

# Parameter Sensitivity in the Probabilistic Model for Ad-hoc Retrieval

Ben He

Department of Computing Science  
University of Glasgow  
Glasgow, The United Kingdom  
ben@dcs.gla.ac.uk

Iadh Ounis

Department of Computing Science  
University of Glasgow  
Glasgow, The United Kingdom  
ounis@dcs.gla.ac.uk

## ABSTRACT

The term frequency normalisation parameter sensitivity is an important issue in the probabilistic model for Information Retrieval. A high parameter sensitivity indicates that a slight change of the parameter value may considerably affect the retrieval performance. Therefore, a weighting model with a high parameter sensitivity is not robust enough to provide a consistent retrieval performance across different collections and queries. In this paper, we suggest that the parameter sensitivity is due to the fact that the query term weights are not adequate enough to allow informative query terms to differ from non-informative ones. We show that query term reweighing, which is part of the relevance feedback process, can be successfully used to reduce the parameter sensitivity. Experiments on five Text REtrieval Conference (TREC) collections show that the parameter sensitivity does remarkably decrease when query terms are reweighed.

**Categories and Subject Descriptors:** H.3.3 [Information Storage and Retrieval]: Retrievalmodels; **General Terms:** Performance, Experimentation; **Keywords:** Query term reweighing, Relevance feedback, Parameter sensitivity

## 1. INTRODUCTION

In Information Retrieval (IR), it is a crucial issue to rank retrieved documents in decreasing order of relevance. A recent survey on the query logs from real Web search engine users concluded that users rarely look beyond the top returned documents [16]. Therefore, it is important to rank the highly relevant documents at the top of the retrieved list. Usually, the document ranking is based on a weighting model. In particular, most weighting models apply a term frequency (*tf*) normalisation method to normalise term frequency, i.e. the number of occurrences of the query term in the document.

Various *tf* normalisation methods have been proposed in the literature, e.g. the pivoted normalisation [24] in the

vector space model [23], the normalisation method of the BM25 weighting model [22], normalisation 2 [1] and normalisation 3 [1, 14] in the Divergence from Randomness (DFR) framework [1]. All the aforementioned normalisation methods normalise term frequency according to document length, i.e. the number of tokens in the document. Each of the aforementioned normalisation methods involves the use of a parameter. The setting of these parameter values usually has an important impact on the retrieval performance of an IR system<sup>1</sup> [5, 14, 15]. In particular, if the retrieval performance of a weighting model is sensitive to a slight change of its parameter value, the weighting model may not be robust enough to provide consistent retrieval performance. This is referred to as the *parameter sensitivity* issue.

In a practical IR context, parameter sensitivity is a very important issue. Since relevance assessment and training data are not always available in a practical environment, it is crucial to ensure that the parameter value used provides a robust retrieval performance. The parameter sensitivity issue has been previously studied in the context of the language modelling approach [18, 26, 27]. In addition, several weighting models that are less sensitive than the classical ones were generated in an axiomatic approach based on parameter constraints [8, 9]. Nevertheless, little work has been done to actually reduce the parameter sensitivity of a weighting model. In this paper, we base our study on the classical BM25 probabilistic model, and the PL2 model of the Divergence from Randomness (DFR) probabilistic framework. These two models have been shown to be effective in various previous TREC experiments [7].

The main contributions of this paper are two-fold. First, we provide a better understanding and explanation of the parameter sensitivity. We argue that parameter sensitivity is caused by the existence of non-informative terms in the query. Second, we show that parameter sensitivity can be reduced by applying query term reweighing, which is part of the relevance feedback.

The rest of this paper is organised as follows. Section 2 introduces the related work, including the BM25 and PL2 models, and previous research on the parameter sensitivity issue. Section 3 provides an explanation for the manifestation of the parameter sensitivity and suggests to apply an

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CIKM'07, November 6–8, 2007, Lisboa, Portugal.

Copyright 2007 ACM 978-1-59593-803-9/07/0011 ...\$5.00.

<sup>1</sup>For instance, training a retrieval system using the PL2 weighting model [1] on TREC 10 ad-hoc queries, gives an MAP of 0.2397 on the TREC 9 queries, compared to an MAP of 0.2174 using the default ( $c = 1$ ) setting. This difference is statistically significant ( $p \leq 0.0009$ ).

appropriate query term reweighing to reduce the parameter sensitivity. Section 4 describes the experimental setting and methodology, and Section 5 provides analysis and discussion on the experimental results. Finally, Section 6 concludes on the paper and suggests possible future research directions.

## 2. RELATED WORK

In this section, we introduce the BM25 and PL2 models in Section 2.1, and briefly describe previous research on the parameter sensitivity issue in Section 2.2.

### 2.1 The BM25 and PL2 Probabilistic Weighting Models

As one of the most established weighting models, Okapi’s *BM25* computes the relevance score of a document  $d$  for a query  $Q$  by the following formula [22]:

$$score(d, Q) = \sum_{t \in Q} w^{(1)} \frac{(k_1 + 1)tfn}{k_1 + tfn} \cdot qtw \quad (1)$$

where

- $w^{(1)}$  is the *idf* factor, which is given by:

$$w^{(1)} = \log_2 \frac{N - N_t + 0.5}{N_t + 0.5}$$

where  $N$  is the number of documents in the whole collection.  $N_t$  is the document frequency of term  $t$ , i.e. the number of documents containing  $t$ .

- $qtw$  is the query term weight that is given by

$$\frac{(k_3 + 1)qtf}{k_3 + qtf}$$

where  $qtf$  is the number of occurrences of the given term in the query.  $k_3$  is a parameter. Its default setting is  $k_3 = 1000$  [22].

- $tfn$  is the normalised term frequency of the given term  $t$ .  $k_1$  is a parameter. Its default setting is  $k_1 = 1.2$  [22].

The  $tf$  normalisation component of the BM25 formula is:

$$tfn = \frac{tf}{(1 - b) + b \cdot \frac{l}{avg\mathcal{L}}} \quad (2)$$

where  $l$  and  $avg\mathcal{L}$  are the document length and the average document length in the collection, respectively.  $tf$  is the term frequency in the document. The document length refers to the number of tokens in a document.  $b$  is a parameter. The default setting is  $b = 0.75$  [22]. Singhal et al.’s *pivoted normalisation*, for normalising the  $tf \cdot idf$  weight in the context of the vector space model [24], can be seen as a generalisation of the above BM25’s  $tf$  normalisation component.

*PL2* is one of the Divergence from Randomness (DFR) document weighting models [3]. The idea of the DFR models is to infer the importance of a query term in a document by measuring the divergence of the term’s distribution in the document from its distribution in the whole collection that is assumed to be random. In the PL2 model, this random

distribution is modelled by an approximation to the Poisson distribution with the use of the Laplace succession for normalising the relevance score. Using the PL2 model, the relevance score of a document  $d$  for a query  $Q$  is given by:

$$score(d, Q) = \sum_{t \in Q} qtw \cdot \frac{1}{tfn + 1} (tfn \cdot \log_2 \frac{tfn}{\lambda} + (\lambda - tfn) \cdot \log_2 e + 0.5 \cdot \log_2(2\pi \cdot tfn)) \quad (3)$$

where  $\lambda$  is the mean and variance of a Poisson distribution. It is given by  $\lambda = F/N$ .  $F$  is the frequency of the query term in the collection and  $N$  is the number of documents in the collection. The query term weight  $qtw$  is given by  $qtf/qtf_{max}$ ;  $qtf$  is the query term frequency.  $qtf_{max}$  is the maximum query term frequency among the query terms.

The normalised term frequency  $tfn$  is given by the so-called *normalisation 2*:

$$tfn = tf \cdot \log_2(1 + c \cdot \frac{avg\mathcal{L}}{l}), (c > 0) \quad (4)$$

where  $l$  is the document length and  $avg\mathcal{L}$  is the average document length in the whole collection.  $tf$  is the original term frequency.  $c$  is the parameter of normalisation 2. Its default setting is  $c = 7$  for short queries and  $c = 1$  for long queries [1].

### 2.2 Previous Work on Parameter Sensitivity

The parameter sensitivity issue has drawn the attention of several previous research. Zhai & Lafferty addressed the parameter sensitivity issue of the smoothing technique for language modelling. They found that query length, the number of unique terms in a query, has a considerable impact on the parameter sensitivity. In particular, the parameters of the smoothing methods are very sensitive for long queries [26, 27]. Similar findings were also observed for the BM25 and PL2 weighting models [13]. Moreover, Fang et al. generated some weighting models that are less sensitive than current one, using their axiomatic approach based on a set of parameter constraints [8, 9].

A quantitative analysis of the parameter sensitivity was conducted by Metzler in [18]. Two measures, namely Entropy ( $H$ ) and Spread ( $S$ ), were proposed to define the parameter sensitivity. The Entropy ( $H$ ) measure is given as follows:

$$H = - \int P(opt, x) \log_2 P(opt, x) \quad (5)$$

where  $P(opt, x)$  is the probability of the parameter value  $x$  being the optimal one. In [18],  $P(opt, x)$  is computed using Bayes’s rule.

*Spread* measures the flatness of a posterior distribution over a set of parameter values. It is given as follows:

$$S = m(max, X) - m(min, X) \quad (6)$$

where  $m(max, X)$  (resp.  $m(min, X)$ ) is the maximum (resp. minimum) posterior over a set  $X$  of parameter values. For example, if the retrieval performance evaluation measure used is the mean average precision (MAP),  $m(max, X)$  is the maximum MAP, and  $m(min, X)$  is the minimum one. In practise, the lower Spread or Entropy is, the lower parameter sensitivity an IR model has. In addition, a low

Spread is preferred over a low Entropy in order to ensure a low parameter sensitivity [18].

The above mentioned research either addressed or quantitatively analysed the parameter sensitivity issue. Nevertheless, little work has been actually done to reduce the parameter sensitivity of a weighting model. In the next section, we explain the parameter sensitivity issue, and show how to apply query term reweighing to reduce the parameter sensitivity of the probabilistic model.

### 3. QUERY TERM WEIGHTS AND PARAMETER SENSITIVITY

In this section, we provide an explanation for the parameter sensitivity issue. We suggest that the parameter sensitivity is caused by inadequate query term weighting. We also propose to reduce the parameter sensitivity by reweighing query terms.

As mentioned in the previous section, query length has an important impact on the parameter sensitivity of *tf* normalisation. In particular, the parameter setting tends to be more sensitive for long queries than for short queries [26, 27].

One of the characteristics differentiating long queries from short queries is that long queries have much more non-informative query terms than short queries. As a consequence, for a long query, it is necessary to address the difference in the informativeness among query terms in the weighting models. For example, in the BM25 and PL2 weighting models (see Equations (1) and (3)), a query term weight (*qtw*) measure is employed to represent the relative informativeness among query terms. Such a query term weight measure is adequate for short queries, because a short query usually consists of highly informative query terms. When the query gets longer, it is “contaminated” by non-informative query terms. Although the query term weight measure is meant to reflect the informativeness of a query term, it accounts only for the query term frequency, and it is still not adequate to differentiate informative query terms from non-informative ones. In this case, a *tf* normalisation parameter, providing a “harsh” normalisation, is needed to neutralise the effect of non-informative query terms on the document ranking. We explain the notion of “harsh” normalisation as follows.

Harter [10] and Amati [1] suggested that document length and term frequency have a linear relationship. Such a linear relationship can be indicated by the linear correlation between these two variables. He & Ounis suggested that the purpose of *tf* normalisation is to adjust the linear dependence between document length and term frequency [14]. They also showed that document length and term frequency are positively correlated. However, when *tf* normalisation is applied, the correlation between these two variables decreases until it reaches a large negative value [14]. In Section 5, we will show that the optimised parameter value of short queries gives a small negative correlation, and that of long queries gives a relatively large negative correlation. In the IR weighting models, e.g. BM25 and PL2, the relevance score usually increases with term frequency. Hence, a large negative correlation indicates that the contribution of each occurrence of a query term on the document ranking decreases rapidly with document length increasing. Therefore, the smaller this correlation value is, the harsher the

*tf* normalisation process is. For long queries, the retrieval performance decreases radically if the parameter value used does not provide a harsh enough normalisation, which leads to a notable parameter sensitivity. One way of dealing with parameter sensitivity is to use the query term weights to address the difference in the informativeness among query terms. However, in current probabilistic IR models, the query terms weights depend only on the occurrences of query terms in the query, which is usually not adequate.

The above explanation suggests that the parameter sensitivity issue is due to the fact that the query term weights cannot adequately reflect the informativeness of each query term. For this problem, we hypothesise that if we reweigh the query terms to achieve an adequate query term weighting, the parameter sensitivity can be reduced. A query term reweighing process takes into account each query term’s distribution in one or a set of assumed highly relevant document(s), returned by the first-pass retrieval. The query terms are reweighed accordingly. Query term reweighing is usually considered as part of the so-called blind relevance feedback technique [4]. In the next sections, we conduct experiments to test the effect of query term reweighing on reducing the parameter sensitivity.

### 4. EXPERIMENTAL SETTING AND METHODOLOGY

The purpose of our experiments is to examine the impact of query term reweighing on the parameter sensitivity. In particular, as per Section 3, we expect the use of query term reweighing to reduce the parameter sensitivity. We introduce the experimental setting in Section 4.1, the term weighting model used for query term reweighing in Section 4.2, and the experimental methodology in Section 4.3.

#### 4.1 Experimental Setting

We experiment on five standard TREC collections. The five collections used are the disk1&2, disk4&5 (minus the Congressional Record on disk4) of the classical TREC collections<sup>2</sup>, and the WT2G [12], WT10G [11] and .GOV2 [6] Web collections<sup>3</sup>. The test queries used are the TREC topics that are numbered from 51 to 200 for disk1&2, from 301 to 450 and from 601-700 for disk4&5, from 401 to 450 for WT2G, from 451 to 550 for WT10G, and from 701 to 850 for .GOV2, respectively (see Table 1). All the test topics used are ad-hoc ones, which require finding as many relevant documents as possible [25].

Each TREC topic consists of three fields, i.e. title, description and narrative. We experiment with two types of queries with respect to the use of different topic fields. These two types of queries are:

- **Title-only (T) queries:** Only the title topic field is used.
- **Full (TDN) queries:** All the three topic fields (title, description and narrative) are used.

<sup>2</sup>Related information of disk1&2 and disk4&5 of the TREC collections can be found from the following URL: [http://trec.nist.gov/data/docs\\_eng.html](http://trec.nist.gov/data/docs_eng.html).

<sup>3</sup>Related information of these three TREC Web collections can be found from the following URL: [http://ir.dcs.gla.ac.uk/test\\_collections/](http://ir.dcs.gla.ac.uk/test_collections/).

**Table 1: Details of the five TREC collections used in our experiments. The second row gives the topic numbers associated to each collection.  $N$  is the number of documents in the given collection.**

	disk1&2	disk4&5	WT2G	WT10G	.GOV2
Topics	51-200	301-450 and 601-700	401-450	451-550	701-850
N	741,860	528,155	247,491	1,692,044	25,205,179

Table 2 gives the average query length of the title-only and full queries. We can see that these two types of queries have largely different query length. Title-only queries usually contain only few keywords, while full queries are much longer than the title-only ones. As reported in [26], in the context of language modelling, query length has an important impact on the setting of parameter values. Therefore, we experiment with two different types of queries to check if query length affects our conclusions.

**Table 2: The average query length of title-only (T) and full (TDN) queries.**

	disk1&2	disk4&5	WT2G	WT10G	.GOV2
T	3.73	2.62	2.40	2.42	2.88
TDN	31.77	18.48	16.74	11.23	15.66

The experiments in this paper are conducted using the Terrier platform [19]. Standard stopword removal and Porter’s stemming algorithm are applied in all our experiments. The evaluation measure used is mean average precision (MAP) that is a standard evaluation measure in the TREC ad-hoc tasks [25].

## 4.2 The Bo1 Term Weighting Model

For query term reweighing, we apply the Bo1 Divergence from Randomness (DFR) term weighting model that is based on the Bose-Einstein statistics [1]. The reason for using Bo1 is twofold. First, Bo1 has been previously shown to be effective in extensive experiments on the TREC collections [1, 2, 20]; Second, Bo1 is a parameter-free term weighting model that does not require any tuning so that our study focuses only on the parameter sensitivity of the  $tf$  normalisation parameters.

The idea of the DFR term weighting model is to measure the informativeness of a term by the divergence of its distribution in the top-ranked documents from a random distribution. Therefore, it uses a set of top-ranked documents, returned by the first-pass retrieval, for relevance feedback. Based on extensive training on the TREC collections,  $exp\_doc$ , the number of top-ranked documents used for relevance feedback, is robust from 3 to 10 [1]. In this paper, we arbitrarily set  $exp\_doc$  to 5. According to our experiments on four out of five TREC collections used, changing  $exp\_doc$  changes the absolute retrieval performance, but does not affect the effectiveness of query term reweighing on reducing the parameter sensitivity. Therefore, we only report experiments using  $exp\_doc = 5$  in this paper. Using the Bo1 model, the weight  $w$  of a term  $t$  in the top-ranked documents is given by:

$$w(t) = tf_x \cdot \log_2 \frac{1 + P_n}{P_n} + \log_2(1 + P_n) \quad (7)$$

where  $tf_x$  is the frequency of the query term in the top-ranked documents.  $P_n$  is given by  $\frac{F}{N}$ .  $F$  is the frequency of the term in the collection and  $N$  is the number of documents in the collection.

After assigning a weight to each unique term that appears in the top-ranked documents, the query term weight  $qtw$  of each query term is revised by a parameter-free query term reweighing formula:

$$\begin{aligned} qtw &= \frac{qt f}{qt f_{max}} + \frac{w(t)}{\lim_{F \rightarrow tf_x} w(t)} \\ &= F_{max} \log_2 \frac{1 + P_{n,max}}{P_{n,max}} + \log_2(1 + P_{n,max}) \end{aligned} \quad (8)$$

where  $qt f$  is the query term weight.  $\lim_{F \rightarrow tf_x} w(t)$  is the upper bound of  $w(t)$ .  $P_{n,max}$  is given by  $F_{max}/N$ .  $F_{max}$  is the frequency  $F$  of the term with the maximum  $w(t)$  in the top-ranked documents. If a query term does not appear in the top-ranked documents, its query term weight remains equal to the original one.

## 4.3 Experimental Methodology

In our experiments, we evaluate if query term reweighing reduces parameter sensitivity. We evaluate using the Entropy ( $H$ ) and Spread ( $S$ ) measures following the work in [18]. Moreover, we evaluate by conducting cross-collection training.

The idea of the evaluation by cross-collection training is as follows. Because of the existence of parameter sensitivity, the parameter value trained on one collection, may not be effective on a different collection. If the query term reweighing process successfully reduces parameter sensitivity, the parameter setting, trained on one collection, should be as good as the optimal one when it is applied for a given new collection. In this case, retrieval performance is not hurt by parameter sensitivity.

In our evaluation by cross-collection training, we use each of the five TREC test collections as the target collection, and the other four collections for training the parameters. On the target collection, we compare the relative difference ( $\Delta$ ) between the resulting MAP values of the parameter setting optimised on the target collection, and the parameter setting optimised on each of the four training collections (there are four  $\Delta$  values for each collection for testing, each  $\Delta$  value corresponds to a training collection). If parameter sensitivity is reduced by query term reweighing, we will see a smaller  $\Delta$  value for the reweighed queries, than that for the original queries.

Moreover, following the work by Metzler [18], we use the Entropy ( $H$ ) and Spread ( $S$ ) measures (see Section (2.2)) to define parameter sensitivity. If parameter sensitivity is successfully reduced by query term reweighing, we will expect to see smaller resulting Entropy and Spread values.

For the computation of Entropy, instead of using Bayes’s rule, we simplify the computation by converting each mean average precision (MAP) value obtained to a ratio as follows:

$$ratio(opt, x_i) = \frac{MAP}{MAP_{opt}}$$

where  $MAP_{opt}$  is the optimal MAP. Note that the sum of  $ratio(opt, x)$  over all possible parameter values is not 1, and the above  $ratio(opt, x_i)$  is not a probability. However, since

we are only interested in how much variation of retrieval effectiveness is there over a working range of parameter values, the above definition is adequate for our study. In our experiments, the  $tf$  normalisation parameters are optimised using the Simulated Annealing method [17]. Our optimisation strategy is to find the parameter value that maximises MAP over a wide range of possible parameter values.

For a set of parameter values  $X$ , we generate a ratio  $ratio(opt, x_i)$  for each parameter value  $x_i$ . The Entropy measure is given by:

$$H = \sum_{x_i \in X} -ratio(opt, x_i) \cdot \log_2 ratio(opt, x_i)$$

An important issue for computing the Entropy measure is how to sample the set  $X$  of parameter values. The sampling of the parameter values is very important for our computation of the Entropy measure. An inappropriate sampling strategy can lead to biased experimental results, and consequently, to erroneous conclusions.

For BM25's  $tf$  normalisation parameter, it is easy to create the samples because BM25's  $tf$  normalisation parameter  $b$  corresponds to a linear trade-off between a gentle and a harsh normalisation. We sample uniformly its parameter  $b$  from 0.05 to 1 with a unique interval of 0.05. For PL2, we could also sample its parameter values uniformly. However, for PL2, because the relation between its parameter  $c$  and the normalisation function is not as clear as it is for BM25, a uniform sampling strategy could lead to a particular range of parameter values that is over-sampled, which can cause biased experimental results. Therefore, for PL2, we study the relation between the  $tf$  normalisation function and its parameter  $c$ . Assuming that  $tfn$  is a function of its parameter, we can derive  $tfn'$ , the derivative of  $tfn$  with respect to its parameter. This derivative provides an indication of how  $tfn$  varies with respect to the change of its parameter value, which helps us sample the parameter values accordingly.

We derive the derivative  $tfn'(c)$  for PL2 as follows (see Equation (4) for the normalisation function):

$$\begin{aligned} tfn'(c) &= \left( tf \cdot \log_2 \left( 1 + c \cdot \frac{avgJ}{l} \right) \right)'(c) \\ &= \frac{tf \frac{avgJ}{l}}{\left( 1 + c \frac{avgJ}{l} \right) \log_e 2} \end{aligned} \quad (9)$$

where  $avgJ$  is the average document length in the whole collection, and  $l$  is the document length.

The derivative  $tfn'(c)$  is a decreasing function of its parameter  $c$ . When  $c$  increases,  $tfn'(c)$  approaches 0. We have  $\lim_{c \rightarrow \infty} tfn'(c) = 0$ , which infers that the increasing rate of  $tfn$  diminishes when  $c$  is very large. Therefore, the variation of  $tfn$  tends to be stable when  $c$  gets larger. Since  $tfn$  is a logarithmic function of  $c$ , the interval between two adjacent sampled  $c$  values increases when  $\log_2 c$  increases. We sample the following  $c$  values from  $[1, 32]^4$  with an increasing interval between adjacent sampled values: from 1 to 4 with an interval of 1, from 6 to 8 with an interval of 2, from 12 to 16 with an interval of 4, and from 24 to 32 with an interval of 8. Ten parameter values, 1, 2, 3, 4, 6, 8, 12, 16, 24 and 32, are sampled in total.

<sup>4</sup>For ad-hoc retrieval on various TREC collections, the optimal parameter  $c$  values are normally within this range [1].

On a given collection, using a given weighting model (i.e. BM25 or PL2), we compute the Entropy  $H$  and Spread  $S$  measures for the following five different query settings:

- OQ, T: Original title-only queries.
- RQ, T: Reweighed title-only queries.
- OQ, TDN: Original full queries.
- RQ, TDN: Reweighed full queries.
- TRQ, TDN: Reweighed full queries with the use of the query terms in the title topic field in the first-pass retrieval.

The last setting (TRQ, TDN) comes from the following idea. We suggest that in the first-pass retrieval, the use of all query terms in a full query, including many non-informative query terms, can possibly cause the query term reweighing process to be biased towards those non-informative query terms. Therefore, we want to test if the use of the few most informative query terms, instead of all query terms, in the first-pass retrieval can lead to a reduced parameter sensitivity and a better retrieval performance. Since query terms in the title topic field are usually very informative, we use them in the first-pass retrieval.

Both the Entropy and Spread measures indicate the parameter sensitivity of the weighting models used. Entropy indicates the variation of MAP over the parameter value set  $X$ , and Spread indicates the flatness of the MAP distribution over  $X$ . If we observe that either Spread or Entropy, or both the measures, of the reweighed queries (RQ or TRQ) are clearly lower than those of the original queries (OQ), we conclude that query term reweighing successfully reduces parameter sensitivity. In addition, if we observe a clearly higher Spread and a clearly lower Entropy brought by query term reweighing, following [18], we consider the reweighed queries to have a higher parameter sensitivity than the original queries. This is because low Spread is preferred over low Entropy, as mentioned in Section 2.2. In the next section, we provide an analysis of the experimental results.

## 5. EXPERIMENTAL RESULTS

In this section, we present and analyse the experimental results. We firstly give the evaluation results by cross-collection training.

Figure 1 plots the  $\Delta$  values obtained against the training collection used. In sub-figures 1(a) - 1(e), each coordinate on the X-axis corresponds to each of the training collections used.  $\Delta$  is the relative difference between the resulting MAP values of the parameter setting optimised on the target collection, and the parameter setting optimised on each of the four training collections. The Y-axis corresponds to  $\Delta$ , the difference between the MAP values, given by the parameter settings optimised on the target collection, and on the training collection, respectively. From sub-figures 1(a) - 1(e), on one hand, we can see that a large number of points of the  $\Delta$  values stay at the bottom of the figures. This indicates no parameter sensitivity problem as the  $\Delta$  values are very small. On the other hand, some points stand out on the top of the figures, indicating high  $\Delta$  values. A high  $\Delta$  value implies the fact that the parameter value, trained on another collection, cannot be reused on the target collection. For

example, on disk1&2, when the parameter setting of PL2 is trained on WT10G, we observe a high  $\Delta$  value (nearly 8%) for the full original queries (see the curve labelled as (PL2, OQ, TDN), marked by stars, in Figure 1(a)). When the queries are reweighed, the corresponding  $\Delta$  value becomes smaller (see the curve labelled as (PL2, RQ, TDN), marked by inversed solid triangles, in Figure 1(a)), which shows a success of query term reweighing in reducing parameter sensitivity. We also observe cases where query term reweighing does not bring a lower  $\Delta$  value than the original queries, especially when the parameter value is trained on WT2G. For example, on .GOV2 (see Figure 1(e)). When the parameter values are trained on WT2G, we observe several high  $\Delta$  values for both the original and the reweighed queries.

For title-only or full queries, we compare the  $\Delta$  values of the original queries (OQ), with those of the reweighed queries (RQ). For full queries, we also compare the  $\Delta$  values of OQ with those of the reweighed queries using the title topic field in the first-pass retrieval (TRQ). Over all the five collections used, we have 120 comparisons between the  $\Delta$  values of OQ and RQ (resp. TRQ). We observe that in 72 out of these 120 comparisons, the  $\Delta$  values are reduced after query term reweighing takes place. The p-value is 0.0358 according to the sign test, which is statistically significant at 0.05 level. Moreover, in 48 cases where query term reweighing does not reduce the  $\Delta$  values, the  $\Delta$  values of OQ and RQ (resp. TRQ) marginally differ (less than 1%) from each other in 38 cases. In other words, RQ and TRQ have a non-marginal higher parameter sensitivity than OQ in only 10 out of 120 cases. Therefore, query term reweighing is shown to be effective in reducing parameter sensitivity according to our evaluation by cross-collection training. Retrieval performance does not seem to be hurt by parameter sensitivity after query term reweighing. Next, we present the evaluation results using the Entropy and Spread measures.

Table 3 lists the optimised parameter values obtained by the optimisation process. The linear correlations between document length and (normalised) term frequency, corresponding to the optimised parameter values, are listed in Table 4. From Tables 3 and 4, we can see that the optimised parameter values for the full queries provide larger negative correlation values than those for the title-only queries. This observation confirms our suggestion in Section 3 that long queries require a relatively harsh normalisation to achieve an optimised retrieval performance.

**Table 3: The optimised (opt) parameter settings for the original title-only (T) and full (TDN) queries.**

	PL2		BM25	
	$c_{opt,T}$	$c_{opt,TDN}$	$b_{opt,T}$	$b_{opt,TDN}$
disk1&2	5.13	1.36	0.34	0.77
disk4&5	11.57	1.97	0.34	0.71
WT2G	31.30	3.54	0.17	0.70
WT10G	11.75	1.90	0.27	0.51
.GOV2	6.56	1.99	0.39	0.69

Tables 5 and 6 list the Entropy (H) and Spread (S) values obtained on the five TREC collections using five different query settings (see Section 4.3 for the definitions of the 5 query settings). Moreover, Figure 2 provides a visual comparison between the parameter sensitivity of the original and reweighed queries. Since lower Entropy and Spread values indicate lower parameter sensitivity, we expect the plots of

**Table 4: The linear correlations ( $\rho$ ) between document length and (normalised) term frequency given by the optimised (opt) parameter values for title-only (T) and full (TDN) queries.**

	PL2		BM25	
	$\rho_{opt,T}$	$\rho_{opt,TDN}$	$\rho_{opt,T}$	$\rho_{opt,TDN}$
disk1&2	-0.1068	-0.1460	-0.09970	-0.1474
disk4&5	-0.09257	-0.1505	-0.09952	-0.1585
WT2G	-0.07222	-0.1903	-0.07422	-0.2211
WT10G	-0.1159	-0.1604	-0.1148	-0.1551
.GOV2	-0.1350	-0.2245	-0.1400	-0.2562

RQ and TRQ to be at the bottom-left corner of the graphs. From Tables 5, 6 and Figure 2, we have the following observations.

First, we compare the Entropy values of the full original queries to those of the title-only original queries. For PL2, the Entropy values of the full original queries are clearly higher than those for the title-only original queries. For example, on disk1&2, using PL2, the Entropy values of the title-only and full original queries are  $H_{OQ} = 0.3716$  and  $H_{OQ} = 2.080$ , respectively. The latter  $H_{OQ}$  value is clearly higher than the former one (see Table 5). This shows that query length has an important impact on PL2’s Entropy value. However, for BM25, the Entropy values of the title-only and full original queries are overall comparable (see BM25’s  $H_{OQ}$  values in Table 5, and BM25’s OQ plots in Figure 2). We therefore conclude that BM25’s Entropy value is not affected by query length, while PL2’s is. This shows that PL2 is more likely to have high Entropy than BM25.

Second, we compare the Spread values of the full original queries to those of the title-only original queries. For both BM25 and PL2, the original title-only and full queries have similar Spread values (see the  $S_{OQ}$  values in Table 6), apart from on disk1&2, where the original full queries have much higher Spread values than the original title-only queries (see rows disk1&2 in Table 6). We conclude that the Spread measure of the original queries is not affected by query length.

Third, we look at the effectiveness of query term reweighing on reducing the parameter sensitivity of title-only queries. For title-only queries, applying query term reweighing does not seem to reduce the Entropy values. From Table 5, we find that the Entropy values of the reweighed title-only queries ( $H_{RQ}$ ) are very comparable with those of the original title-only queries ( $H_{OQ}$ ). Although the  $H_{RQ}$  values are higher than  $H_{OQ}$  in 8 out of 10 cases (see the  $H_{RQ}$  values marked with stars for title-only queries in Table 5), the difference between  $H_{RQ}$  and  $H_{OQ}$  is not large enough to claim a success of query term reweighing in reducing Entropy for title-only queries. However, from Table 6, we find that the use of query term reweighing does clearly reduce the Spread values for the title-only queries in 7 out of 10 cases (see the  $S_{OQ}$  and  $S_{RQ}$  values of the title-only queries in Table 6). This can also be observed from Figure 2. From Figure 2, we can see that the plots of query setting (RQ, T) have a clearly lower value on the  $S$  axis than the plots of query setting (OQ, T) in 7 out of 10 cases. Therefore, we conclude that overall applying query term reweighing does reduce the parameter sensitivity of title-only queries.

Finally, we check if query term reweighing also reduces the parameter sensitivity of full queries. For full queries, we observe that applying query term reweighing reduces En-

tropy and Spread in 9 out of 10 cases (see the  $H_{RQ}$ ,  $H_{TRQ}$ ,  $S_{RQ}$  and  $S_{TRQ}$  values for full queries marked with stars in Tables 5 and 6, and the RQ and TRQ plots in Figure 2). Therefore, we conclude that applying query term reweighing also reduces the parameter sensitivity of full queries.

To summarise, query term reweighing has successfully reduced the parameter sensitivity according to our experiments on five TREC test collections, for both title-only and full queries.

**Table 5: The Entropy ( $H$ ) values obtained over the sampled parameter values. OQ and RQ stand for the original and reweighed queries, respectively. TRQ stands for the reweighed queries using terms in the title topic field in the first-pass retrieval. An  $H_{RQ}$  value marked with a star indicates a drop in the Entropy value from  $H_{OQ}$ .**

	T		TDN		
	$H_{OQ}$	$H_{RQ}$	$H_{OQ}$	$H_{RQ}$	$H_{TRQ}$
PL2					
disk1&2	0.3716	0.3332*	2.080	1.419*	1.467*
disk4&5	0.3098	0.3052*	1.114	0.6429*	0.6609*
WT2G	0.9227	1.089	0.9238	0.6228*	0.6049*
WT10G	0.4677	0.6124	1.047	0.9031*	0.8565*
.GOV2	0.6130	0.6100*	1.522	1.189*	1.093*
BM25					
disk1&2	1.496	1.294*	2.322	1.614*	1.707*
disk4&5	0.9877	0.8591*	1.473	1.100*	1.049*
WT2G	2.950	2.856*	1.736	1.707*	1.784
WT10G	2.118	2.003*	1.811	1.683*	1.964
.GOV2	2.402	2.290*	2.321	2.024*	2.084*

**Table 6: The Spread ( $S$ ) values obtained over the sampled parameter values. OQ and RQ stand for the original and reweighed queries, respectively. TRQ stands for the reweighed queries using terms in the title topic field in the first-pass retrieval. An  $S_{RQ}$  value marked with a star indicates a drop in the Spread value from  $S_{OQ}$ .**

	T		TDN		
	$S_{OQ}$	$S_{RQ}$	$S_{OQ}$	$S_{RQ}$	$S_{TRQ}$
PL2					
disk1&2	0.0221	0.0339	0.2421	0.2126*	0.2139*
disk4&5	0.1517	0.0139*	0.1727	0.1331*	0.1283*
WT2G	0.1962	0.2154	0.1192	0.1273	0.1295
WT10G	0.1198	0.0229*	0.1692	0.1345*	0.1391*
.GOV2	0.2265	0.1468*	0.1563	0.0783*	0.0745*
BM25					
disk1&2	0.0456	0.0486	0.2578	0.2128*	0.2193*
disk4&5	0.1393	0.0297*	0.1663	0.1313*	0.1313*
WT2G	0.1933	0.1083*	0.1583	0.0444*	0.1528*
WT10G	0.1781	0.0799*	0.1758	0.1508*	0.1672*
.GOV2	0.1060	0.0611*	0.1264	0.0438*	0.1048*

In addition to our above described observations, we also find that the use of the terms in the title topic field in the first-pass retrieval (i.e. query setting TRQ) provides similar parameter sensitivity values compared to when using all query terms in the first-pass retrieval (i.e. query setting RQ). This indicates that TRQ has a comparable effectiveness in reducing the parameter sensitivity with RQ. Moreover, Table 7 compares the retrieval performance of these two query settings. The MAP values in Table 7 are

**Table 7: MAP(RQ) and MAP(TRQ) are the mean average precision values obtained using query settings RQ and TRQ, respectively. diff. indicates the relative difference between MAP(RQ) and MAP(TRQ) in percentage. The p-values are given by Wilcoxon matched-pairs signed-ranks test. A p-value marked with a star indicates a statistically significant difference between MAP(RQ) and MAP(TRQ).**

Coll.	MAP(RQ)	MAP(TRQ)	diff. (%)	p-value
PL2				
disk1&2	0.3189	0.3194	$\approx 0$	0.5608
disk4&5	0.2968	0.3020	+1.75	0.04263*
WT2G	0.2984	0.3095	+3.72	0.1008
WT10G	0.2539	0.2553	$\approx 0$	0.8717
.GOV2	0.3323	0.3386	+1.90	0.03033*
BM25				
disk1&2	0.3172	0.3191	$\approx 0$	0.8762
disk4&5	0.3005	0.3022	$\approx 0$	0.7256
WT2G	0.3125	0.3227	+3.26	0.2628
WT10G	0.2512	0.2562	+1.99	0.08292
.GOV2	0.3276	0.3374	+2.99	2.948e-03*

all maximised through the optimisation process described in Section 4. From Table 7, we find that although TRQ and RQ provide similar retrieval performance, TRQ results in a higher MAP than RQ in all 10 cases (see the MAP(RQ) and MAP(TRQ) values in Table 7). In particular, the difference between MAP(RQ) and MAP(TRQ) is statistically significant in 3 out of 10 cases. This suggests that it is more robust to use few most informative query terms (TRQ), than using all query terms (RQ), in the first-pass retrieval.

## 6. CONCLUSIONS AND FUTURE WORK

In this paper, we have studied the parameter sensitivity issue in the context of the probabilistic model for ad-hoc retrieval. Two popular probabilistic weighting models, namely BM25 and PL2, are included in our study. We have provided a better understanding and explanation for the parameter sensitivity issue. The main argument of this paper is that parameter sensitivity is caused by the existence of non-informative terms in the query. In order to avoid the dominance of non-informative query terms in the document ranking, a harsh normalisation is required, which can cause a high parameter sensitivity. However, by differentiating informative query terms from non-informative ones through a query term reweighing process, the parameter sensitivity can be reduced. In our experiments on five TREC test collections, we have shown that query term reweighing successfully achieves a remarkably reduced parameter sensitivity in most cases.

More specifically, the effect of query term reweighing on reducing parameter sensitivity has been evaluated by a cross-collection training process, and by using the Entropy and Spread measures, following [18]. The experimental results, by cross-collection training, show that query term reweighing allows the parameter setting optimised on one collection to be reused on another collection. Moreover, the evaluation using the Entropy and Spread measures suggest that for short queries, query term reweighing provides remarkably reduced Spread. This indicates an increased flatness of the retrieval performance distribution. For long queries,

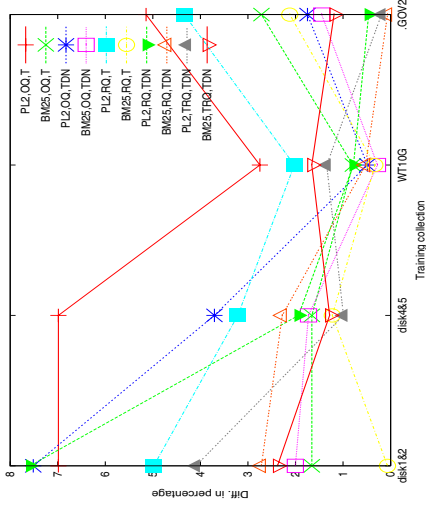
query term reweighing reduces both Spread and Entropy, indicating a remarkably reduced parameter sensitivity. In addition, the experimental results show that for long queries, it is more effective to use a few most informative query terms, instead of all query terms, in the first-pass retrieval of a query term reweighing process. However, the use of TRQ requires the query to have a shorter form (title-only query) and a longer form (full query), which is usually difficult in practise. Therefore, it will be helpful to devise an automatic method for selecting the most informative query terms, which is in our future research plan.

The study in this paper focuses on the BM25 and PL2 weighting models for ad-hoc retrieval. In the future, we plan to extend our study to other IR models and retrieval tasks. For example, we will study the parameter sensitivity in the language modelling approach [21, 26, 27]. The smoothing technique for language modelling, e.g. the Dirichlet Priors, has been shown to have a similar functionality with *tf* normalisation in dealing with the relationship between term frequency and document length [1, 14]. Therefore, we believe that our approach in this paper is applicable to the smoothing technique for language modelling.

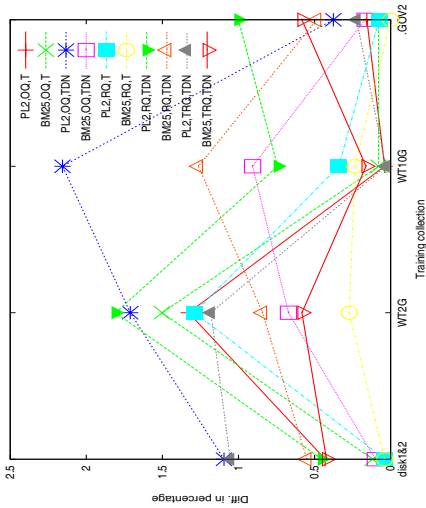
## 7. REFERENCES

- [1] G. Amati. *Probabilistic models for information retrieval based on Divergence from Randomness*. PhD thesis, Department of Computing Science, University of Glasgow, 2003.
- [2] G. Amati, C. Carpineto, and G. Romano. Fondazione Ugo Bordoni at TREC 2003: Robust and Web Track. In *Proceedings of TREC 2003*, Gaithersburg, MD, 2003.
- [3] G. Amati and C. J. Van Rijsbergen. Probabilistic models of information retrieval based on measuring the Divergence from Randomness. In *ACM Transactions on Information Systems (TOIS)*, volume 20(4), 2002.
- [4] C. Buckley, G. Salton, J. Allan, and A. Singhal. Automatic query expansion using SMART. In *Proceedings of the TREC-3*, Gaithersburg, MD, 1995.
- [5] A. Chowdhury, M. C. McCabe, D. Grossman, and O. Frieder. Document normalization revisited. In *Proceedings of SIGIR'02*, Tampere, Finland, 2002.
- [6] C. Clarke, N. Craswell, and I. Soboroff. Overview of the TREC-2004 Terabyte Track. In *Proceedings of TREC 2004*, Gaithersburg, MD, 2004.
- [7] C. Clarke, F. Scholer, and I. Soboroff. Overview of the TREC 2005 terabyte track. In *Proceedings of TREC 2005*, Gaithersburg, MD, 2004.
- [8] H. Fang, T. Tao, and C. Zhai. A formal study of information retrieval heuristics. In *Proceedings of SIGIR'04*, Sheffield, United Kingdom, 2004.
- [9] H. Fang and C. Zhai. An exploration of axiomatic approaches to information retrieval. In *Proceedings of SIGIR'05*, Salvador, Brazil, 2005.
- [10] S. Harter. *A probabilistic approach to automatic keyword indexing*. PhD thesis, The University of Chicago, 1974.
- [11] D. Hawking. Overview of the TREC-9 Web Track. In *Proceedings of the TREC-9*, Gaithersburg, MD, 2000.
- [12] D. Hawking, E. Voorhees, N. Craswell, and P. Bailey. Overview of the TREC-8 Web Track. In *Proceedings of TREC 8*, Gaithersburg, MD, 1999.
- [13] B. He and I. Ounis. A study of parameter tuning for term frequency normalization. In *Proceedings of CIKM'03*, New Orleans, LA, 2003.
- [14] B. He and I. Ounis. A study of the Dirichlet Priors for term frequency normalisation. In *Proceedings of SIGIR'05*, Salvador, Brazil, 2005.
- [15] B. He and I. Ounis. Term frequency normalisation tuning for BM25 and DFR model. In *Proceedings of ECIR'05*, Santiago de Compostela, Spain, 2005.
- [16] B. Jansen and A. Spink. How are we searching the World Wide Web? A comparison of nine search engine transaction logs. *Information Processing & Management*, 42(1):248–263, 2006.
- [17] S. Kirkpatrick, C. Gelatt, and M. Vecchi. Optimization by simulated annealing. *Science*, 220(4598), 1883.
- [18] D. Metzler. Estimation, sensitivity, and generalization in parameterized retrieval models. In *Proceedings of CIKM'06*, Arlington, VA, 2006.
- [19] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, and C. Lioma. Terrier: A high performance and scalable judgements retrieval platform. In *Proceedings of the OSIR Workshop 2006*, 2006.
- [20] V. Plachouras, B. He, and I. Ounis. University of Glasgow at TREC 2004: Experiments in Web, Robust, and Terabyte Tracks with Terrier. In *Proceedings of TREC 2004*, Gaithersburg, MD, 2004.
- [21] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *Proceedings of SIGIR'98*, Melbourne, Australia, 1998.
- [22] S.E. Robertson, S. Walker, and M. Beaulieu. Okapi at TREC-7: automatic ad hoc, filtering, VLC and interactive. In *Proceedings of TREC 7*, Gaithersburg, MD, 1998.
- [23] G. Salton. *The SMART Retrieval System*. Prentice Hall, New Jersey, 1971.
- [24] A. Singhal, C. Buckley, and M. Mitra. Pivoted document length normalization. In *Proceedings of SIGIR'96*, Zurich, Switzerland, 1996.
- [25] E. Voorhees. *TREC: Experiment and Evaluation in Information Retrieval*. The MIT Press, 2005.
- [26] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of SIGIR'01*, New Orleans, LA, 2001.
- [27] C. Zhai and J. Lafferty. Two-stage language models for information retrieval. In *Proceedings of SIGIR'02*, Tampere, Finland, 2002.

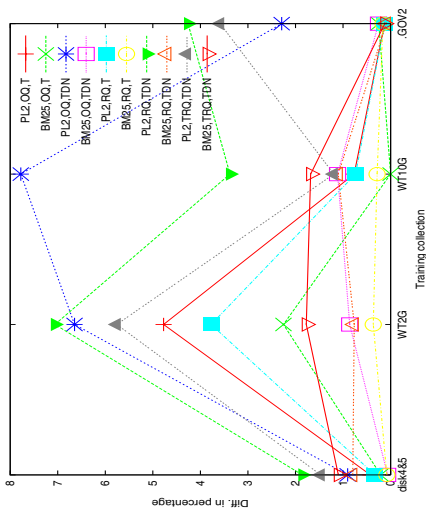




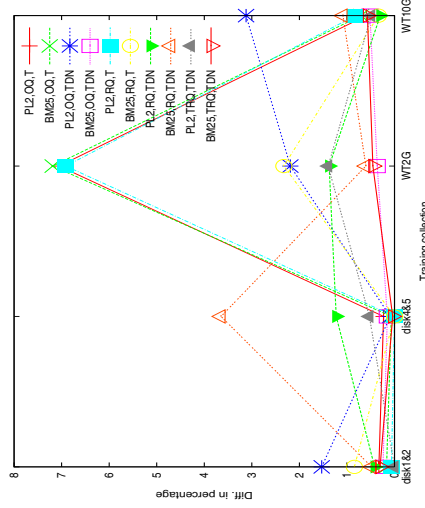
(a) disk1&2



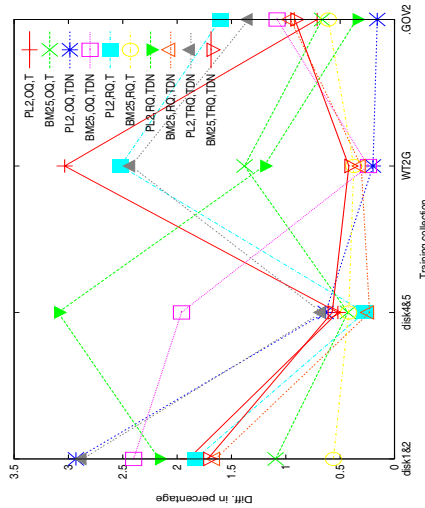
(b) disk4&5



(c) WT2G

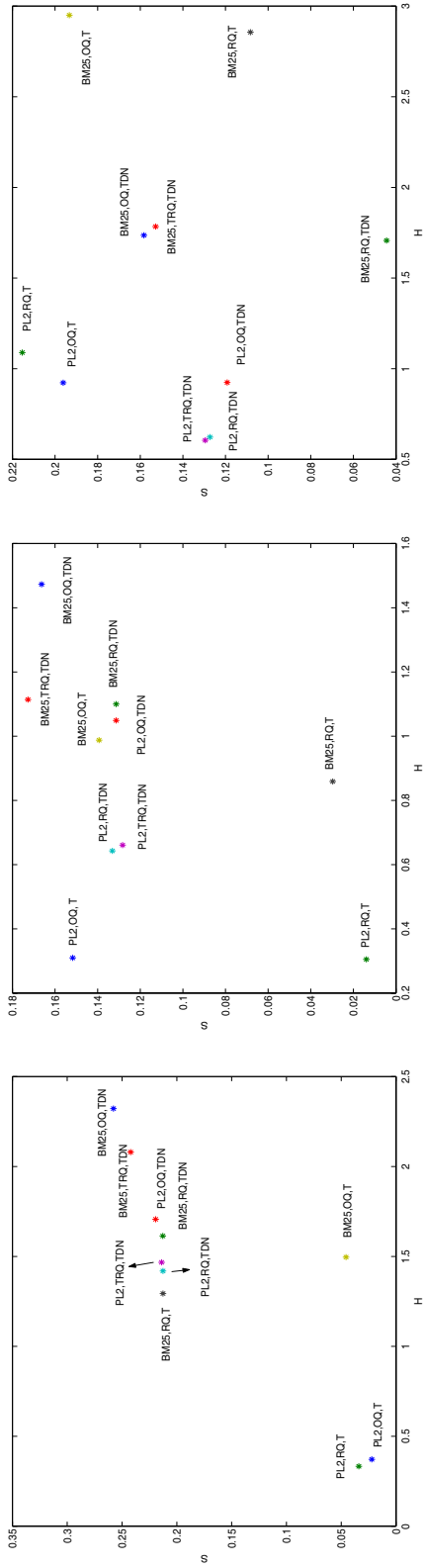


(d) WT10G

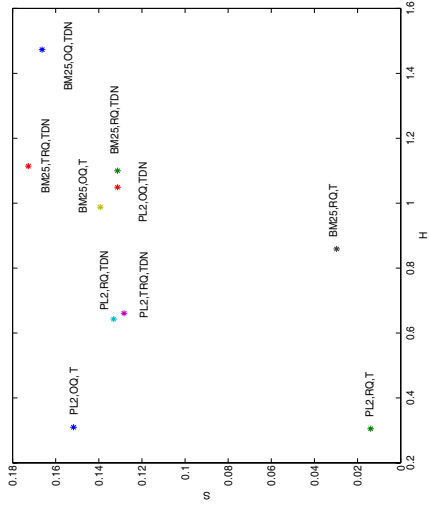


(e) .GOV2

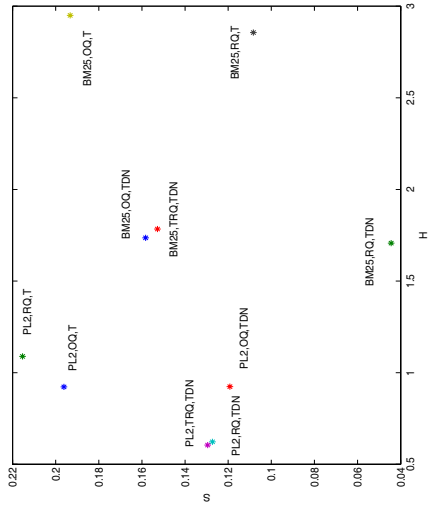
Figure 1: The results of the cross-training experiments.



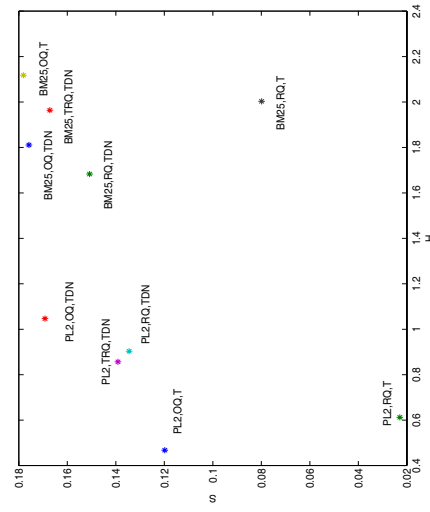
(a) disk1&2



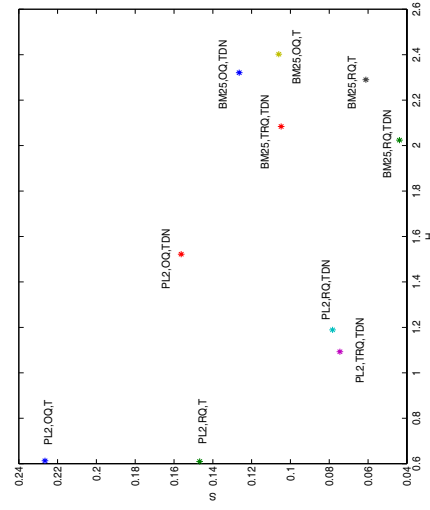
(b) disk4&5



(c) WT2G



(d) WT10G



(e) .GOV2

Figure 2: The Entropy (H) - Spread (S) plots on the five TREC collections used.