Artif Intell Rev (2007) 28:51–68
DOI 10.1007/s10462-008-9074-5

# An axiomatic comparison of learned term-weighting schemes in information retrieval: clarifications and extensions

**Ronan Cummins · Colm O'Riordan**

Published online: 13 September 2008


**Abstract** Machine learning approaches to information retrieval are becoming increasingly widespread. In this paper, we present term-weighting functions reported in the literature that were developed by four separate approaches using genetic programming. Recently, a number of axioms (constraints), from which all *good* term-weighting schemes should be deduced, have been developed and shown to be theoretically and empirically sound. We introduce a new axiom and empirically validate it by modifying the standard *BM*25 scheme. Furthermore, we analyse the *BM*25 scheme and the four learned schemes presented to determine if the schemes are consistent with the axioms. We find that one learned term-weighting approach is consistent with more axioms than any of the other schemes. An empirical evaluation of the schemes on various test collections and query lengths shows that the scheme that is consistent with more of the axioms outperforms the other schemes.

**Keywords** Information retrieval · Genetic programming · Axiomatic constraints

## 1 Introduction

Term-weighting is crucial to the problem of document ranking within most, if not all, information retrieval (IR) systems. Many approaches to term-weighting have been developed over the years. These schemes have been produced from various types of models, ranging from empirical learning models to purely theoretical models. With the increase in computing resources and the advances in machine learning techniques, many attempts have been made to learn term-weighting schemes assuming a *bag of words* approach. In particular, genetic programming (GP) is becoming popular due to the freedom offered in the definition of the problem and representation of the possible solutions. The basis behind many of these approaches is

R. Cummins (✉) · C. O'Riordan
Department of Information Technology, National University of Ireland, Galway, Ireland
e-mail: ronan.cummins@nuigalway.ie

C. O'Riordan
e-mail: colm.oriordan@nuigalway.ie

that useful features and properties within a population of solutions survive and propagate. GP produces a symbolic representation of the solution that is useful for analysis and generalisation. There have been a number of attempts using GP to evolve term-weighting schemes for ad hoc retrieval (Oren 2002b; Fan et al. 2004; Trotman 2005; Cummins and O'Riordan 2006). Although there has been much research into learning weights for terms in IR, many of these approaches do not produce term-weighting schemes.

An axiomatic approach to IR (Fang and Zhai 2005) has been developed which refines a number of constraints (axioms) (Fang et al. 2004) to which all *good* weighting functions should adhere. This approach and, in particular, the constraints are useful in attempting to theoretically motivate term-weighting functions that are developed from purely automated learning (empirically-based) models. More precisely, we believe that the best functions produced from a valid learning approach to IR should adhere to these existing constraints. The satisfaction of the constraints serve as a useful guide to the optimality of the solutions produced. These constraints can potentially be used in a number of different ways. They could be used to constrain the search space so that the adopted learning approach searches a much smaller space. In such a case, the learning approach could search a space in which good solutions are already known to exist.

They can also be used to validate a specific learning paradigm or learning framework in an IR domain by showing that the solutions (term-weighting schemes) produced adhere to them (i.e. that the learning approach adopted navigates the search space effectively and finds the area of the space in which good term-weighting schemes lie). This is the how the constraints are used in this work. This paper presents an analysis, using the existing axiomatic framework for IR, of term-weighting schemes learned using four different GP approaches. Firstly, we introduce the existing axioms and postulate a new axiom that complements the existing ones. We present one benchmark term-weighting scheme ($BM25$) and four learned term-weighting schemes produced from separate GP approaches to the problem. All of the learned term-weighting schemes here have been developed using a GP process in a *bag of words* retrieval framework. We empirically validate the new axiom and furthermore, we show that the scheme that satisfies the most axioms outperforms the others over multiple collections for various types of query without the need for tuning parameters.

The remainder of the paper is organised as follows: Section 2 introduces the existing axioms and motivates a further axiom. Section 3 presents five term-weighting approaches together with a brief analysis of each scheme. In Sect. 4, we present results that provide empirical evidence for the validity of the axioms. Finally, our conclusions are outlined in Sect. 5. This paper extends previous work (Cummins and O'Riordan 2007a) by including a separate empirical validation of the new axiom. Furthermore, a weaker version of one of the axioms is included as it is shown that, contrary to previous work (Fang and Zhai 2005, Cummins and O'Riordan 2007a), two axioms are not unconditionally adhered to by *any* efficient modern term-weighting schemes. A discussion of, and resolution to, this issue is included as Appendix.

## 2 Axioms for term-weighting

### 2.1 Term-weighting anatomy

High performance term-weighting schemes are typically comprised of three components. These include a *term-discrimination* aspect, which is the basic weight assigned to a

term ($w(t)$), a *term-frequency* aspect ($tff()$) and a *normalisation* aspect ($n()$). The term-discrimination aspect (usually some type of $idf$) promotes terms which are more likely to convey the meaning of the document (i.e. terms that significantly contribute to the topic of the document). The term-frequency aspect aims to promote documents with more occurrences of these useful terms. Finally, the normalisation aspect aims to promote shorter documents with high occurrences of query terms. In a term-weighting scheme, each of these aspects can be realised using different features and characteristics of the terms, the documents and the collection. There are also many ways in which the interplay between the three aspects of a term-weighting scheme can differ. Ensuring that a term-weighting scheme contains these three aspects does *not* guarantee that the scheme will adhere to any of the constraints outlined in the next section, as these aspects may be realised using different features. Neither does it guarantee a high performance term-weighting scheme. It is typically the interplay between these three aspects that is constrained in some way by the axioms.

### 2.2 Axioms

We will briefly introduce the previously developed constraints using an inductive framework (Fang and Zhai 2005). The idea of this inductive framework is to define a base case function that describes the score (weight) assigned to a document containing a single term matching (or not matching) a query containing a single term. All other cases can be dealt with inductively using two separate functions. A *document growth function* describes the change in the score when a single term is added to the document, while a *query growth function* describes the change in the score when a single term is added to the query. This is an elegant approach to formalising necessary characteristics of *good* term-weighting functions. It is also possibly a more realistic approach to the human determination of relevance, as human readers will tend to update their relevance judgments as they read through documents.

A number of axioms have been postulated and these can be used to validate or to develop term-weighting schemes in a constrained space. Thus we use the term axiom and constraint analogously in this paper. Assume $S(Q, D)$ is a function which scores a document $D$ in relation to a query $Q$ in a standard *bag of words* retrieval model. With notation similar in style to the original work (Fang and Zhai 2005), the constraints can be formalised as follows, where $t \in T$ is a term $t$ in the set of terms in a corpus and $\delta_t(t, D, Q) = S(Q, D \cup \{t\}) - S(Q, D)$ (i.e. the change in score as $t$ is added to the document $D$):

**Constraint 1** $\forall Q, D$ and $t \in T$, if $t \in Q$, $S(Q, D \cup \{t\}) > S(Q, D)$
Constraint 1 states that adding a new query term to the document must *always* increase the score of that document. This captures the basic behaviour of a term-frequency aspect. This constraint ensures that the basic weight of a term (an $idf$ type measure) must be positive and that any penalisation due to the document becoming longer (normalisation) must be less than the increase in score due to the term being added.

**Constraint 2** $\forall Q, D$ and $t \in T$, if $t \notin Q$, $S(Q, D \cup \{t\}) < S(Q, D)$
Constraint 2 states that adding a non-query term to a document must *always* decrease the score of that document. This constraint ensures that some sort of normalisation is present and specifies its basic operating principle.

**Constraint 3** $\forall Q, D$ and $t \in T$, if $t \in Q$, $\delta_t(t, D, Q) > \delta_t(t, D \cup \{t\}, Q)$
Constraint 3 states that adding successive query terms to a document should increase the score of the document less with each successive addition. Essentially, the term-frequency

influence must be sub-linear. The intuition behind this constraint is that it is ultimately the first occurrence of a term that indicates that the document is on-topic (i.e. related to the query). Due to characteristics of natural language usage, it is known that when a term first appears in a document, the likelihood of re-appearance increases. Thus, the weight given to successive occurrences of a query term should be reduced.

#### *2.2.1 Weaker version of Constraint 1*

A weaker constraint can be deduced from the first two constraints and can be described as follows:

**Constraint 1.1** $\forall Q, D$ and $t_1 \in T$, $t_2 \in T$, if $t_1 \in Q$ and $t_2 \notin Q$, $S(Q, D \cup \{t_1\}) > S(Q, D \cup \{t_2\})$
Constraint 1.1 states that adding a new query term to the document should result in a higher score for the document than adding a non-query term. It is true that if Constraints 1 and 2 are satisfied then Constraint 1.1 is satisfied accordingly. This constraint has previously been introduced (Fang et al. 2004) and is included because Constraints 1 and 3 are *not* unconditionally satisfied by any efficient modern term-weighting scheme.[1] This is because the normalisation used in term-weighting schemes penalises all of the existing terms in a document. Therefore, a query term with a low discrimination value ($idf$) that is added to the document, may not increase the score of the document sufficiently to offset the penalisation due to the document increasing in length. Therefore, Constraint 1 will be deemed satisfied if the score, attributable from the term, and not the document as a whole, increases for every occurrence of that term. Similarly, Constraint 3 will be deemed satisfied if the score increase, attributable from a particular term, grows sub-linearly.

These constraints are used to check the validity of term-weighting schemes before evaluation. Furthermore, term-weighting schemes which adhere to these constraints are shown empirically to outperform weighting schemes that fail to adhere to one or more of the constraints (Fang et al. 2004; Fang and Zhai 2005). The constraints are also useful in defining valid bounds on tuning parameters that appear in many existing term-weighting schemes. It should be noted that simply adhering to these constraints does not guarantee a high performance weighting scheme. Rather it is the violation of one or more of the constraints that indicates the performance is non-optimal (i.e. breaks some rule of the proposed model of relevance). It is worth noting that these axioms typically constrain a term-weighting scheme's *within-document* features (i.e. its term-frequency aspect and normalisation aspect), although they do enforce some limitations on the type of term-discrimination aspect used.

### 2.3 New axiom

We now propose a new constraint which aims to avoid over-penalising successive occurrence of non-query terms in documents.

**Constraint 4** $\forall Q, D$ and $t \in T$, if $t \notin Q$, $|\delta_t(t, D, Q)|^{-1} > |\delta_t(t, D \cup \{t\}, Q)|^{-1}$.
According to Heaps' law (1978), the appearance of new, previously unseen terms in a corpus grows in roughly a square-root relationship (sub-linearly) to the document length (in words). Therefore, as *non-query terms* appear in a document they should be penalised less

[1] See Appendix for a discussion and a solution that unconditionally satisfies the constraints.

with successive occurrences. This constraint avoids over-penalising longer documents by ensuring that the normalisation aspect is sub-linear. For example, consider a document that has 9 words ($dl = 9$) and contains 3 unique terms (i.e. vector length of 3). If this document grows in length to 100 words ($dl = 100$), the expected number of unique terms would be approximately 10. Thus, as the document grows in length, the topic broadens sub-linearly. Furthermore, it is the number of occurrences (term-frequency) of these unique terms that indicates the strength of each different aspect (i.e. dimension of the vector) of the topic.

Essentially, the inverse of the score reduction due to non-query terms being added (an increasing value) should be sub-linear. This follows intuitively from Constraint 3 which controls how the score of a document changes as successive query terms are added to a document. Therefore, it is the first appearance of a non-query term that ultimately indicates a change in the topic of a document and successive occurrences of this term do not indicate that the topic of that document is drifting from the query to the same degree.

### 2.4 Summary

The following are some useful properties of term-weighting schemes that can be deduced from the axioms:

- A term-weighting scheme *must* contain a positively-increasing term-frequency component.
- The basic weight of a term (i.e. some type of $idf$) must *always* be positive.
- The term-frequency aspect *must* be sub-linear (as more of the *same* query terms occur, the increase in score must be smaller).
- A scheme *must* contain a normalisation aspect (some method of penalising longer documents).
- The normalisation aspect must be sub-linear (as more non-query terms occur, the increase in penalisation must be smaller).
- The penalisation of document score, due to the document increasing in length (normalisation), must be less than the increase in score when a query term is added.

## 3 Term-weighting scheme analysis

In this section, one benchmark and four learned term-weighting schemes are introduced and briefly analysed with respect to the previously outlined axioms. Table 1 summarises the results from this section.

**Table 1** Constraints satisfaction

| Rank | Scheme | Constraints | | | | |
|---|---|---|---|---|---|---|
| | | 1.1 | 1 | 2 | 3 | 4 |
| 1 | $F4(Q, D)$ | Yes | Yes* | Yes | Yes* | Yes |
| 2 | $BM25(Q, D)$ | Cond. | Cond. | Yes | Cond. | No |
| 3 | $F1(Q, D)$ | Cond. | Cond. | Yes | Cond. | No |
| 4 | $F3(Q, D)$ | Cond. | Cond. | Cond. | Cond. | No |
| 5 | $F2(Q, D)$ | Yes | Cond. | No | Cond. | No |

### 3.1 *BM*25

The *BM*25 weighting scheme, developed by Robertson et al. (1995), is a weighting scheme based on the probabilistic model. The score of a document $D$ in relation to a given query $Q$ can be calculated as follows:

$$BM25(Q, D) = \sum_{t \in Q \cap D} \left( \frac{tf_t^D}{tf_t^D + k_1 \cdot \left( (1-b) + b \cdot \frac{dl}{dl_{\text{avg}}} \right)} \cdot \log \left( \frac{N - df_t + 0.5}{df_t + 0.5} \right) \cdot tf_t^Q \right) \quad (1)$$

where $tf_t^D$ is the frequency of a term $t$ in $D$ and $tf_t^Q$ is the frequency of the term in the query $Q$. $dl$ and $dl_{\text{avg}}$ are the length and average length of the documents respectively measured in non-unique terms. $N$ is the number of documents in the collection and $df_t$ is the number of documents in which term $t$ appears. $k_1$ is the term-frequency influence parameter which is set to 1.2 by default. The query term weighting used here ($tf_t^Q$) is slightly different to the original weighting method proposed (Robertson et al. 1995) but has been used successfully in many studies (Fang and Zhai 2005). $b$ is the document normalisation influence parameter and has a default value of 0.75.

#### *3.1.1 Analysis*

The term-frequency aspect is sub-linear and the normalisation used can never lead to a decrease in weight (attributable from that term) when a query term is added. However, it can be noted that the $idf$ component in the *BM*25 ($\log(\frac{N-df_t+0.5}{df_t+0.5})$) function will return a negative value when $df_t > \frac{N}{2}$ and thus violates Constraints 1, 1.1 and 3 when stop-word removal is not used (as very frequent terms remain in the system). Constraint 2 is satisfied unconditionally as the addition of a non-query term always decreases the score of the document. It can also be seen that the normalisation function used in this function is linear. This suggests that it needs to be tuned on specific collections as it may over-penalise long documents on certain collections. Thus, Constraint 4 is also violated.

### 3.2 Oren

One of the first approaches to evolve term-weighting schemes (Oren 2002a,b) uses non-atomic features of the terms, documents, queries and the collection to evolve term-weighting functions for use in IR. Using parts of existing functions as terminals in the GP can be viewed as a type of seeding or biasing, as prior knowledge about what constitutes a good ranking function is used. This type of approach can arbitrarily limit the search space. This work uses a small document collection (1,239 documents) and 70 queries to evolve functions using a population of 100 individuals run for 150 generations. It has been noted by the author that they believe that many of the functions are non-generalisable. One of the schemes outlined in this work can be re-written as follows:

$$F1(Q, D) = \sum_{t \in Q \cap D} \left( \frac{tf_t^D}{tf_t^D + df_t + dl \cdot (1 + 0.436 \cdot \frac{tf_t^D}{tf_{\max}^D} \cdot (cf_{\max} + \log(cf_{\max})))} \right) \quad (2)$$

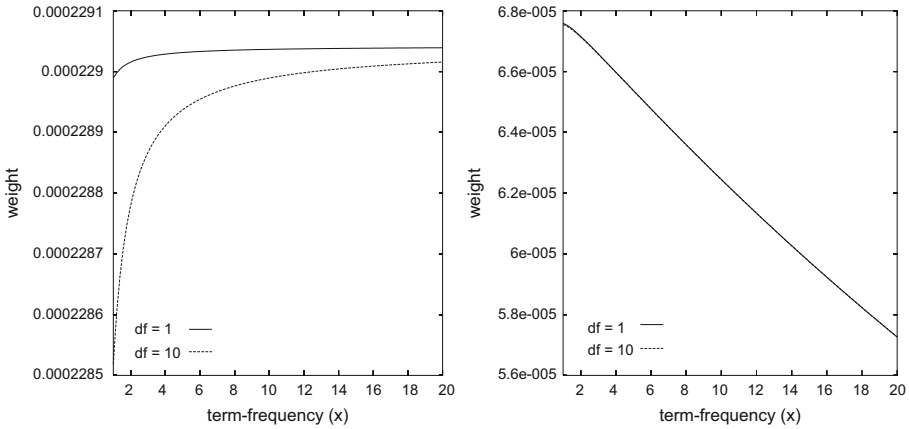


**Fig. 1** Change in weight for $F1$ for scenarios 1 and 2 respectively

where $tf^D_{\max}$ is the frequency of the most common term in $D$ and $cf_{\max}$ is the frequency of the most common term in the collection.

### *3.2.1 Analysis*

This function can be written as $\frac{tf_t^D}{tf_t^D + df_t + dl \cdot (1 + \frac{tf_t^D}{tf^D_{\max}} \cdot K_1)}$ where $K_1$ is a constant such that $K_1 > 0$. Consider the simple case where a document consists of multiple occurrences of the same query-term (scenario 1). In such a circumstance, let $x = tf_t^D = dl = tf^D_{\max}$. The function can then be re-written as $F1 = \frac{x}{x + df_t + x \cdot (1 + K_1)}$ where $df_t$ is a constant for a particular term. Figure 1 shows the change in score for the first 20 occurrences of the query term in this scenario. An extra occurrence of a query term will make the score of the document higher and the term-frequency aspect will grow sub-linearly. In these circumstances, Constraints 1 and 3 are satisfied.

Now, consider a document with 100 terms ($dl = 100$) and a maximum term-frequency of 30 for one of the terms ($tf^D_{\max} = 30$) already occurring in the document (scenario 2). The function can then be written as $F1 = \frac{x}{x + df_t + (100 + x) \cdot (1 + \frac{x}{30} \cdot K_1)}$. Figure 1 also shows the change in weight for the first 20 occurrences of another query term added to the document in this scenario. It can be seen that in this circumstance that the score of the document (attributable from that query term) actually decreases. This violates Constraints 1 and 3. Constraint 2 is satisfied as the score of a document will always decrease as non-query terms are added due to $dl$ in the denominator. It can be shown that Constraint 1.1 is also satisfied. However, it can be determined that the normalisation component is linear in nature (i.e. the normalisation component $dl$ is linear in the denominator) violating Constraint 4. As a result this function is likely to perform poorly overall, and possibly even worse for long queries where the weights of a number of different query terms will interact incorrectly.

## 3.3 Fan et al.

Another approach to evolving weighting schemes (Fan et al. 2004) which assumes a simplistic query term weighting (i.e. $tf_t^Q$) has also been attempted. In this research, a term-weighting

function was learned using short queries and a population of 200 individuals for 30 generations. The best function outlined therein can be re-written as follows:

$$F2(Q, D) = \sum_{t \in Q \cap D} \left( \frac{\log(tf_t^D \cdot EXP)}{vl + 2 \cdot tf_{\max}^D + 0.373} \right) \cdot tf_t^Q \tag{3}$$

where

$$EXP = \left( tf_{\text{avg}}^D + \frac{tf_t^D}{\log\left(tf_t^D \cdot 2 \cdot tf_{\text{avg}}^D\right)} + \frac{tf_t^D \cdot N \cdot tf_{\text{avg}}^D \cdot \left(tf_{\max}^D + vl\right)}{df_t^2} \right) \tag{4}$$

where $tf_{\text{avg}}^D$ is the average term-frequency in $D$ and $vl$ is the length of the document vector (unique terms).

### *3.3.1 Analysis*

Again, consider the simple case where a document consists of multiple occurrences of the same term (scenario 1). In such a circumstance, let $x = tf_t^D = dl = tf_{\max}^D = tf_{\text{avg}}^D$ and $vl = 1$. The weight of the document can be written as:

$$F2 = \log\left( x \cdot \left( x + \frac{x}{\log(x \cdot 2 \cdot x)} + \frac{x \cdot N \cdot x \cdot (x+1)}{df_t^2} \right) \right) \Big/ (1 + 2 \cdot x + 0.373)$$

Figure 2 shows the score change when $N$=100,000 and when $df_t = 1$ and when $df_t = 10$. It can be seen that Constraints 1 and 3 will be violated in this simplistic case. Now consider the more common case where a document already contains a number of terms (e.g. $dl = 100$, $tf_{\max}^D = 30$ and $vl = 4$). Figure 2 shows the score change (attributable from a particular query term) when it is added to this type of document. The score change can be re-written as follows for a particular term:

$$F2 = \log\left( x \cdot \left( \frac{100 + x}{5} + \frac{x}{\log\left(x \cdot 2 \cdot \frac{100+x}{5}\right)} + \frac{x \cdot N \cdot \frac{100+x}{5} \cdot (30+5)}{df_t^2} \right) \right) \Big/ (5 + 2 \cdot 30 + 0.373)$$

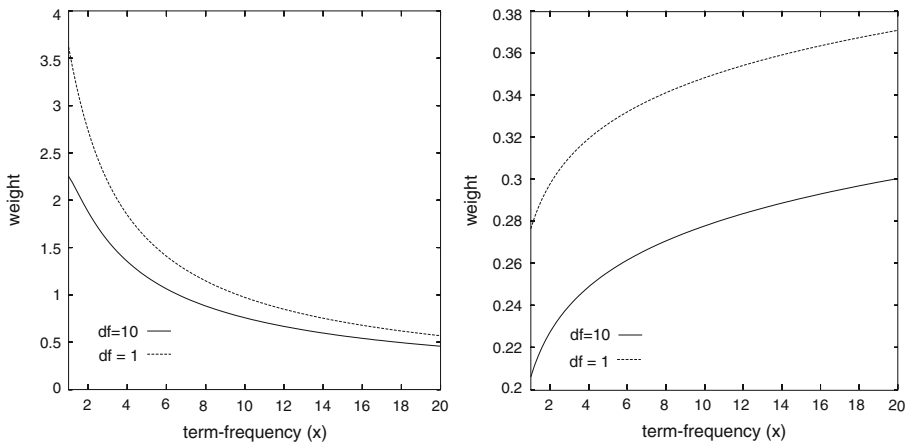


**Fig. 2** Change in weight for $F3$ for scenarios 1 and 2 respectively

In this second scenario, a new occurrence of a query term results in a higher score for the document and the term-frequency is sub-linear. As a result, Constraints 1 and 3 are conditionally satisfied. As the normalisation used is the number of unique terms (vector length), Constraints 2 and 4 are violated. If a non-query term which has already appeared in the document re-occurs, the weight of the document will not decrease as the vector length remains unchanged. Even if the document length factor used was changed to the document length (i.e. $dl$), Constraint 4 would still be violated, as the normalisation factor is linear in nature.

## 3.4 Trotman

An approach using primitive atomic features of terms, documents, queries and the document collection has also been attempted (Trotman 2005). This approach uses a large terminal and function set with little or no constraints on the search space. In this approach a seeded population of 100 individuals (96 random and 4 existing functions) was run for 100 generations 13 times. One of the best performing schemes (named run 5, Trotman 2005) can be re-written as follows:

$$F3(Q, D) = \sum_{t \in Q \cap D} \left( \log_2 \left| \frac{N - \log_2 |N|}{2 \cdot df_t} \right| \cdot \frac{cf_t}{df_t} \cdot \frac{tf_t^D \cdot tf_t^Q \cdot cf_{\max}}{\max\left(C_3, C_4 + \frac{C_1 \cdot (\log |C_2 + tf_t^Q| + cf_t) \cdot dl}{N \cdot dl_{\text{avg}}}\right) + tf_t^D} \right) \tag{5}$$

where $cf_t$ is the frequency of $t$ in the entire collection of $N$ documents. $C_1$, $C_2$, $C_3$ and $C_4$ are constants with values 33.40102, 23.94623, 1.2 and 0.25, respectively.

### *3.4.1 Analysis*

Firstly, the $\log_2 |\frac{N - \log_2 |N|}{2 \cdot df_t}|$ part can lead to a negative weight for terms with a high document frequency. This leads to Constraints 1, 1.1 and 3 being violated in circumstances similar to those of the *BM*25 scheme. We can see that when $C_3 > C_4 + \frac{C_1 \cdot (\log |C_2 + tf_t^Q| + cf_t) \cdot dl}{N \cdot dl_{\text{avg}}}$, which typically occurs when $cf_t$ is low (i.e. for rare terms), the within document part of the formula reduces to $\frac{tf_t^D}{1.2 + tf_t^D}$ which is a primitive form of the *BM*25 local weighting. This form of the function has no normalisation component and thus violates Constraint 2 in these circumstances. However, this form conditionally satisfies Constraints 1 and 3 as adding a query term to a document always increases its score (Constraint 1) and is sub-linear (Constraint 3).

When $C_3 < C_4 + \frac{C_1 \cdot (\log |C_2 + tf_t^Q| + cf_t) \cdot dl}{N \cdot dl_{\text{avg}}}$ which typically occurs when the collection frequency for a term is high (i.e. more common terms), a different form of the function is used. Consider a typical case when $tf_t^Q$ is 1 and $N$ is large (e.g. 100,000). In such a case, this reduces to approximately $0.25 + \frac{3.2 + cf_t}{3000} \cdot \frac{dl}{dl_{\text{avg}}}$. For high values of $cf_t$, this will exceed $C_3$ (i.e. 1.2). Interestingly, this can be re-written as $\frac{tf_t^D}{0.25 + K_2 \cdot \frac{dl}{dl_{\text{avg}}} + tf_t^D}$ (where $K_2$ is a global constant for a particular term) which contains a normalisation form similar to the *BM*25 scheme. When the function takes this form, Constraints 1, 1.1, 2 and 3 are satisfied. However, the normalisation scheme is not sub-linear and thus does not satisfy the new constraint (Constraint 4).

### 3.5 Cummins–O'Riordan

In this approach, a three-stage incremental approach was used to develop an entire term-weighting scheme by evolving, in turn, three constituent parts of a function (Cummins and O'Riordan 2007b). The term-discrimination (or global) part was developed using a population of 100 for 50 generations, while the term-frequency and normalisation parts were developed using a population of 200 for 25 generations each. The following is a typical entire term-weighting scheme:

$$F4(Q, D) = \sum_{t \in Q \cap D} \left( \frac{ntf}{ntf + 0.45} \cdot \sqrt{\frac{cf_t^3 \cdot N}{df_t^4}} \cdot tf_t^Q \right) \tag{6}$$

The term-frequency influence factor here (i.e. $\frac{ntf}{ntf+0.45}$) has been modelled to reflect the effect of an evolved term-frequency influence function by measuring the relative term-frequency (Cummins and O'Riordan 2005). The normalisation aspect was evolved on three different collections (indexed separately) with varying document length characteristics as most normalisation functions (including $BM25$) tend to be collection specific. One of the best normalisation functions found using this approach was $\sqrt{dl/dl_{\text{avg}}}$. Thus, the entire function can be recovered when $ntf$ is as follows:

$$ntf = \frac{tf_t^D}{\sqrt{dl/dl_{\text{avg}}}} \tag{7}$$

#### *3.5.1 Analysis*

It can be seen that the term-discrimination part of this function is always positive and the term-frequency aspect is sub-linear. The normalised term-frequency ($ntf$) is normalised as $tf_t^D/\sqrt{dl/dl_{\text{avg}}}$. Thus, if a query term is added both $tf_t^D$ and $dl$ will increase by 1, thereby increasing the value of the $ntf$ component. As a result, Constraints 1, 1.1 and 3 are deemed satisfied. If a non-query term is added, $ntf$ will decrease in value satisfying Constraint 2. As more non-query terms are added, the document penalisation grows sub-linearly. This leads to Constraint 4 being satisfied.

### 3.6 Summary of constraint satisfaction

Table 1 shows the constraints that each scheme satisfies. The conditional satisfaction (denoted "Cond.") means that the constraint is satisfied in many circumstance (as noted in the analysis) but is not unconditionally satisfied. Schemes that satisfy Constraints 1 and 3 (ignoring the typically small penalisation of previously existing terms in the document) are denoted as "Yes*" as they adhere to the constraints proposed in the original paper (Fang et al. 2004) but not to the slightly more general ones that are reformulated in the inductive framework (Fang and Zhai 2005).

We have ranked these schemes based on the number constraints they satisfy. We ranked $BM25$ and $F3$ ahead of $F2$ mainly due to the fact that $BM25$ and $F3$ will only violate Constraints 1 and 3 if stop-words are not removed. Stop-words are removed for the experiments outlined in this paper and indeed are typically removed for many IR systems. Thus, $F2$ will typically break constraints more often than $BM25$ or $F3$ will. It should be noted that this ranking is coarse as we do not know if violations of different constraints lead to equal levels

of suboptimality. We are also unsure if the schemes identified are specific to a type of query or indeed the specific environment in which they were trained. Nonetheless, given these details of constraint satisfaction, it seems an intuitive and possibly useful way of ordering the schemes by expected performance.

## 4 Empirical comparison

In this section we present experiments to empirically validate the newly introduced axiom and the previous analysis.

### 4.1 Document collections and evaluation

We used the LATIMES, FBIS and FT (years 1991–1993) collections from TREC disks 4 and 5 as test collections. Topics 401–450 were used for each of these collections. For each set of topics we create a set of short queries consisting of the title field of the topics, a set of medium length queries consisting of the title and description fields, and a set of long queries consisting of the title, description and narrative fields. We also use documents from the OHSUMED collection (years 1989 and 1990–1991) as two further test collections. We created short queries for the OHSUMED collections by simply removing terms from the description field of the 63 topics. Standard stop-words from the Brown Corpus[2] are removed and remaining words are stemmed using Porter's algorithm (1980). Furthermore, mean average precision (MAP) is used as the evaluation metric as it was directly used as the fitness function in all of the learning approaches mentioned herein. It is a widely used and stable measure of IR system performance (Buckley and Voorhees 2000).

### 4.2 Validation of new axiom

In this section, we validate the new axiom (Constraint 4) which states that the normalisation aspect should be sublinear. We use the $BM25$ scheme and find the optimal performance of the normalisation parameter $b$ for each collection. We manually tune the $BM25$ scheme using eight values of $b$ (0.125, 0.25, 0.375, 0.5, 0.625, 0.75, 0.825 and 1) which are within its acceptable range of values. We use the $BM25$ scheme for this evaluation as its normalisation scheme is linear for all values of $b$ and the normalisation part was developed in conjunction with the full scheme. The best ($BM25_{\text{opt}}$), worst ($BM25_{\text{low}}$) and average ($BM25_{\text{avg}}$) performance of the $BM25$ scheme for the eight values of $b$ are shown in Tables 2 and 3. We again use the $BM25$ scheme and incorporate the evolved normalisation ($\sqrt{dl/dl_{\text{avg}}}$) factor into the scheme. Thus, the evolved sub-linear normalisation ($\sqrt{dl/dl_{\text{avg}}}$) replaces the standard linear normalisation in the $BM25$ scheme ($(1-b)+b\cdot\frac{dl}{dl_{\text{avg}}}$). Although this normalisation was learned in a different framework, it is incorporated into the $BM25$ scheme. Normalisation schemes, and indeed other parts of schemes, have previously been analysed and substituted separately in many studies (Salton and Buckley 1988; He and Ounis 2003). Therefore, the *only* difference between the two schemes is that the modified $BM25$ scheme ($BM25_{\text{mod}}$) adheres to the new axiom (Constraint 4) and the original $BM25$ does not.

The modified $BM25$ scheme ($BM25_{\text{mod}}$) performs comparably to the optimal performance of the original $BM25$ ($BM25_{\text{opt}}$) scheme and sometimes surpasses it. The perfor-

[2] http://www.lextek.com/manuals/onix/stopwords1.html.

**Table 2** %MAP of $BM25$ and modified $BM25$ on three TREC collections

| #Docs | FT91–93 (401–450) (138668) | | | FBIS (401–450) (130471) | | | LATIMES (401–450) (131896) | | |
|---|---|---|---|---|---|---|---|---|---|
| | Short | Medium | Long | Short | Medium | Long | Short | Medium | Long |
| Scheme | | | | | | | | | |
| $BM25_{opt}$ | 32.97 | 35.74 | 36.22 | 31.35 | 32.00 | 30.09 | 25.92 | 26.73 | 29.63 |
| $BM25_{avg}$ | 32.00 | 35.07 | 34.17 | 28.22 | 29.50 | 28.87 | 24.47 | 25.46 | 27.08 |
| $BM25_{low}$ | 30.00 | 32.95 | 30.24 | 23.80 | 26.54 | 25.58 | 21.94 | 23.94 | 23.54 |
| $BM25_{mod}$ | 33.03 | 36.08 | 36.30 | 31.01 | 31.64 | 28.40 | 25.80 | 26.11 | 27.45 |

**Table 3** %MAP on OHSUMED collections

| #Docs | OH89 (1–63) (74869) | | OH90–91 (1–63) (148162) | |
|---|---|---|---|---|
| | Short | Medium | Short | Medium |
| Scheme | | | | |
| $BM25_{opt}$ | 28.13 | 31.23 | 25.71 | 28.08 |
| $BM25_{avg}$ | 26.99 | 27.28 | 24.55 | 27.03 |
| $BM25_{low}$ | 23.66 | 30.10 | 22.36 | 24.98 |
| $BM25_{mod}$ | 27.71 | 31.19 | 25.55 | 28.76 |

mance for the worst value of $b$ is significantly worse than the optimal setting of $b$ and consequently $BM25_{mod}$. The modified $BM25$ scheme consistently outperforms the average performance of the original $BM25$ scheme. The optimal $b$ parameter (used in the $BM25_{opt}$ scheme) varies considerably on the collections and queries used in these experiments. This free parameter varies on a per collection and per query basis as has been shown in many previous studies (Chowdhury et al. 2002; He and Ounis 2003). There have been many studies into more efficient and effective ways to tune these normalisation parameters (He and Ounis 2003, 2005), other than manually tuning them over a range of values. However, it is interesting that a *non-parametric* scheme that adheres to the new axiom performs comparably to the *optimal* performance of the linear normalisation. Furthermore, it is reasonable to assume that this parameter is needed in the original $BM25$ scheme because the linear function shape is not suitable for normalisation (and thus needs to be tuned depending on the data that is presented).

### 4.3 Empirical comparison of schemes

The results (Tables 4 and 5) show that $F4$ outperforms most of the other schemes on the various test data. The schemes tend to perform in accordance with the rank in Table 1. However, $F1$ is the exception to this rule. This scheme adheres to some of the constraints but performs poorly on many of the collections. This scheme was learned on a very small set of documents (1,239 documents) and it would appear that this is not a suitable size collection to learn generalisable term-weighting schemes. Indeed, it is suggested (Oren 2002b) that this formula is not likely to be generalisable. $F2$, which was trained using only short queries, is very specific

**Table 4** %MAP on three TREC collections

| #Docs | FT91–93 (401–450) (138668) | | | FBIS (401–450) (130471) | | | LATIMES (401–450) (131896) | | |
|---|---|---|---|---|---|---|---|---|---|
| | Short | Medium | Long | Short | Medium | Long | Short | Medium | Long |
| Scheme | | | | | | | | | |
| $F4$ | 32.72 | 36.30 | 38.46 | 31.49 | 31.45 | 31.30 | 26.50 | 26.87 | 28.80 |
| $BM25$ | 31.27 | 35.33 | 35.35 | 27.54 | 28.87 | 30.00 | 24.85 | 26.73 | 28.86 |
| $F1$ | 23.13 | 23.40 | 18.92 | 19.72 | 16.50 | 10.17 | 15.40 | 13.73 | 11.83 |
| $F3$ | 31.95 | 33.82 | 33.61 | 26.93 | 28.51 | 30.63 | 24.30 | 25.81 | 25.68 |
| $F2$ | 26.61 | 15.40 | 08.00 | 29.10 | 24.90 | 12.87 | 12.56 | 08.74 | 02.59 |

**Table 5** %MAP on OHSUMED collections

| #Docs | OH89 (1–63) (74869) | | OH90–91 (1–63) (148162) | |
|---|---|---|---|---|
| | Short | Medium | Short | Medium |
| Scheme | | | | |
| $F4$ | 27.56 | 32.80 | 25.53 | 30.07 |
| $BM25$ | 27.29 | 30.67 | 25.59 | 28.08 |
| $F1$ | 20.26 | 15.70 | 17.70 | 13.51 |
| $F3$ | 26.10 | 30.72 | 23.19 | 26.27 |
| $F2$ | 06.97 | 00.90 | 06.60 | 01.05 |

to these types of queries. It can be seen that the performance of $F2$ gets progressively worse as the query length increases (even though there is more evidence supplied as to the user's intention). $F1$, which was trained on a small amount of test data, behaves similarly. It is intuitive that as a user supplies more evidence about his information need (even if some of these terms are noisy) the performance for that query should tend to improve. This is be a desirable property in any good term-weighting scheme.

### 4.4 Discussion

The existing axioms and the newly postulated axiom are useful estimators of term-weighting optimality. They can be useful in estimating the performance of a scheme prior to evaluation. An interesting result is that many of the learned approaches conditionally adhere to some of the constraints. This would suggest that they did indeed learn useful methods for weighting terms in their training environment but that their training data is quite specific (i.e. the constraints are satisfied for the characteristics of the training data but are not unconditionally satisfied).

These results can tell us something about how to learn term-weighting functions. Small collections (less than 10,000 documents) should be avoided when aiming to learn *generalisable* term-weighting schemes. Indeed, it has already been shown that the term-discrimination (global) part of a term-weighting scheme can indeed be learned on a small collection but it is typically the within-document (local) part of these schemes that is not generalisable (Cummins and O'Riordan 2006). To overcome the collection dependence problem (which

typically affects the type of normalisation to use), it is advisable to use multiple varied training collections indexed separately in order to learn schemes that will adhere to the new constraint specified herein. On a related note, it can be determined that using only the vector length for normalisation will lead to violations of Constraints 2 and 4 (as outlined in Sect. 3.3). Furthermore, it is advisable to use medium or long queries when learning such term-weighting schemes. Short queries, for which $F2$ was learned (Fan et al. 2004), do not provide as much information about how the weights of many different terms should interact with each other (particularly in a term-discrimination context). This can be seen by the particularly poor performance of this scheme for long queries.

## 5 Conclusions

A new axiom for IR has been introduced. This axiom has been shown to be theoretically and empirically sound. We have presented four learned term-weighting schemes and one term-weighting scheme which was developed analytically. Only one of the schemes is consistent with all of the axioms. Interestingly, this scheme is one of the learned schemes. An empirical evaluation of the performance of the term-weighting schemes validates the analysis.

An interesting future direction would be to constrain the search space using the axioms and then use a learning technique to search this reduced space. Another interesting direction would be to determine if the number of particular violations of a constraint leads to poor retrieval for certain queries. Although some constraints are only conditionally satisfied (i.e. Constraints 1 and 3) by modern retrieval functions, they may or may not be violated in a normal retrieval setting. Using an inductive approach where documents and queries are examined as they grow in length, it could be determined how often a specific term-weighting scheme violates these constraints for a typical document collection. The number of times these constraints are violated may well be related to query performance. It would be interesting to further refine this set of heuristics so ultimately term-weighting performance could be judged a priori.

**Acknowledgements** This work is being carried out with the support of IRCSET (the Irish Research Council for Science, Engineering and Technology) under the Embark Initiative. The authors would also like to thank the anonymous reviewers for their detailed comments and suggestions.

## Appendix: Violations and satisfaction of constraints

Violations

*Violations of Constraints 1 and 3*

Due to the nature of the normalisation schemes used in modern retrieval functions, when a term-weighting scheme uses the document length explicitly to penalise the document, Constraint 1 (and consequently Constraint 3) can *never* be satisfied unconditionally. Consider the case where a term with an extremely low $idf$ value (i.e. where the term has negligible semantic content) is added to a document. The penalisation due to the document increasing in length will more than offset the increase in weight as the term is added (as all existing terms in the document are penalised by the document length accordingly). For the $BM25$

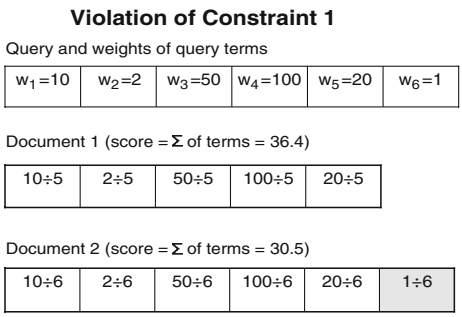


**Fig. 3** Violation of Constraint 1

scheme this will only tend to happen for terms with a very low $idf$ value (terms that appear in close to half of the documents).

Figure 3 shows a set of query terms with some basic weights applied to them. Document 1 contains 5 of the query terms while document 2 contains 6 of the query terms. The normalisation part used in the example is simply the document length. The normalisation (division by the document length) reduces the weight of *all* of the existing terms in the document and therefore, the score of a document may not increase as a query term is added. In the example shown, the score of document 1 is calculated by summing up the scores of the 5 query terms (36.4). The score of document 2 is calculated similarly (summing up the 6 query terms). As the query term added to document 2 has a very low term-discrimination weight ($w_6 = 1$) compared to the other query terms, the increase in weight due to this query term being added does not offset the increase in penalisation. The score of document 2 is only 30.5, although document 2 is created by adding a query term to document 1. However, the potential for violations of the type just described may be more prevalent in different types of term-weighting schemes. It is worth noting for the discussion presented in Appendix section "Satisfaction" that the efficiency of these types of schemes is $O(N \times |Q|)$ where $|Q|$ is the length of the query vector (usually less than 20 for even the longest queries, but often only 2 or 3 for shorter queries) and $N$ is the number of documents in the collection. This is because only query terms appearing in the document are used to determine the score of a document. This efficiency is important as test collections are becoming extremely large.

*Choice of normalisation*

The document length is typically used explicitly to penalise documents. $S1(Q, D)$ and $S2(Q, D)$ describe two possible ways of normalising a document that are used in modern weighting schemes.

$$S1(Q, D) = \sum_{t \in Q \cap D} \left( \frac{tff()}{n()} \cdot w(t) \right) \tag{8}$$

where $w(t)$ is the term-discrimination aspect, $tff()$ is the term-frequency aspect and $n()$ is some normalisation aspect. Other functions, such as the $BM25$ scheme, penalise the actual term-frequency as follows:

$$S2(Q, D) = \sum_{t \in Q \cap D} \left( tff(\frac{tf_t^D}{n()}) \cdot w(t) \right) \tag{9}$$

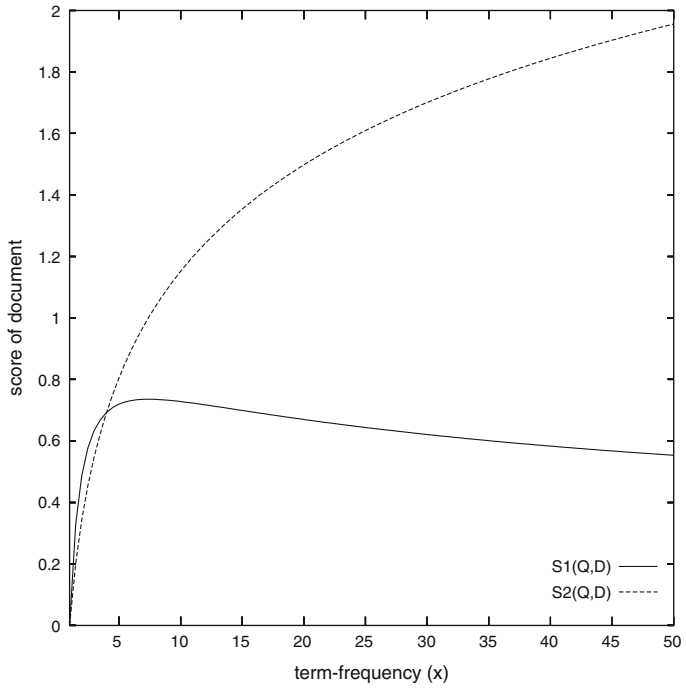


**Fig. 4** Change in score for different methods of normalisation

Both of these approaches to normalisation lead to the same potential violations of Constraints 1 and 3. Furthermore, the first method of normalisation presented ($S1(Q, D)$ in Eq. 8) violates Constraints 1 and 3 for more reasons. As the normalisation ($n()$) is independent of the term-frequency influence ($tff()$), it may grow to such a degree that the penalisation more than outweighs the increase of weight that the term-frequency provides.

Consider these two somewhat similar methods of applying normalisation (i.e. $S1(Q, D)$ and $S2(Q, D)$) from an inductive perspective. Now, let $x$ define the term-frequency for a query term. Consider a document that is made up of successive occurrences of this term. In such a case, $x$ also defines the document length. Let $\log(x)$ be the term-frequency factor and $\sqrt{x}$ be the normalisation aspect. In isolation they would appear to adhere to the aforementioned constraints (i.e. the term-frequency is sublinear and the normalisation is sublinear). Figure 4 shows that $S1(Q, D)$ (i.e. $\log(x)/\sqrt{x}$) does not always increase for successive occurrences of query-terms. $S2(Q, D)$ does adhere to this constraint in the simplest inductive case. Thus, when normalisation is explicitly used to penalise document it should be applied to the actual term-frequency as $S2(Q, D)$ (i.e. $\log(x/\sqrt{x})$) to help satisfy Constraint 1 for the simplest inductive case.

### Satisfaction

This problem identified (potential violations of Constraints 1 and 3) in Appendix section "Violations" can be overcome by penalising documents in the same way as documents are promoted when query terms occur. In the following solution, the document length is not used

explicitly to penalise a document, but when a term is found which does not occur in the query, the document is penalised as follows:

$$S3(Q, D) = \sum_{t \in D} \begin{cases} tff() \cdot w(t) \text{ if } t \in Q \\ -b \cdot tff() \cdot w(t) \text{ if } t \notin Q \end{cases}$$

where $b$ is some constant factor. In this formulation it can be seen that the document length is not used explicitly to penalise the document (i.e. there is no $n()$ function used). As non query terms occur, a weight is substracted from the overall score. Furthermore, this penalisation can be different for different types of non-query terms. In this framework, normalisation is implicit in the weighting scheme, and not explicit, as is usually the case. This type of weighting scheme has previously been explored (Jung et al. 2000). As no summary description of the document length is explicitly used, this may lead to better normalisation and subsequent retrieval (Jung et al. 2000). Just as important words are more heavily weighted, words of a high term-discrimination value that are not in the query lead to a higher penalisation as they indicate that the topic of the document varies considerably (i.e. may relate more to other subjects). This type of normalisation deals with each term separately and as such deals with the semantic content of the entire document and not just of the query (via its query terms).

Consider two documents ($D_1$ and $D_2$) that match a similar number of query terms and are of similar length. If $D_1$ contains non-query terms that are of negligible semantic content (low term-discrimination) and $D_2$ contains non-query words that have a high term-discrimination, it may be better to rank $D_1$ higher than $D_2$ as its topic is not as broad (i.e. the subject of $D_1$ is not associated to as much off-topic material as $D_2$). As such it is probably more useful to the user. This intuitively seems like a desirable property in retrieval. With such a formulation it is easy to see that Constraints 1 and 2 are adhered to. Consequently, it must adhere to the weaker Constraint 1.1. As more of the same non-query terms are added, the increase in penalisation is also reduced. This approach, however, is less efficient as the entire document vector must be examined, instead of the much shorter query vector. The efficiency of this scheme is $O(N \times |D|)$ where $|D|$ is the length of the document vector (on average this is about 150 for the collections used in the experiments described herein). Thus, there is a trade-off between unconditionally satisfying Constraints 1 and 3 and the efficiency of the approach adopted.

## References

Buckley C, Voorhees EM (2000) Evaluating evaluation measure stability. In: Proceedings of the 23rd annual international ACM SIGIR conference on research and development in information retrieval (SIGIR '00). ACM Press, New York, pp 33–40

Chowdhury A, McCabe MC, Grossman D, Frieder O (2002) Document normalization revisited. In: Proceedings of the 25th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR '02). ACM Press, Tampere, pp 381–382

Cummins R, O'Riordan C (2005) An evaluation of evolved term-weighting schemes in information retrieval. In: CIKM, pp 305–306

Cummins R, O'Riordan C (2006) Evolving local and global weighting schemes in information retrieval. Inf Retr 9(3):311–330

Cummins R, O'Riordan C (2007a) An axiomatic comparison of learned term-weighting schemes in information retrieval. In: 18th Irish conference on artificial intelligence and cognitive science, AICS 2007, Dublin Institute of Technology

Cummins R, O'Riordan C (2007b) An axiomatic study of learned term-weighting schemes. In: SIGIR'07 workshop on learning to rank for information retrieval (LR4IR-2007). Amsterdam, Netherlands, pp 11–18

Fan W, Gordon MD, Pathak P (2004) A generic ranking function discovery framework by genetic programming for information retrieval. Inf Process Manage 40(4):587–602

Fang H, Zhai C (2005) An exploration of axiomatic approaches to information retrieval. In: Proceedings of the 28th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR '05). ACM Press, New York, pp 480–487

Fang H, Tao T, Zhai C (2004) A formal study of information retrieval heuristics. In: Proceedings of the 27th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR '04). ACM Press, New York, pp 49–56

He B, Ounis I (2003) A study of parameter tuning for term frequency normalization. In: Proceedings of the twelfth international conference on information and knowledge management (CIKM '03). ACM Press, New York, pp 10–16

He B, Ounis I (2005) Term frequency normalisation tuning for BM25 and DFR models. In: ECIR, Santiago de Compostela, Spain, pp 200–214

Heaps HS (1978) Information retrieval: computational and theoretical aspects. Academic Press Inc., Orlando

Jung Y, Park H, Du D (2000) A balanced term-weighting scheme for effective document matching. Tech. Rep. TR008, Department of Computer Science, University of Minnesota, Minneapolis

Oren N (2002a) Improving the effectiveness of information retrieval with genetic programming. Master's Thesis, Faculty of Science, University of the Witwatersrand, South Africa

Oren N (2002b) Re-examining tf.idf based information retrieval with genetic programming. In: Proceedings of SAICSIT 2002 conference, pp 224–234

Porter M (1980) An algorithm for suffix stripping. Program 14(3):130–137

Robertson SE, Walker S, Hancock-Beaulieu M, Gull A, Lau M (1995) Okapi at TREC-3. In: Harman DK (ed) The third Text REtrieval Conference (TREC-3). NIST, Gaithersburg

Salton G, Buckley C (1988) Term-weighting approaches in automatic text retrieval. Inf Process Manage 24(5):513–523

Trotman A (2005) Learning to rank. Inf Retr 8:359–381