Numeric Query Ranking Approach

Jie Wu
Shanghai Jiao Tong University
Shanghai, 200240, P. R. China
wujie27@gmail.com

Yi Liu
Microsoft Research Asia
Beijing, 100080, P. R. China
lewisliu@microsoft.com

Ji-Rong Wen
Microsoft Research Asia
Beijing, 100080, P. R. China
jrwen@microsoft.com

ABSTRACT

We handle a special category of Web queries, queries containing numeric terms. We call them *numeric queries*. Motivated by some issues in ranking of numeric queries, we detect *numeric sensitive queries* by mining from retrieved documents using *phrase operator*. We also propose features based on numeric terms by extracting reliable numeric terms for each document. Finally, a ranking model is trained for numeric sensitive queries, combining proposed numeric-related features and traditional features. Experiments show that our model can significantly improve relevance for numeric sensitive queries.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Performance

Keywords

numeric sensitive queries; numeric queries; web search

1. INTRODUCTION

Among large set of web queries, certain portion of queries contain numbers. Users enter numbers for different kinds of intents. For search queries, numbers always contain more *precise* user intention and information than words. For example, for the query "pdf reader 9", user wants exactly "pdf reader 9", rather than "pdf reader 7" or "pdf reader 11". According to our analysis, only ∼6.8% of search queries contain numbers; however considering the large scale of Web queries, this is still a huge amount. In this paper, we regard queries containing numbers as *numeric queries*. As far as we know, there was no previous work that tackled the issues regarding numeric queries. It should be clearly noted that our work differs from previous work on temporal queries [2, 3, 4] because numeric terms are *explicit* in numeric queries, while temporal expressions are *implicit* in temporal queries.

According our analysis on a data set containing 13,920 queries, the average NDCG@1s of all queries and numeric queries are 45.9 and 62.0, respectively. The issues in ranking of numeric queries, which lead to the poor ranking performance, are listed below:

Issue 1: *The feature values of numeric terms are low compared to other terms in queries.* This problem is essentially caused by the high document frequency (DF) of numeric terms, which leads to the low TF-IDF even though term frequency (TF) of some numeric terms may be high. According to our calculation, average TF-IDFs of non-numeric terms (excluding stop words) and numeric terms in queries are 0.00041 and 0.1523, respectively; therefore, ranking for numeric queries is likely to be dominated by non-numeric terms in *queries*.

Issue 2: *Numeric terms in retrieved documents are messy and various.* The reasons are: 1) the high DF of numeric terms as discussed in issue 1 and 2) meanings of the same numeric term differ greatly in documents, which leads to the wrong match of numeric terms in queries and documents.

Motivated by above issues, as a first attempt, we propose a machine learned model to rank the *numeric sensitive queries* in this work. First, we use a numeric sensitive query detector to identify a numeric sensitive query. Second, the ranking model which is trained for numeric sensitive queries responds to the query. We use numeric-related features based on numeric terms extraction of each document to provide numeric evidences for documents.

2. NUMERIC SENSITIVE QUERY IDENTIFICATION

The simplest solution for issue 1 is to enlarge the feature values of numeric terms. However, we will show in experiments that simply boosting features of numeric terms hurts the relevance. The reason is that, numeric terms are not closely related to retrieval accuracy in some numeric queries, therefore, boosting features based on numeric terms may not be helpful. For example, numeric terms in queries like "download firefox 4", "Big Brother 2009" contain strong user intent, but for the query like "number 1 cameras on the market", if we force the search engine to retrieve documents containing exactly "number 1", the performance will drop.

We determine whether a numeric query is *numeric sensitive* or not by looking at the numeric distribution from top retrieved documents of the query. If the query is numeric sensitive, we boost the numeric-related features to emphasize numeric terms in the query. Specifically, to represent numeric distribution of query q in some streams s, we define *phrase operator* as

$$PO(q, s) = \{(x, f(x)) | f(x) = w(x, n_q)\#(x, t_q, s)\}, \quad (1)$$

WWW 2013 Companion, May 13–17, 2013, Rio de Janeiro, Brazil.
ACM 978-1-4503-2038-2/13/05.

where n_q is the numeric term in query q, t_q is the non-stopword term before n_q in q. t_q is closely related to n_q, such as "windows" in query: "buy windows 7". $w(x, n_q)$ can be any similarity measure of numeric terms x and n_q, giving a higher weight to x that are closer to n_q. $\#(x, t_q, s)$ is the number of times that x co-occur with t_q in s. In this way, the extracted numeric terms by phrase operator are less noisy and messy, which alleviates the problem in issue 2. Therefore, we use phrase operator to identify numeric sensitive queries. A numeric query q is *numeric sensitive* if

$$|PO(q, R_q)| > 1, \tag{2}$$

where R_q is top k documents of query q ($k = 10$). The larger size of phrase operator indicates that the query has a closer relation with various numeric terms, thus the query should be identified as the numeric sensitive query.

3. RANKING FOR NUMERIC SENSITIVE QUERIES

In this section, we show the numeric-related features we add for ranking and describe the training/ranking process.

Feature Generation: When a numeric query is identified as a numeric sensitive query, we apply our ranking model to the query to rank documents according to the numeric-related and other representative features of documents. For each document d, the numeric-related features are defined as $Sim(N(d), N(q))$, representing the similarity between numeric terms of document $N(d, q)$ and numeric terms of query $N(q)$. For similarity measure $Sim(\cdot, \cdot)$, we choose 0-1 binary match and L1 distance, thus generate two numeric-related features: NMatch and NSimilarity.

To extract reliable numeric terms from each document of queries, we use phrase operator to filter out the messy numeric terms. The reliable numeric distribution of document d for query q is defined as:

$$\begin{aligned} ND(d, q) = \gamma_t PO(q, d.t) + \gamma_b PO(q, d.b) \\ + \gamma_a PO(q, d.a) + \gamma_u PO(q, d.u), \end{aligned} \tag{3}$$

where $d.t, d.b, d.a, d.u$ represent the stream of title, body, anchor and url, respectively. The coefficients of four streams satisfy $\gamma_t > \gamma_a > \gamma_u > \gamma_b$. We set $N(d, q)$ as numeric term(s) x with highest $f(x)$ in $ND(d, q)$. Note that for the set of queries containing years, which is a special subset of numeric queries, we use a different set of coefficients.

Training/Ranking: We propose learning a ranking model for numeric sensitive queries detected by phrase operator, so that intuitively the documents with more matched numeric terms in queries are ranked higher. The ranking model can be learned in training phrase and be applied in testing phrase. We train our ranking model using RankNet [1].

4. EXPERIMENTS AND DISCUSSION

In the ranking experiment, we use a data set containing 13,920 queries obtained from a commercial search engine. Each query-URL pair is represented by a five-level human relevance label. We select queries that contain numeric terms from query set and refer them as numeric queries (942/13,920=6.8%). Then we use phrase operator to select numeric sensitive queries (542/942=57.5%) from numeric queries. In our experiment, we perform 5-fold cross validation and report the average of the individual runs.

Table 1: NDCG (N) results of baseline, BoostN and Numeric models on numeric queries (NQ) and numeric sensitive queries (NSQ) respectively.

Model	Data	*N@1*	*N@3*	*N@5*	N@10
Baseline	NQ	0.419	0.409	0.421	0.523
BoostN Δ_{ndcg}	NQ	0.407 (-2.9%)	0.39 (-2.8%)	0.411 (-2.4%)	0.489 (-6.5%)
Numeric Δ_{ndcg}	NQ	**0.420** (0.2%)	0.408 (-0.3%)	0.420 (-0.3%)	0.495 (-5.4%)
Baseline	NSQ	0.424	0.413	0.422	0.494
BoostN Δ_{ndcg}	NSQ	**0.428** (1.1%)	0.409 (-1.1%)	0.417 (-1.1%)	0.490 (-0.9%)
Numeric Δ_{ndcg}	NSQ	**0.443** (4.5%)	**0.422** (2.2%)	**0.430** (1.9%)	**0.496** (0.4%)

The three ranking models for comparison are:

1. **Baseline Model**: The baseline is the original ranking of the top 10 documents provided by Bing search engine, a competitive baseline model combining various features.

2. **BoostN Model**: For each document, we add TF-IDF of numeric terms in the query as a numeric-related feature, therefore feature values of numeric terms in queries are enlarged. To avoid new numeric-related features dominating the ranking, we add 4 representative features including BM25 score, words found in title, static rank, and output ranking score of baseline model so as to balance the ranking.

3. **Numeric Model**: Similiar as BoostN model, features of Numeric model include our novel numeric-related features (NMatch and NSimilarity) proposed, and 4 representative features mentioned above.

We conduct experiments to apply the three ranking models to the set of numeric queries (NQ) and numeric sensitive queries (NSQ), respectively. Performances of the models are shown in Table 1. Performances of both BoostN and Numeric models for NSQ are better than for NQ, because numeric terms have a higher confidence in NSQ. This result also verifies the effectiveness of our numeric sensitive query detector. For NSQ, the performance of Numeric model is significantly better than both baseline and BoostN model. This demonstrates the proposed novel numeric-related features capture more numeric evidences than traditional features of numeric terms such as TF-IDF. Overall, results of the proposed model show that integrating numeric evidences into Web search for numeric queries is a promising direction that can significantly improve the Web search performance.

5. REFERENCES

[1] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *ICML*, pages 89–96, 2005.

[2] W. Dakka, L. Gravano, and P. Ipeirotis. Answering general time sensitive queries. In *CIKM*, pages 1437–1438, 2008.

[3] F. Diaz and R. Jones. Using temporal profiles of queries for precision prediction. In *SIGIR*, pages 18–24, 2004.

[4] D. Metzler, R. Jones, F. Peng, and R. Zhang. Improving search relevance for implicitly temporal queries. In *SIGIR*, pages 700–701, 2009.