Predicting Query Times

Rodger McNab, Yong Wang, Ian H. Witten, Carl Gutwin
Department of Computer Science
University of Waikato
New Zealand
{rjmcnab, yongwang, ihw, gutwin}@cs.waikato.ac.nz

Abstract We outline the need for search engines to provide user feedback on the expected time for a query, describe a scheme for learning a model of query time by observing sample queries, and discuss the results obtained for a set of actual user queries on a document collection using the MG search engine.

1 Introduction

Hundreds of millions of queries are made daily to search engines and digital libraries around the world. Because processing and network times are highly variable, awaiting the results incurs considerable frustration. Human factors research suggests that when response times exceed one second, a system should provide feedback about its activity [1]. Regrettably, search engines provide no information about the progress of a query—leaving users in the dark, wondering whether things are working properly. Even a rough idea of expected time can help one decide whether to wait, abandon the query, or resubmit it in modified form.

This paper describes a methodology for predicting response times of full-text retrieval systems. This is used to predict the duration of new queries and to provide information about the search (for instance, why it is taking so long), improving usability. The scheme is tested by creating a model for the MG search engine based on observed queries to the *Computer Science Technical Reports* collection of the New Zealand Digital Library (http://www.nzdl.org/cstr). Until network delays can be modeled, the results are principally applicable to corporate intranets where network response time is reasonably predictable.

2 Modeling query times

Our goal is to form a model of response time so that it can be predicted. While in principle achievable by analyzing the search engine's internal operation, this would be a tedious exercise that must be repeated whenever the implementation or hardware parameters change. Instead, we construct a model of response time from observed performance data. We use machine learning techniques to build the model from about twenty attributes that affect the processing time for a query (listed in Table 1). Some apply in general to many retrieval systems, while others are peculiar to the search engine used here (MG [2]). Our methodology applies to any search engine, whatever its characteristic parameters.

3 Methodology

The model was based on the largest and most widely-used collection within the New Zealand Digital Library, comprising 40,000 technical reports that total 2.3 Gb of plain text. To generate training and test data, 20,000 actual user queries logged for this collection over one year were resubmitted to the system while it was in normal, routine operation; and the values of the attributes were recorded. Most queries took only a brief time to process, 2500 took between one and three seconds, and about 1500 took longer—up to 371 seconds. The resulting data was split into a training set of 12,000 queries and a test set of 8,000 queries.

The machine learning algorithm M5' [3] was used to build a predictive model from the training data. This induces a decision tree, splitting the data at each branch based on the values of a carefully-chosen attribute. However, instead of storing a particular "class" value at each leaf as a regular decision tree does, a regression model is calculated for those training examples corresponding to the leaf. A fragment of the resulting tree appears in Figure 1 (the linear models at the leaves are left unspecified).

Collection Attributes
Size of collection
Average size of document
Query Attributes
Index level (document or paragraph)
Query type (boolean or ranked)
Maximum number of documents to return
Whether stemming is specified
Whether case-folding is specified
Number of terms in the query
Frequency of most frequent query terms
Whether the documents need to be post-processed
General Attributes
Machine load

Table 1. Attributes used in the prediction model

Category (sec)	Instances	Mean Absolute Error	90% Confidence Interval (sec)
$t < 1$	8691	0.14	{–0.36, 0.29}
$1 \le t < 3$	1351	0.55	{–1.2, 0.92}
$3 \le t < 8$	786	1.8	{–5.6, 2.9}
$t \ge 8$	44	8.0	{–9.8, 7.0}

Table 2. Performance of the prediction model

We evaluated the model on the queries in the test set. For each one, M5' predicted the response time, and the error was calculated as the difference between this and the actual response time. Since the data is heavily skewed towards short queries, we consider error values separately in four different categories, corresponding to those suggested in the human factors literature for issuing response-time feedback [1].

4 Results

The mean error value over all test queries was 0.37 seconds. This is low because of the preponderance of short queries, and error values for the four categories are shown in Table 2. To present predictions to users in a comprehensible form, we also calculated from the data the interval that contains the correct value 90% of the time.

A detailed analysis of the actual errors is instructive. Many major errors (greater than two seconds) relate to queries that involved postprocessing for phrases, for which the time taken depends greatly on the number of documents returned—because MG decompresses and scans each one. Greater accuracy could likely be obtained by predicting the number of documents returned for such queries (a task to which the same methodology might apply). Several other errors involved situations where the processor loading was light initially but increased dramatically during the query; again, a separate prediction of loading might be useful.

5 Conclusion

We aim to present response-time feedback to search engine users. Although the confidence intervals in each error category are quite wide, the predictions can nevertheless provide a valuable indication of how long a query will take. Moreover, the decision tree allows us to determine what it is about a query that contributes most to processing time.

As digital libraries grow in size, complexity and popularity, the need for feedback will become ever more pressing. We have shown how to predict query time by automatically modeling the performance of a search engine. While there are prospects for improving prediction accuracy, this technique can already provide information that alleviates the daily frustration of search engine users.

References

[1] Shneiderman, B. Designing the User Interface: Strategies for Effective Human-Computer Interaction. Addison-Wesley, 1992.

[2] Wang, Y. and Witten, I.H. "Induction of model trees for predicting continuous classes" *Proc European Conference on Machine Learning Poster Papers*, Prague, Czech Republic, pp. 128–137; April 1997.

[3] Witten, I.H., Moffat, A. and Bell, T.C. Managing Gigabytes. Van Nostrand Reinhold, 1994.

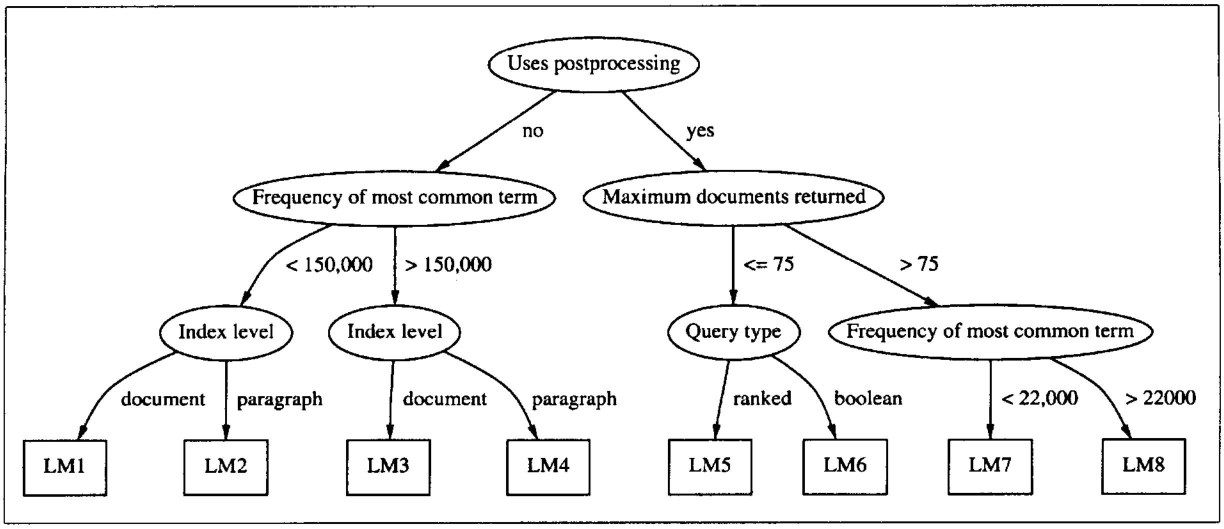

Figure 1. Uppermost levels of the prediction model