Analysis of Long Queries in a Large Scale Search Log

Michael Bendersky
Center for Intelligent Information Retrieval
Department of Computer Science
University of Massachusetts
Amherst, MA 01003
bemike@cs.umass.edu

W. Bruce Croft
Center for Intelligent Information Retrieval
Department of Computer Science
University of Massachusetts
Amherst, MA 01003
croft@cs.umass.edu

ABSTRACT

We propose to use the search log to study long queries, in order to understand the types of information needs that are behind them, and to design techniques to improve search effectiveness when they are used. Long queries arise in many different applications, such as CQA (community-based question answering) and literature search, and they have been studied to some extent using TREC data. They are also, however, quite common in web search, as can be seen by looking at the distribution of query lengths in a large scale search log.

In this paper we analyze the long queries in the search log with the aim of identifying the characteristics of the most commonly occurring types of queries, and the issues involved with using them effectively in a search engine. In addition, we propose a simple yet effective method for evaluating the performance of the queries in the search log using a combination of the click data in the search log with the existing TREC corpora.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Query Formulation

General Terms

Human Factors, Experimentation, Algorithms

Keywords

Long queries, click data, web search

1. INTRODUCTION

Analysis of large scale commercial web search logs has recently become an active topic of research. User activity recorded in these logs has proved to be a valuable resource for the researchers in the fields of information retrieval, data mining, machine learning and natural language processing. Large volumes of user queries and the corresponding click data in the search logs were successfully leveraged for a supervised learning of retrieval functions [14, 1], spelling corrections [2, 8], abbreviation disambiguations [35], query segmentations [5] and term associations [34], among other tasks.

In addition to being a rich source of data for improving supervised learning algorithms, search logs also provide an insight into the searcher behavior. Jones and Klinkner [16] proposed a hierarchical model of user search activity consisting of search missions and corresponding goals. Downey et al. [9] investigated how user behavior is influenced by the frequency of the underlying information needs. Pass et al. [28] examined issues such as the distribution of the popular search topics and users demographics. Mei and Church [24] proposed click entropy as a measure of search difficulty and found hourly and daily fluctuations in search patterns.

In this paper, we extend the research on search logs analysis by examining a relatively small yet significant segment of queries in the search logs: long queries. Although long queries are still a minority among the searches on the web, we believe it is important to study them for several reasons.

First, long natural language queries have been shown to provide a better way for expressing complex and specific information needs than the short keyword queries [20, 29]. Although this quality makes the long queries less suitable for certain types of common navigational web searches (such as those described by Broder [7]), it makes them crucial for informational search activities such as search in Q&A archives [36], search-in-context [11], enterprise or academic search [13], or searches in environments where keyword queries might be hard to articulate, e.g., voice-activated search.

Second, existing retrieval methods, in general, perform worse for the long queries than for the short ones; a result which has been consistently and independently affirmed both in the experiments on TREC corpora [4, 18] and in the web search setting [9]. This result has been attributed to the rarity of the long queries [9], a lack of sufficient natural language parsing by the underlying retrieval method [19], term redundancy [18] and a difficulty in distinguishing between the key and the complementary concepts [4].

To the best of our knowledge, this paper is the first attempt to study the characteristics of the long queries in a large scale search log of a commercial search engine. The main goals of this study are to

- gain an insight on how people formulate the long queries on the web, and how they behave in response to their results,

WSCD '09, Feb 9, 2009 Barcelona, Spain.

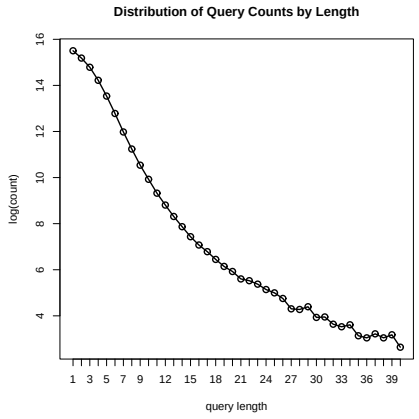


Figure 1: Distribution of query lengths on log **scale (truncated at $l(q) = 40$).**

- discuss potential methods for improving retrieval effectiveness for the long queries,
- provide a common basis for retrieval performance evaluation using both implicit relevance feedback from the search log and the TREC corpora.

For our analysis we use the MSN Search query log excerpt, including ∼15M queries and the associated clicks, sampled over a period of one month. The remainder of the paper is organized as follows. In Section 2 we discuss the structural characteristics of the queries in the search log. In Section 3 we examine the correlation of user clicks and these characteristics. Section 4 is dedicated to the discussion of the previous work on improving the retrieval effectiveness of the long queries. Section 5 describes our method of combining click data and TREC corpora for evaluation purposes. Section 6 details the conclusions and the potential areas of future work.

2. QUERY ANALYSIS

In this section we look into the structural characteristics of the queries in the search log. For each query instance in the search log[1], q, we define the following attributes.

Query length, $l(q)$, is the length of the query string associated with q in terms of number of *word tokens* (sequences of characters separated by space). We discuss the distribution of query lengths in Section 2.1.

Query type, $t(q)$, is the type of the query string associated with q. We define the type of the query based on the structural characteristics of the query string, which are detailed in Section 2.2.

2.1 Length Distribution

Figure 1 shows the distribution of queries by length. Query lengths demonstrate a power-law distribution, with the long queries in the tail.

Most queries in the search log are short. Queries with $l(q) \leq 4$ account for 90.3% of the total queries. For 99.9% of the queries, $l(q) \leq 12$. We concentrate on these queries for the rest of our analysis and experiments, and disregard ∼14K queries (0.01% of the queries) for which $l(q) > 12$. The reason for this is two-fold. First, the number of queries in this region, binned by length, is small, and the click data is sparse. Second, we found a considerable number of seemingly bot-generated queries among these queries, which, if included, might have skewed the results of our consequent analysis and experiments.

The *expected length* of a query is 2.4, which is in line with the previous studies [28, 32]. We calculate the expected query length as $\frac{\sum_l (n_l l)}{N}$, where l is the query length, n_l is the number of queries of length l, and N is the total number of queries.

Total Queries: 14,921,286		
Long Queries ($5 \leq l(q) \leq 12$) **:** 1,423,664		
Type	Count	% of Long
Questions (QE)	106,587	7.49
Operators (QE)	78,331	5.50
Composite (CO)	910,103	63.93
Noun Phrases (NC_NO)	209,906	14.74
Verb Phrases (NC_VE)	118,736	8.34

Table 1: Summary of query types.

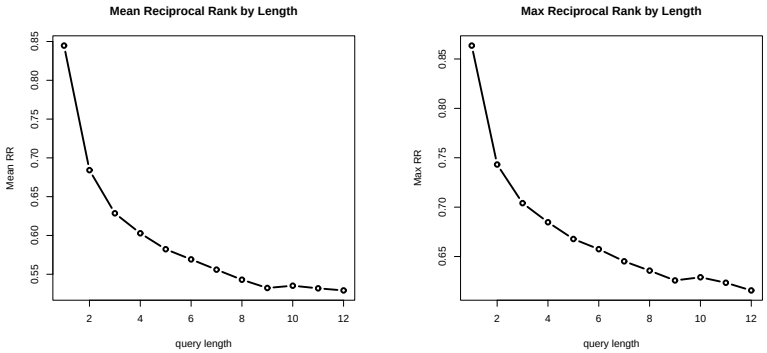
2.2 Query Types

Due to our focus on the long queries in this paper, we divide the queries in the log into two (unequally sized) main types: *short* and *long*. For simplicity, the division is based on query length (as defined above), such that *short* queries are queries for which $l(q) \leq 4$ and *long* queries are queries for which $5 \leq l(q) \leq 12$. All the short queries are assigned to a type SH, while the long queries are divided between five mutually exclusive types, which are summarized in Table 1. In what follows, we present a detailed description of each query type, along with some examples[2].

- *Questions* (QE). Questions are defined as queries that begin with the one of the words from the set: {*what, who, where, when, why, how, which, whom, whose, whether, did, do, does, am, are, is, will, have, has*}.
  - *Examples*
    - `What is the source of ozone?`
    - `how to feed meat chickens to prevent leg problems`
    - `do grover cleveland have kids`
- *Operators* (OP). Operators are defined as queries that contain (at least) one of the boolean operators {*AND, OR, NOT*}, one of the phrase operators {*+, "*}, or one of the special web-search operators defined by MSN search {*contains:, filetype:, inanchor:, inbody:, intitle:, ip:, language:, loc:, location:, prefer:, site:, feed:, hasfeed:, url:*} .
  - *Examples*
    - `bristol, pa AND senior center`
    - `"buffalo china " pine cone"`
    - `site:dev.pipestone.com ((Good For A Laugh))`
- *Composite* (CO). Queries that can be represented as a composition of queries of type SH. A dynamic programming algorithm (similar to the one described in [33]) is

[1]Note that query instance is determined by a single issue of the query, denoted by a unique ID in the search log. Thus, different query instances can contain identical query strings.

[2]Spelling and punctuation of the original queries is preserved.



(a) $meanRR(q)$ (b) $maxRR(q)$

Figure 2: Mean (a) and maximum (b) reciprocal ranks of clicks, averaged by length.

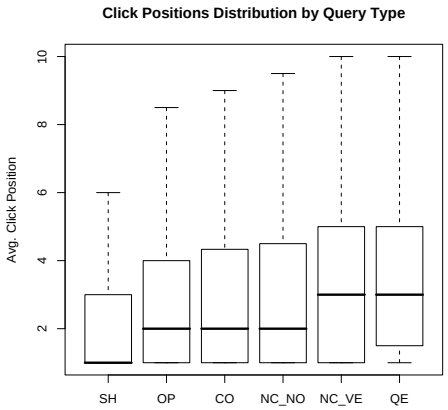


Figure 3: Boxplot of the distribution of the average click positions per query for different query types.

used to find all the possible non-trivial query segmentations, such that each segment in the segmentation corresponds to a short query from the log. Non-trivial segmentation is a segmentation that includes at least one segment of length greater than one. Queries for which no such segmentations are found, are marked as non-composite. Queries of type `QE` and `OP` are excluded.

  - *Examples*
    - `persian rug dealers in austin texas`
    - `T.I. the rapper web site`
    - `merryhill schools a noble learning community`

- *Non-Composite.* Queries that cannot be represented as a composition of the queries of the type `SH` are divided into two types: *noun phrases* (`NC_NO`) and *verb phrases* (`NC_VE`). These types are distinguished by a presence of a verb in the query (a verb is defined as a word token tagged[3] by a tag from Penn Treebank Tagset [23], satisfying the regular expression `VB.*`).

  - *Examples* — `NC_NO`
    - `TEMPLE OF THE FULL AUTUMN MOON`
    - `Hp pavilion 503n sound drive`
    - `lessons about children in the bible`
  - *Examples* — `NC_VE`
    - `detect a leak in the pool`
    - `teller caught embezzling after bank audit`
    - `eye hard to open upon waking in the morinig`

3. CLICK DATA ANALYSIS

In this section we perform analysis, based on the click data in the search logs, aimed at better understanding the user behavior and the retrieval performance of the search engine for the long queries. To this end, we examine the clicks in the search log associated with q. For the purpose of our work, we associate several click-based measures with each query instance. $meanRR(q)$ and $maxRR(q)$ are the mean and maximum reciprocal ranks of the clicks for query instance q, respectively [31]; $meanPos(q)$ is the mean click position for q. All the measures are calculated over the queries with at least one click.

We examine the relation of these measures to query length (Section 3.1), query type (Section 3.2) and query frequency (Section 3.3).

3.1 Clicks and Query Length

Previous research on TREC collections [4, 18] showed that using the existing retrieval techniques, the effectiveness of the verbose description queries is lower, in general, than the effectiveness of their shorter keyword counterparts. Figure 2 presents a similar picture for the queries in the search log.

Figure 2 shows the mean and maximum reciprocal rank of the clicks ($meanRR(q)$ and $maxRR(q)$, respectively), averaged by query length. Note, that if we assume direct relation between the reciprocal rank of the clicks and the effectiveness of the retrieval, the effectiveness decreases as the query length increases. For comparison, there is a 29% decrease in the expected reciprocal rank of the first click between the shortest ($l(q) = 1$) and the longest ($l(q) = 12$) queries in our data set.

3.2 Clicks and Query Type

The types of queries discussed in Section 2.2 are derived from the structure of the query strings. However, although the proposed taxonomy is reasonable from a syntactical point of view, we are more interested in its utility for the improvement of the retrieval with long queries. Accordingly, in this section we explore whether the users interaction with the search engine differs for each of the query types.

Since the number of queries differs significantly across the query types, we utilize sampling for fair comparisons of aggregate statistics by type. We collect the complete click information for a random sample of 10,000 queries from each of the six query types (the short queries and the five types of the long queries).

Figure 3 shows the distribution of $meanPos(q)$ measure for the six types of queries in the random sample. Note that a larger value in the boxplot translates into a *lower* position of the clicks in the ranked list. For example, for the short queries (type `SH`) the median of $meanPos(q)$ is the

[3]MontyLingua (*web.media.mit.edu/∼hugo/montylingua*) is used as a POS tagger.

first result in the ranked list, while for the question queries (type QE), the median is the third result.

Figure 3 demonstrates that (a) users tend to click lower in the result list for the long queries than for the short ones, and (b) there are differences in user click behavior between the different types of long queries.

Table 2 details the means of the query length, the mean and maximum reciprocal rank of the clicks ($meanRR(q)$ and $maxRR(q)$, respectively), and the abandonment rate ($abRate(q)$), which is the fraction of queries with no clicks, for each of the query types in the random sample. The latter three measures are calculated as in [31].

Table 2 shows that there are statistically significant differences between the short and the long queries, as well as between the different types of long queries. Taking the assumption that a higher mean and maximum reciprocal rank of the clicks indicates better retrieval performance of the search engine, we can state that the search with the short queries is generally more effective than the search with the long queries, which is in line with the click data presented in Figure 2. Specifically for the long queries, operators, composite queries and noun phrases are more effective than verb phrases and questions.

Although, as Figure 2 shows, longer queries, in general, perform worse than the short ones, the difference between the performance of the various types of long queries cannot be attributed solely to the query length. For instance, operators (OP) are longer than the noun phrases (CO, NC_NO), but their performance in terms of $meanRR(q)$ and $maxRR(q)$ is the same. On the other hand, the performance of the questions (QE) and verb phrases (NC_VE) (in terms of $maxRR(q)$) is worse by, respectively, 6.7% and 4.5% from the average performance of the queries of the same length.

Abandonment rate trends are generally similar to the reciprocal rank measures. Abandonment rate is lower for the short (SH) and composite (CO) queries than for the other query types, which indicates a higher level of user satisfaction with the retrieval results in response to these queries. There are a few interesting contradictions between the reciprocal rank and the abandonment rate measures, however. In general, it is reasonable to assume that the reciprocal rank is inversely related to the abandonment rate, since both low reciprocal rank and high abandonment rate correlate with user dissatisfaction [31].

Surprisingly, this does not hold for operator queries (OP) — although the reciprocal rank of clicks is higher than for the other long query types, the abandonment rate is high. This can be attributed to the Boolean nature of these queries: either the user finds the correct result, or the query is abandoned entirely.

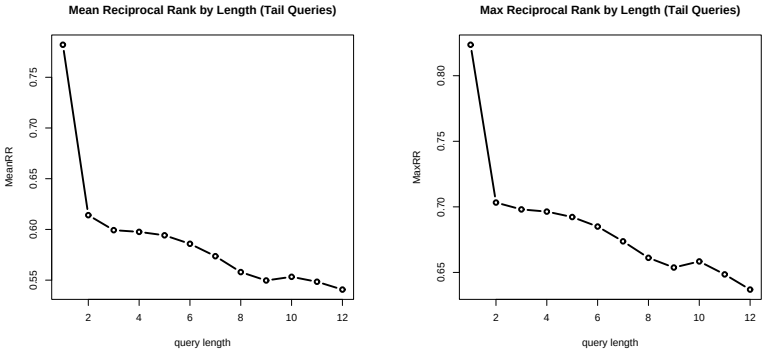
On the other hand for questions (QE), the abandonment rate is relatively low, indicating that users who ask questions, in almost half of the cases, do tend to find at least one plausible answer. This is in contrast to non-composite queries (NC_NO, NC_VE), where users abandon the search in $\sim 60\%$ of the cases.

3.3 Clicks and Query Frequency

Downey et al. [9] investigated the influence of query frequency on user behavior, and found that the rarer queries result in less clicks and page visits and more query reformulations than the more frequent ones. They concluded that users tend to be more satisfied with the results of the more

Type	$l(q)$	$meanRR(q)$	$maxRR(q)$	$abRate(q)$
SH	1.99	0.73	0.77	0.40
CO	5.67	0.59	0.67	0.41
OP	6.05	0.58	0.67	0.58
NC_NO	5.77	0.58	0.67	0.61
NC_VE	6.35	0.53	0.62	0.59
QE	6.75	0.51	0.61	0.51

Table 2: Click information analysis for the different query types. Query types are grouped (groups marked by double horizontal lines), so that there are no statistically significant differences in $meanPos(q)$ within the group according to two-tailed t-test ($\alpha = 0.05$).



(a) $meanRR(q)$ (b) $maxRR(q)$

Figure 4: Mean (a) and maximum (b) reciprocal ranks of clicks for *tail queries*, averaged by length.

popular queries. This result is relevant to our work, since as we have shown in Figure 1, the long queries are in the tail of the query frequency distribution.

Downey et al. also investigated the influence of length on query performance, summarizing their findings by stating that *"... query frequency is more important than query length, indicating perhaps that Web search engines are optimized to handle common requests"*. We note however, that this finding was based on examining only queries for which $1 \leq l(q) \leq 5$, that is the range of query lengths most of which we define as the short queries in this paper.

Figure 4 plots the mean and the maximum reciprocal rank of the clicks for the *tail queries*. For simplicity, we define tail queries as query strings that appear only once in the search log. Figure 4 demonstrates that, in addition to the query frequency, the query length affects the retrieval performance. When compared to the general query population (Figure 2), both the mean and the maximum reciprocal rank decrease for the short tail queries, while increasing for the long ones. For comparison, for queries with $l(q) = 1$, there is a 5% *decrease* in $maxRR(q)$, while for queries with $l(q) = 10$, there is a 5% *increase* in $maxRR(q)$.

In general, however, the retrieval performance, expressed by $meanRR(q)$ and $maxRR(q)$, still decreases as a function of $l(q)$, even when considering only the tail queries. For comparison, there is a 20% decrease in the expected reciprocal rank of the first click between the shortest ($l(q) = 1$) and the longest ($l(q) = 12$) tail queries in our data set. This is less than the 29% decrease reported for all the queries (see Section 3.1), but still significant.

4. PREVIOUS WORK

Detailed implementation and evaluation of existing and novel techniques for improving the retrieval performance of the long queries is out of the scope of this paper. However, this paper would be incomplete without at least a brief discussion of the existing research. In what follows, we survey several promising techniques that were previously shown to improve retrieval effectiveness of the long queries.

Query Reduction. This technique aims to improve the performance of the long queries by eliminating the redundancy. For instance, Kumaran and Allan [18] have proposed an interactive method that helps the user to reduce a natural language query to a few keyword terms, thereby improving the retrieval effectiveness of the resulting queries.

Query Expansion. This is a well known technique that can be applied to both short and long queries. However, the quality of the initial retrieval is vital for the success of query expansion. Long queries may often produce unsatisfactory initial results, which will in turn yield unhelpful terms for query expansion. Kumaran and Allan [19] have shown that an interactive technique that helps the user to decide between query expansion and query reduction works better that either by itself.

Query Reformulation. This is a broad definition that covers, among others, query suggestions [22, 21, 26], term substitution by synonyms or contextual terms [34], resolution of abbreviations [35], spelling correction [8] and "translation" of the query terms using some form of term association [36]. Marchionini and White [22] suggest that query formulation requires a *semantic mapping* of the user's vocabulary onto the system's vocabulary. Query reformulation can potentially improve the quality of this mapping. Thus, although pertinent for all query types, query reformulations are especially important for the long queries, where there is more opportunity for incorrect semantic mappings to occur.

Term and Concept Weighting. Mei et al. [25] proposed a Poisson query generation model for information retrieval that allows for term-specific smoothing based on collection statistics. Bendersky and Croft [4] proposed a supervised method for weighting concepts (determined by noun phrase extraction) in verbose natural language queries. Both methods were shown to be more effective than a standard query-likelihood model [30] for the long queries.

Query Segmentation. This technique aims to segment the query into one or more atomic concepts, which, if done correctly, allows a more precise application of the above techniques on the level of concepts, rather than terms. A simple segmentation method based on mutual information was first proposed by Jones et al. [17]. More recent work proposed more accurate supervised segmentation methods [5, 12] and a competitive unsupervised method based on a large web collection and Wikipedia titles [33]. Guo et al. [12] have also shown that query segmentation has a positive impact on the retrieval performance.

Although several of the above methods were shown to improve retrieval effectiveness on a variety of test collections [4, 12, 25, 34], they were never compared or combined using a single test-bed. One of the purposes of this paper is to propose a unified framework for evaluating the existing and novel methods using a combination of search log data and a standard TREC collection.

5. RETRIEVAL EVALUATION

In this section we examine the potential utility of combining the search log data and an existing standard TREC collection for evaluating the effectiveness of some of the existing retrieval methods for both short and long queries.

Some previous work [4, 18, 19] evaluated the performance of the long queries using TREC topics. However, as witnessed in the search log, the structure of the verbose queries provided with these topics (the "description" and the "narrative" topic parts) differ significantly from the structure of the long queries being issued by the search engine users.

In order to use the queries from the search log for evaluation, we would like to leverage the user activity recorded in the search log such as click data, dwell times, reformulation activity, etc., as implicit relevance judgments [15]. A popular approach to inferring relevance from user feedback is using clicks as pairwise relevance judgments [14, 15, 31, 10]. For this approach to be accurate, however, a large number of clicks is required [14, 10]. In addition, in some cases, such an evaluation requires a controlled experimental environment [15, 31].

As Figure 1 demonstrates, long queries are in the tail of queries distribution, hence they produce less clicks to be exploited. In addition, more often than not, the researchers do not have access to the entire corpus on which the clicks were performed. Hence the retrieval method can be applied to (and evaluated using) only a small subset of the clicks that occur within an existing TREC web corpus. As we show, this dramatically decreases the amount of information that can be used as implicit relevance feedback.

Due to these constraints, we propose using the clicks as absolute relevance judgments, similarly to some earlier work [6]. Although the sparse click data does not allow evaluating the different retrieval systems directly, we show that it is sufficient for determining the relative performance of the retrieval methods.

5.1 Search Log and the TREC Terabyte Track

In what follows, we propose a simple procedure for generating a set of implicit relevance judgments from click data for a retrieval performance assessment on an existing TREC collection. For the purpose of our experiments we use the GOV2 test collection and the accompanying topics and relevance judgments, which are a part of TREC Terabyte Track[4]. GOV2 is currently the largest publicly available TREC web collection, and it contains $\sim 25M$ documents from .gov domain.

To select queries to be evaluated, we first obtain a complete list of URL's in GOV2 collection, denoted U_{GOV2}. We then pick a set of query strings such that each query string in the set

- occurs more than once in the search log
- is associated with at least one click on a URL in U_{GOV2}.

The resulting evaluation set contains a total of $13,890$ queries. Long queries constitute 8.5% of the set. Due to the sparseness of the available click data, we ignore the *position bias* and the *context bias* [15] of the clicks, and treat each click as an absolute relevance judgment.

[4]See *http://www-nlpir.nist.gov/projects/terabyte/* for more details on the TREC Terabyte Track.

5.2 Evaluating the Short Queries

To examine whether the evaluation using the click data obtained from the search log is reliable, we compare it to the evaluation using explicit manual relevance judgments available (as a part of the TREC Terabyte Track) for 150 topics. For evaluating the retrieval performance, we use four retrieval systems, based on either query-likelihood [30] or a Markov random field [27] retrieval models, all implemented using the Indri search engine[5].

The first retrieval system, QL, is a standard query-likelihood model [30] with Dirichlet smoothing parameter set to 1500. The second retrieval system, $QL_{\bar{m}}$, is the same query-likelihood model, but with no smoothing. Previous research shows that smoothing is crucial for optimizing the performance of the language modeling techniques for information retrieval [37], and hence $QL_{\bar{m}}$ is expected to attain significantly inferior results to those attained by QL.

The third retrieval system, SDM, is a sequential dependence variant of a Markov random field model for information retrieval [27], which, in contrast to the bag-of-words approach of the standard query-likelihood, employs term dependencies between all pairs of the adjacent query terms. SDM has consistently outperformed QL on the GOV2 collection [27], and was among the top performing retrieval systems at the Million Query Track 2007 [3], which also used the GOV2 test collection.

All of the above systems employ light stopword removal, removing a list of 25 common stopwords from the queries. The fourth system, $QL_{\bar{s}}$, uses the same retrieval model as QL, but with no stopword removal at runtime.

To evaluate the performance of the four retrieval systems we use as short queries (a) titles of the 150 topics for which explicit relevance judgments on GOV2 are available, and (b) a random sample of 700 queries with at least two clicks on a URL in U_{GOV2}, such that $2 \leq l(q) \leq 4$. Table 3 details the retrieval performance, in terms of precision at 5 and mean average precision.

Note that the relative performance of the four methods, as presented in Table 3, is similar when using either the TREC topics titles or the short queries from the search logs. In both cases (a) the smoothing has a statistically significant positive impact on the retrieval performance, (b) there is no statistically significant difference between retrieval with or without stopwords, and (c) SDM is significantly more effective than all the query-likelihood methods.

This similarity in performance measurement is encouraging, since even though the available click data is sparse (2.2 clicked URL's in U_{GOV2} per query, on average) and a click on the URL does not necessarily represent its relevance, the resulting evaluation set is at least good enough for reliably distinguishing between retrieval systems of a significantly varying quality (as attested by Wilcoxon sign test results at Table 3).

5.3 Evaluating the Long queries

Similarly to the evaluation of the performance of the short queries in the search logs, we proceed to evaluate the performance of the retrieval systems described above with the different types of long queries. As the click data for the long queries is even sparser than for the short ones, we use for evaluation all the long queries that appear more than once in the search logs and have at least one click on a URL in U_{GOV2}. Four out of five types of long queries are represented by more than 60 such queries, and the retrieval results for these queries using both QL and SDM are shown in Table 4. There are only 17 such queries of the operator query (`OP`) type, and the results for this type are not shown.

Table 4 shows that SDM, in general, retrieves more clicked pages than QL for `CO` and `NC_NO` methods, while for the `QE` and `NC_VE` both methods retrieve the same percentage of clicked pages. In 3 out of 4 cases, SDM retrieves more clicked pages in the top 5 ranks. Though not directly comparable, due to the different characteristics of the queries, these results are in line with the previous work, which have shown that SDM attains better performance for the "description" portions of TREC topics on GOV2 collection [4].

Clearly, the number of the available queries and click-based relevance judgments for the long queries is too low in most cases to make a confident decision about the retrieval performance of a certain method. In the future work, we would like to augment this data using either additional implicit feedback data or traditional pooling methods.

	(a) 150 TREC titles		(b) 700 Search Log Queries	
	prec@5	MAP	prec@5	MAP
$QL_{\bar{m}}$	35.44	20.04	3.11	6.03
$QL_{\bar{s}}$	57.32	29.68	3.69	7.09
QL	56.64	29.56	3.77	7.14
SDM	62.01	32.40	4.40	8.01

Table 3: Retrieval results using (a) 150 TREC topic titles with explicit relevance judgments and (b) 700 short queries ($2 \leq l(q) \leq 4$) with clicks in U_{GOV2} as relevance judgments. Double line indicates a statistically significant difference (Wilcoxon sign test, $\alpha = 0.05$) in MAP between the methods below and above the line.

Qry Type	Method	# Qry	#Clk	prec@5	% Clk-Ret
`CO`	QL	920	978	3.04	59.1
	SDM			3.22	63.5
`NC_NO`	QL	97	104	6.60	76.92
	SDM			5.77	80.77
`NC_VE`	QL	67	67	6.87	97.01
	SDM			7.76	97.01
`QE`	QL	88	93	4.09	77.42
	SDM			4.32	77.42

Table 4: Retrieval results for QL and SDM methods for four types of long queries. For each query type and retrieval method the following is shown: number of queries, number of available clicked pages in U_{GOV2}, precision at 5 (using clicked pages as relevance judgments) and % of clicked pages that are retrieved by the method.

6. CONCLUSIONS AND FUTURE WORK

In this paper we used the search log for analyzing the characteristics of the long queries issued by the web search engine users. We examined the most common query types, and showed that user click behavior is correlated with query length, type of the query and query frequency. We also performed some initial retrieval experiments that demonstrate that the click data in the search logs and the existing TREC corpora can be combined to reliably distinguish between retrieval systems of a significantly varying quality.

[5]Available at *http://www.lemurproject.org/indri/*.

This paper opens up several potential directions for both short-term and long-term future work on improving the quality of search with long queries. In the short term, the existing retrieval methods presented in Section 4 can be evaluated on TREC corpora on actual user queries using the evaluation methods that incorporate the click data from the search logs. The click data can be further augmented by introducing manual relevance judgments.

In the longer term, specific retrieval methods can be developed, targeting the most common types of queries identified in the search logs. Arguably, this would yield better results than the existing "one size fits all" approach to retrieval. For instance, a natural language processing approach should be more suitable for the verb phrases and questions, while noun phrase queries might be better served by a syntax agnostic query segmentation.

7. ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval and by NSF grant #IIS-0711348. Microsoft Live Labs provided the query log. Any opinions, findings and conclusions or recommendations expressed in this material are the authors' and do not necessarily reflect those of the sponsor.

8. REFERENCES

[1] E. Agichtein, E. Brill, and S. Dumais. Improving web search ranking by incorporating user behavior information. In *Proceedings of SIGIR*, pages 19–26, 2006.

[2] F. Ahmad and G. Kondrak. Learning a spelling error model from search query logs. In *Proceedings of HLT*, pages 955–962, 2005.

[3] J. Allan, B. Carterette, J. Aslam, V. Pavlu, B. Dachev, and E. Kanoulas. Million query track 2007 overview. In *Proceedings of TREC*, 2008.

[4] M. Bendersky and W. B. Croft. Discovering key concepts in verbose queries. In *Proceedings of SIGIR*, pages 491–498, 2008.

[5] S. Bergsma and Q. Wang. Learning Noun Phrase Query Segmentation. In *Proceedings of EMNLP-CoNLL*, pages 819–826, 2007.

[6] J. Boyan, D. Freitag, and T. Joachims. A machine learning architecture for optimizing web search engines. In *Proceedings of AAAI*, volume 264, 1996.

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

[8] S. Cucerzan and E. Brill. Spelling correction as an iterative process that exploits the collective knowledge of web users. In *Proceedings of EMNLP*, pages 293–300, 2004.

[9] D. Downey, S. Dumais, D. Liebling, and E. Horvitz. Understanding the relationship between searchers' queries and information goals. In *Proceedings of CIKM*, pages 449–458, 2008.

[10] G. E. Dupret and B. Piwowarski. A user browsing model to predict search engine click data from past observations. In *Proceedings of SIGIR*, pages 331–338, 2008.

[11] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin. Placing Search in Context: The Concept Revisited. *ACM Transactions on Information Systems*, 20(1):116–131, 2002.

[12] J. Guo, G. Xu, H. Li, and X. Cheng. A unified and discriminative model for query refinement. In *Proceedings of SIGIR*, pages 379–386, 2008.

[13] D. Hawking. Challenges in enterprise search. In *CRPIT '04: Proceedings of CRPIT*, pages 15–24, 2004.

[14] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of KDD*, pages 133–142, 2002.

[15] T. Joachims, L. Granka, B. Pan, H. Hembrooke, F. Radlinski, and G. Gay. Evaluating the accuracy of implicit feedback from clicks and query reformulations in web search. *ACM Trans. Inf. Syst.*, 25(2):7, 2007.

[16] R. Jones and K. L. Klinkner. Beyond the session timeout: automatic hierarchical segmentation of search topics in query logs. In *Proceedings of CIKM*, pages 699–708, 2008.

[17] R. Jones, B. Rey, O. Madani, and W. Greiner. Generating query substitutions. In *Proceedings of WWW*, pages 387–396, 2006.

[18] G. Kumaran and J. Allan. A case for shorter queries, and helping user create them. In *Proceedings of HLT*, pages 220–227, 2006.

[19] G. Kumaran and J. Allan. Effective and efficient user interaction for long queries. In *Proceedings of SIGIR*, pages 11–18, 2008.

[20] T. Lau and E. Horvitz. Patterns of search: analyzing and modeling web query refinement. In *Proceedings of UM*, pages 119–128, 1999.

[21] H. Ma, H. Yang, I. King, and M. R. Lyu. Learning latent semantic relations from clickthrough data for query suggestion. In *Proceeding of CIKM*, pages 709–718, 2008.

[22] G. Marchionini and R. White. Find what you need, understand what you find. *International Journal of Human-Computer Interaction*, 23(3):205–237, 2007.

[23] M. P. Marcus, M. A. Marcinkiewicz, and B. Santorini. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330, 1993.

[24] Q. Mei and K. Church. Entropy of search logs: how hard is search? with personalization? with backoff? In *Proceedings of WSDM*, pages 45–54, 2008.

[25] Q. Mei, H. Fang, and C. Zhai. A study of Poisson query generation model for information retrieval. In *Proceedings of SIGIR*, pages 319–326, 2007.

[26] Q. Mei, D. Zhou, and K. Church. Query suggestion using hitting time. In *Proceeding of CIKM*, 2008.

[27] D. Metzler and W. B. Croft. A Markov Random Field model for term dependencies. In *Proceedings of SIGIR*, pages 472–479, 2005.

[28] G. Pass, A. Chowdhury, and C. Torgeson. A picture of search. In *Proceedings of InfoScale*. ACM Press, 2006.

[29] N. Phan, P. Bailey, and R. Wilkinson. Understanding the relationship of information need specificity to search query length. In *Proceedings of SIGIR*, pages 709–710, 2007.

[30] J. M. Ponte and B. W. Croft. A language modeling approach to information retrieval. In *Proceedings of SIGIR*, pages 275–281, 1998.

[31] F. Radlinski, M. Kurup, and T. Joachims. How does clickthrough data reflect retrieval quality? In *Proceedings of CIKM*, pages 43–52, 2008.

[32] 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.

[33] B. Tan and F. Peng. Unsupervised query segmentation using generative language models and wikipedia. In *Proceeding of WWW*, pages 347–356, 2008.

[34] X. Wang and C. Zhai. Mining term association patterns from search logs for effective query reformulation. In *Proceeding of CIKM*, pages 479–488, 2008.

[35] X. Wei, F. Peng, and B. Dumoulin. Analyzing web text association to disambiguate abbreviation in queries. In *Proceedings of SIGIR*, pages 751–752, 2008.

[36] X. Xue, J. Jeon, and W. B. Croft. Retrieval models for question and answer archives. In *Proceedings of SIGIR*, pages 475–482, 2008.

[37] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems*, 22(2):179–214, 2004.