated the precision of a classifier based on these features and obtained a 54% precision for 50 queries manually labeled.

A similar focus was used by Liu *et al.* [14] to build a query classifier. They proposed two features based on click-through data that allowed query characterization: nRS (number of query sessions that register clicks before a position n in the ranking data) and nCS (number of query sessions registering less than n clicks). Using a decision tree as a classifier, their precision rate achieved was almost 80% for a set of 400 manually labeled queries.

Baeza-Yates *et al.* [1] proposed to analyze three categories: informational, equivalent to the category defined by Broder, non-informational, which considered Broder's navigational and transactional categories; and an ambiguous category that included queries whose intention was difficult to perceive based solely on the query terms, such as polysemic queries. Through an experiment based on expert opinion, a set of queries were classified using the proposed categories. As a result, the informational, non-informational, and ambiguous categories were distributed in 61%, 21% and 18% respectively. Later they establish the intent of a user query by analyzing the relationship between queries and 16 categories from the Open Web Directory (ODP). Using techniques such as Support Vector Machines (SVMs) and Probabilistic Latent Semantic Analysis (PLSA), they reached 60% precision (approximately) on a dataset of 6,000 queries semi-automatically classified into the Broder categories (the vector representations were clustered and then those clusters were labeled).

Recently, Jansen *et al.* [10], using a classifier based on query features drawn from logs, such as query terms, IP numbers, and length of the query, achieved a precision of 74% using a training data set of 400 queries classified by experts.

3 The Classifiers

In this section we introduce three vector representations of queries, based on a combination of two information sources: text and click-through data. The idea is to be able to represent queries in the vector space formed by the collection of query terms and the descriptive terms of the documents selected in the query sessions. The vector representations consider variables extracted from click-through data as influential factors for each term. Our idea is to model queries by means of the variables that influence user preferences. In order to do this we will consider only the terms that the user read before their selections, considering also the time spent reading the selected documents and the clicks that the documents register in the query sessions. These variables are combined differently to see their effect on the precision of the classifiers. Finally, in this section we will detail the characteristics most relevant to the classification technique used.

3.1 Vector representation based on descriptive text and clicks

Following Wen *et al.* [17], a *query session* consist of a query instance and the URLs the user clicked on. Thus:

$$\texttt{querySession} := < \texttt{query}, (\texttt{clickedURL})^* > .$$

In the strict sense of the definition, each query session represents a query instance formulated by an anonymous user in a defined point of time. As an extension to this idea, we will represent queries by means of the set of query sessions where the query was formulated.

The list of results shown by a search engine to a user describes each recommended page / site with the following three text components: the page / site title, the URL and the *snippet* or extract of the document content, often the header or dynamic summary. If at least one of the three components is related to the meaning of the query, this will be selected. According to this idea, we propose using a vector representation of queries based on a variation of the $\texttt{Tf} - \texttt{Idf}$ schema, where the vocabulary will be generated by the terms in the titles, URLs and snippets of the selected documents.

Given a query q, we use S_q to denote the set of sessions in which q has been formulated. Let D_S be the set of documents selected in S_q. The influence of each descriptive term on the vector representation of q will be proportional to the number of occurrences of that term in each document d of D_S (factor $\texttt{Tf}$). It will also be proportional to the fraction of clicks of each document d of D_S calculated over the clicks registered in S_q (factor $\texttt{Pop}_{d,q}$). Based on these facts, the component associated with the i-th vocabulary term in the vector representation of a query q will be given by:

$$q_{[i]} = \sum_{d \in D_S} \texttt{Pop}_{d,q} \cdot \frac{\texttt{Tf}_{i,d}}{\max_l \texttt{Tf}_{l,d}}, \tag{1}$$

where the second quotient is the factor $\texttt{Tf}$ normalized by the maximum frequency calculated for all the terms mentioned in the descriptive text of d, and $\texttt{Pop}_{d,q}$ is the fraction of clicks to d in the clicks registered in S_q. According to this vector representation, a term would have greater influence for q to the degree that the term has a greater number of occurrences in the descriptive text of the document (factor $\texttt{Tf}$) and the document registers more preferences in the sessions of q (factor $\texttt{Pop}_{d,q}$). That is, $\texttt{Pop}_{d,q}$ plays the role of *Idf* in the well-known $\texttt{Tf} - \texttt{Idf}$ weighting scheme for the vector model. Finally, the component $q_{[i]}$ is calculated considering all the documents selected in the sessions of q where the term was used.

3.2 Vector representation based on descriptive text and reading time

Claypool *et al.* [5], in the context of general web browsing, show how implicit interest measures are related to the interests of users. Comparing data about implicit interest indicators and explicit judgments of Web pages visited, they found that the time spent on a page has a strong correlation with explicit interest. In the same sense, Fox *et al.* [8] showed that the time spent on the search result page and the click-through data are the best predictors of user's satisfaction. Moreover, they proved that combinations of these implicit relevance feedback measures are useful to predict the quality of search results.

From the above, we will introduce a new vector representation of queries that combines both descriptive text and the time spent reading each selected document. In this section we will combine only descriptive terms and reading time. In the following section we will use these variables with the $\texttt{Pop}_{d,q}$ factor introduced in Equation 1. The same as in the vector representation introduced previously, the vocabulary we will represent in the queries will be formed by all the terms that make up the page titles, URLs, and *snippets*.

Given a query q and the set of sessions S_q in which q has been formulated; D represents the collection of documents selected and registered in the log; N_D represents the size of D; and D_S is the set of documents selected in S_q. Let Q be the set of queries formulated and registered in the log and N_Q be the size of Q. With t_d we refer to the average reading time spent on document d calculated over S_q sessions. t_S represents the total duration of all the sessions in S_q. The $q_{[i]}$ component associated with the i-th term in the vocabulary of the vector representation of q will be given by:

$$\begin{aligned} q_{[i]} = {} & \left(0.5 + 0.5\,\frac{\texttt{Tf}_{i,q}}{\max_l\ \texttt{Tf}_{l,q}}\right) \times \log\frac{N_Q}{n_{i,Q}} \\ & + \sum_{d\,\in\,D_S} \frac{\texttt{Tf}_{i,d}}{\max_k\ \texttt{Tf}_{k,d}} \times \frac{t_d}{t_S} \times \log\frac{N_D}{n_{i,D}}, \end{aligned} \tag{2}$$

where $\texttt{Tf}_{i,q}$ and $\texttt{Tf}_{i,d}$ represent the number of occurrences of the term in query q and in document d, respectively, and $n_{i,Q}$ and $n_{i,D}$ represent the number of queries and the number of documents in which the term appears, respectively.

The first part of Equation 2 corresponds to the modified schema `Tf-Idf` for queries introduced by Salton and Buckley [16], which allow us to incorporate query terms such as descriptive text. The second part represents the effect of the page's descriptive text on reading time. Intuitively, a term will have greater influence on the vector representation of q as the term has more occurrences in the descriptive text of the selected document (factor `Tf`) and the user has invested more time in reading it (factor $\frac{t_d}{t_S}$). As the time spent in each query differs by query type (for example, the time spent viewing the answers of a closed-class question like "Barack Obama's 47th birthday" is less than a general information query like "History of United States"), we normalize t_d using t_S, calculating the time factor as the fraction of time spent in d over the time spent in the sessions of q.

Finally, in Equation 2, the inverse frequency of the term in the collection D (factor `Idf`) has been considered to give more or less relevance to the terms with greater or lesser frequency in the set. The influence $q_{[i]}$ is calculated considering all the documents selected in the sessions of q where the term is used.

3.3 Vector representation based on descriptive text, reading time and clicks

This query representation corresponds to a combination of the factors considered in Equations 1 and 2. For this representation we have also considered the vocabulary formed by the page's descriptive text (titles, URLs and snippets). Regarding this set of terms, we have considered the variables of reading time and clicks. The same as in the representation of Equation 2, the descriptive text models the attraction effect that triggers the selection. The incorporation of factor $\texttt{Pop}_{d,q}$ aims to give greater relevance to the terms used in documents that have been selected in other sessions. Also, the reading time variable is still considered to give more influence to the terms used in documents that have attracted more user attention.

According to the above, the component $q_{[i]}$ associated to the i-th term in the vector representation of q is given by:

$$\begin{aligned} q_{[i]} = & \left(0.5 + 0.5\,\frac{\texttt{Tf}_{i,q}}{\max_l \texttt{Tf}_{l,q}}\right) \times \log\frac{N_Q}{n_{i,Q}} \\ & + \sum_{d \in D_S} \frac{\texttt{Tf}_{i,d}}{\max_k \texttt{Tf}_{k,d}} \times \frac{t_d}{t_S} \times \texttt{Pop}_{d,q} \times \log\frac{N_D}{n_{i,D}}, \end{aligned} \tag{3}$$

The second sum in the expression represents the influence of the term according to the attractiveness of the descriptive text in the document selections. According to this representation, a term will have greater influence on the vector representation of q in so far as it has more occurrences in the descriptive text of d, the average reading time of d is significant with respect to t_S, and d registers an important amount of preferences in the sessions of q.

3.4 Classification Technique

Since the vector representations in Equations 1, 2, and 3 are calculated based on large term collections, it is necessary to use a classification technique that behaves well with high-dimensional data. In this context the support vector machines (SVMs) [6] have proven to be useful in processing high-dimensional vectors (over 10^6 features) even when the vectors are sparse, such as in the case of text [2]. Leopold and Kindermann [13] showed that SVMs even perform well in categorizing documents without pre-selecting features (*pre-filtering*), which means they compare favorably to other techniques like Decision Trees (e.g. C4.5), Artificial Neural Networks (ANN) or Bayesian Networks. Due to the fact that we are using high dimensional term vectors we will use SVMs.

As has been argued in previous works [13, 2, 1], it is recommendable to use radial basis function as a kernel for the setup of the SVM function, since this has shown good performance in text categorization.

4 Experimental Results

4.1 Data set

For this paper we have processed a commercial search engine log. The file corresponds to a period of 3 months from the year 2006 and contains 594,564 queries associated to 765,292 sessions. The log also contains 1,124,664 clicks on 374,349 different URLs. To construct the query vector representations based on text, the queries and descriptive text have been processed, eliminating accent marks, punctuation, and the top-50 stopwords. Using the log file we also estimate the time spent in each document visit, calculating the time gap between consecutive selections in the same query session. For the last click, we estimate the reading time as the average time spent in the query session.

Experts from our labs have manually classified a set of 2,000 queries considering the categories proposed by Broder. The queries considered are the top-2,000 most frequent queries of the query log analyzed. They are associated to 126,287 sessions. The experts had to respond to questions focused on identifying the intent of a user query, similar to those formulated in Broder's experiments. The questions asked were able to identify if the intent was to find a particular site or topic, if the desired results should be found in one website or in many, and finally, if the goal of the query was to read the results obtained or to interact in some other way with the resource, allowing later categorization in the taxonomies considered in this experiment.

The results obtained have been used in the following way: the definitive set of queries considered in the experiments is formed by those that have been classified in the same categories by all the experts. That is, those whose intent has been determined by consensus. Those queries that were classified in two or more categories were revised again by the experts in a second review. As a result of the classification process, 1,953 queries were labeled by consensus, distributed in the different categories as follows: the informational, navigational and transactional categories were distributed in 52%, 33% and 15%, respectively.

70% (1,367 queries) of the manually classified queries were considered as training data, leaving the remaining 30% for evaluation (586 queries). This partition was calculated using a simple random sample in each category in order to preserve the original distribution determined by the experts.

4.2 SVM tuning

The SVM implementation called LIBSVM, developed by Chang & Lin [4], which is freely available, was used. The version of SVM used is that proposed by Cortes and Vapnik [6] known as C-SVM since the associated optimization problem is parameterized by a penalty factor C, $C > 0$. Since we will use radial basis functions ($K(x, y) = e^{-\gamma\|x-y\|^2}, \gamma > 0$) where the variables (x, y) represent labeled instances of the data (queries in our case), a second parameter is added to the problem, γ, which models the length of the radial basis.

The parameter tuning process proposed by Hsu *et al.* [9] was followed. This process consists in exhaustive cross-validation testing that yields the best (C, γ) pair. As the query vector representations introduced in this paper are variants of the well known $\mathtt{Tf} - \mathtt{Idf}$ model, we will compare our performance results considering the $\mathtt{Tf} - \mathtt{Idf}$ query vector representation as a baseline. Table 1 shows the values found for each proposed query representation and for the baseline.

Method	c	γ
(0) `tf-idf`	2048	3.0517578e-05
(1) `tf-pop`	8192	3.0517578e-05
(2) `tf-idf-time`	2048	3.0517578e-05
(3) `tf-idf-pop-time`	2048	3.0517578e-05

Table 1. SVM tuning results for the Broder's taxonomy

4.3 Performance evaluation

In a first analysis and in order to evaluate the overall performance of the classifiers, we compare the nominal and predicted categories for each query, tabulating error rates per category (the proportion of errors over the whole set of instances). These results are shown in Table 2.

Method	Inf.	Nav.	Tran.
`tf-idf`	19.8%	19.28%	**2.9%**
`tf-pop`	25.8%	12.8%	55.1%
`tf-idf-time`	**8.9%**	1.6%	59.5%
`tf-idf-pop-time`	24%	**0.5%**	28.6%

Table 2. Overall performance for the classifiers (error rates). Bold fonts indicate the best error rate for each category.

As we can see in Table 2, for the navigational category the best performance is reached by the $\mathtt{tf} - \mathtt{idf} - \mathtt{pop} - \mathtt{time}$ method. It seems that for the identification of transactional queries the text is the most useful information source, being the $\mathtt{tf} - \mathtt{idf}$ method the one which reaches the best results. We can observe also in Table 2 that the vector query representation defined from Equation 2 achieves the best result for the informational category. In general, the worst result is obtained in the transactional category of the Broder's taxonomy and the best result is obtained in the navigational category.

We expect that the third method outperforms the other methods for all the categories considered in the experiments because it considers all the factors (popularity, reading time and text) but this is not true for the informational category. We intend to illustrate with one example why the second method outperforms the third method in the informational category. The query "buyer of ingersoll rand compressed air dryer" was manually classified in the informational category. The predicted categories for the methods 2 and 3 were informational and transactional. The error of the third method suggests that the use of the $\mathtt{Pop}_{d,q}$ variable in the vector query representation from Equation 3 introduces noise in the weight of the term "buyer", which matchs with instances classified in the transactional category. When we see the whole evaluation data set, we can observe that this kind of error is very frequent for the third method. The confusion between transactional and informational categories represents an error of 19.5% obtained by this classifier in the informational category (over the 90% of the error rate obtained by the third method in this category). On the other hand, this kind of error achieves only the 3% for the second method.

Following the analysis and in order to count the classification costs we consider the four possible cases presented when comparing the nominal class with the predicted class: true positives (*tp*), false positives (*fp*), false negatives (*fn*), and true negatives (*tn*). We calculate the following measures for each classifier of the evaluation set: Precision ($\frac{tp}{tp+fp}$), FP rate ($\frac{fp}{fp+tn}$), TP rate or Recall ($\frac{tp}{tp+fn}$), and F-measure ($\frac{\mathbf{2}\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$). In the case of the F-measure we have specifically considered the version F_1 equivalent to the harmonic average of the Precision and Recall measures given that it establishes a compromise between both criteria (it only has high values if both measures are high).

Given that this decision problem is multiclass, we will use the generalized version of the analysis based on two classes. Following Fawcett [7], we will consider each category as a reference class and will conduct the evaluation comparing it to the other classes. Let C be the set of all the classes considered in the decision problem and let c_i be the reference class upon which we will conduct the evaluation. Let P_i and N_i be the positive and negative classes of the performance analysis based on two classes. Thus for every $c_i \in C$ we will consider:

$$P_i = c_i,$$

$$N_i = \bigcup_{j \neq i} c_j \in C.$$

Then, for each reference class $c_i \in C$, we calculate the measures based on the analysis of both classes using the new classes P_i and N_i. This way, each classifier can be analyzed as a binary classifier according to its performance for the reference class. The results of this analysis can be revised in Table 3.

Table 3 shows the results obtained for the pairs of reference / other classes comparison, considering all the categories of the analyzed taxonomy. Using this

Method	Measures			
	TP Rate	FP Rate	Precision	F-Measure
Informational - Other				
(0) `tf-idf`	0.6538	**0.0292**	**0.9623**	0.7786
(1) `tf-pop`	0.58576	0.13281	0.84186	0.69084
(2) `tf-idf-time`	**0.92614**	0.05141	0.89071	**0.90808**
(3) `tf-idf-pop-time`	0.65000	0.23711	0.31138	0.42105
Navigational - Other				
(0) `tf-idf`	0.9655	0.2597	0.6109	0.7483
(1) `tf-pop`	0.92131	0.13475	**0.88088**	0.90064
(2) `tf-idf-time`	**0.99485**	0.06870	0.87727	**0.93237**
(3) `tf-idf-pop-time`	0.45455	**0.01603**	0.83333	0.58824
Transactional - Other				
(0) `tf-idf`	0.91	**0.0165**	0.9192	0.9146
(1) `tf-pop`	0.75692	0.05344	**0.94615**	0.84103
(2) `tf-idf-time`	**0.98438**	0.05316	0.90000	**0.94030**
(3) `tf-idf-pop-time`	0.70000	0.13153	0.41880	0.52406

Table 3. Performance evaluation of the proposed classifiers. Bold fonts indicate the best result for each evaluation.

we obtain the Informational / Other classes, Navigational / Other Classes, Transactional / Other classes for the Broder's taxonomy. The measures have been calculated for each classifier, with the baseline being identified as $\mathtt{tf} - \mathtt{idf}$, with the one from Equation 1 being identified as $\mathtt{tf} - \mathtt{pop}$, the one from Equation 2 as $\mathtt{tf} - \mathtt{idf} - \mathtt{time}$ and that from Equation 3 as $\mathtt{tf} - \mathtt{idf} - \mathtt{pop} - \mathtt{time}$. As we can see in Table 3, the baseline reaches very good rates for false positives for the informational and transactional categories. On the other hand, the baseline reaches the worst FP-rate for the navigational category. Regarding the predictive capacity of the baseline, it reaches low TP Rates for the informational category being more competitive in the case of navigational queries. The classifier with the best performance in terms of TP Rate was $\mathtt{tf} - \mathtt{idf} - \mathtt{time}$, which is the one that obtained the best proportion of positives over the total. The previous measure indicates that the classifier based on the $\mathtt{tf} - \mathtt{idf} - \mathtt{time}$ representation has the most predictive capacity. It should be noted that the classifier $\mathtt{tf} - \mathtt{pop}$ obtains a precision greater than $\mathtt{tf} - \mathtt{idf} - \mathtt{time}$ for the Navigational and Transactional reference classes, but in the global analysis poorer performance is registered because the values for TP Rate are considerably less that those reached by $\mathtt{tf} - \mathtt{idf} - \mathtt{time}$ in these categories. Evaluating the Precision / Recall tradeoff, and considering the measurement F_1, the classifier based on the $\mathtt{tf} - \mathtt{idf} - \mathtt{time}$ representation also obtained the best performance.

5 Conclusion

In this paper we have explored the use of query classifiers according to the intent of a user query. For this, we have proposed three vector representations for queries based on click-through data and descriptive text, identifying four relevant factors: frequency of terms, (Tf), inverse frequency in documents (Idf), user preferences (Pop), and reading time of selected documents (Time). Using SVMs we have evaluated the performance of the three representations over a set of queries categorized by experts.

The experimental results show that the third method reaches good results when we consider overall performance measures such as error rates. When we consider the cost of making wrong decisions incorporating to the analysis false positives and false negatives, the second method outperforms the other proposed methods. The most relevant reason that explains this performance is its ability to identify informational and navigational queries, which represent a significant proportion of the whole data set. Finally, when we considerate costs into the analysis, the poor result obtained for the transactional category considering the plain error rate was minimized.

One of the most relevant characteristics of the second method is its high predictive / discriminative capacity. Furthermore, the quality of their results depends on the intent that their are identifying. Our approach has an advantange in this sense because we reach higher precision results for all the categories considered in the experiments. Among the factors that explain the success of the proposal we emphasize the incorporation of new factors drawn from click-through data such as time, which was considered for the first time in this problem.

Our classifiers consider only queries that appears in the query-log. We are working on extensions to deal with new queries submitted by users. In order to do this, we have to address the problem of determining distance functions that achieve good results measuring distances between queries with partial information (e.g. query terms) and our query vector representations. If this is possible, we could represent a new query with a close query registered in the query-log file.

Another problem is to conduct an analysis that will allow us to determine why the combination of factors $\mathtt{tf} - \mathtt{idf} - \mathtt{time}$ outperforms $\mathtt{tf} - \mathtt{idf} - \mathtt{pop} - \mathtt{time}$. This work showed that the Pop factor allows for greater precision of the classifiers for the Navigational and Transactional classes and that the factor Time achieves the same but for the Informational class. This suggests that the user preferences are a relevant source of information for navigational and transactional queries, and that the time factor is more relevant for the informational class. This is intuitive in the sense that the reading time allows us to identify resources that capture attention based on content, usually pages, as opposed to clicks which would tend to concentrate more on sites than pages, making them more related to transactional and navigational queries.

Acknowledgments

Mr. Zamora was supported by a fellowship for scientific initiation of the Graduate School of the UTFSM, Chile. We would like to thank the contributions of Ricardo Baeza-Yates, Hector Allende, Hector Allende-Cid, Libertad Tansini and Katharine Sherwin.

References

1. R. Baeza-Yates, L. Calderón-Benavides, and C. González-Caro. The intention behind web queries. In *SPIRE '06: Proceedings of the 13th International Conference on String Processing and Information Retrieval*, pages 98–109, Oct 11th - 13th, 2006, Glasgow, Scotland, Springer LNCS 4209.
2. A. Basu, C. R. Watters, and M. A. Shepherd. Support vector machines for text categorization. In *HICSS '03: Proceedings of the 36th Hawaii International Conference on System Sciences*, page 7, Jan 6th - 9th, 2003, Hawaii, USA, IEEE Computer Press.
3. A. Broder. A taxonomy of web search. *SIGIR Forum*, 36(2):3–10, 2002.
4. C. C. Chang and C. J. Lin. LIBSVM: a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm
5. M. Claypool, D. Brown, P. Le, and M. Waseda. Inferring user interest. *IEEE Internet Computing*, 5(6):32–39, 2001.
6. C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.
7. T. Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.
8. S. Fox, K. Karnawat, M. Mydland, S. Dumais and T. White Evaluating implicit measures to improve web search. *ACM Transactions on Information Systems*, 23(2):147–168, 2005.
9. C. W. Hsu, C. C. Chang and C. J Lin. A practical guide to support vector classification. Department of Computer Science and Information Engineering, National Taiwan University, 2003.
10. B. Jansen, D. Booth, A. Spink. Determining the informational, navigational and transactional intent of Web queries. *Information Processing and Management*, 44(3):1251–1266, 2008.
11. I.-H. Kang and G. Kim. Query type classification for web document retrieval. In *SIGIR '03: Proceedings of the 26th International ACM SIGIR Conference*, pages 64–71, Jul 28th - Aug 1st, 2003, Toronto, Canada, ACM.
12. U. Lee, Z. Liu, and J. Cho. Automatic identification of user goals in web search. In *WWW '05: Proceedings of the 14th International Conference on World Wide Web*, pages 391–400, May 10th - 14th, 2005, Chiba, Japan, ACM.
13. E. Leopold and J. Kindermann. Text categorization with support vector machines. how to represent texts in input space? *Machine Learning*, 46(1-3):423–444, 2002.
14. Y. Liu, M. Zhang, L. Ru, and S. Ma. Automatic query type identification based on click through information. In *AIRS '06: Proceedings of the Third Asia Information Retrieval Symposium*, pages 593–600, Oct 16th - 18th, 2006, Singapore, Springer LNCS 4182.
15. D. E. Rose and D. Levinson. Understanding user goals in web search. In *WWW '04: Proceedings of the 13th International Conference on World Wide Web*, pages 13–19, May 17th - 20th, 2004, New York, NY, USA, ACM.
16. G. Salton and C. Buckley. Term-weighting approaches in automatic retrieval. *Information Processing and Management*, 24(5):513–523, 1988.
17. J. Wen, J. Nie and H. Zhang Clustering user queries of a search engine. In *WWW '01: Proc. of the 10th Int. Conf. on World Wide Web*, pages 162–168, May 1st - 5th, 2001, Hong Kong, ACM.