Study of Heuristic IR Constraints Under Function Discovery Framework

Parantapa Goswami  Eric Gaussier  Massih-Reza Amini
Université Grenoble Alps,
CNRS-LIG/AMA
Grenoble, France
firstname.lastname@imag.fr

ABSTRACT
In this paper we investigate the effect of the heuristic IR constraints on IR term-document scoring functions within the recently proposed function discovery framework. In the earlier study the constraints were empirically validated as a whole. Moreover, only the group of form constraints was utilized and the other prominent group, the adjustment constraints, was not considered. In this work we will investigate all the constraints individually and study them with two different term frequency normalization, namely normalization scheme used in DFR models and relative term count normalization used in language models.

Keywords
IR Theory, Function Discovery, Heuristic IR Constraints

1. INTRODUCTION
Fang et. al. [3] proposed a set of constraints which all “good” IR scoring functions should follow. These are divided into two categories, form and adjustment constraints (details are in Section 3). Among them Clinchant and Gaussier[1, 2] expressed the form conditions in analytical forms in terms of first and second order derivatives of the IR scoring function. They also studied these constraints under pseudo-relevance feedback (PRF) framework and derived conditions that PRF models should satisfy. However, these studies did not consider the other important category - the adjustment constraints.

In the recently proposed function discovery approach [5] the form constraints are successfully used as a tool to prune the search space. It is ensured that the generated functions satisfy the form constraints. An experimental validation of these constraints is also provided in light of the proposed framework, which is inline with other empirical validations of these constraints [6, 4].

However, in the original study these constraints are considered together as a single module. In this paper we will investigate the effect of each individual form constraint on the scoring functions through function discovery framework. We will also investigate two adjustment constraints, which were not considered in the original work. We will do so by taking into account two different term frequency normalization techniques, namely the normalization used in divergence from randomness (DFR) models and relative term count normalization used in language models.

2. FUNCTION DISCOVERY FRAMEWORK
The general form of retrieval status value or RSV of a document $d$ with respect to a query $q$ can be formulated as:

$$RSV(q,d) = \sum_{w \in q} a(t_w^q) \ g(w,d)$$

Where, $t_w^q$ is the number of occurrences of term $w$ within query $q$, $a : \mathbb{R}^+ \rightarrow \mathbb{R}^+$ is a positive real-valued function usually set to the identity function and the function $g(w,d)$ is called a scoring function which assigns a score to $d$ for a term $w \in q$. Standard IR models like BM25, language models, information based models, DFR models etc. all fit in the form of the above equation; and depending on the model in use, the form of the scoring function varies. Table 1 summarizes notations used throughout the paper.

| $t_w^q$ | term frequency - # of occurrences of term $w$ in document $d$ |
| $t_w^q$ | # of occurrences of term $w$ in query $q$ |
| $x_w^d$ | normalized version of term frequency |
| $N_w$ | document frequency - # of documents in the collection containing $w$ |
| $y_w$ | normalized version of document frequency |
| $N$ | # of documents in a given collection |
| $l_d$ | Length of document $d$ in # of terms |
| $l_{avg}$ | Average length of documents in a given collection |

Table 1: Notations

The function discovery approach [5] deploys a context free grammar to generate closed form formulas to be used as scoring functions. Two variables are considered in that grammar, normalized term frequency denoted by $x_w^d$ and normalized document frequency denoted by $y_w$. A real valued constant is also considered, but in experiments it is taken as 1 which we follow here as well. Thus scoring functions in this framework can also be written as $g(x_w^d, y_w)$.

Normalized term frequency can be expressed as a function of $t_w^q$ and $l_d$ in the form $NTF(t_w^q, l_d)$. [5] considers normal-
ization used in DFR models (DFR normalization) and in this study we also consider relative term count (RTC) normal- 
ization commonly used in language models. Thus one has:

$$NTF(t_w^* , l_d) = \begin{cases} t_w^* \log \left(1 + \frac{y_w}{d_w} \right) & \text{DFR normalization} \\
\frac{t_w^*}{y_w} & \text{RTC normalization} \end{cases}$$

Here $c$ is a free parameter which is taken as 1, its default value, in this work.

The normalized document frequency of a term $w$ considered in [5], as well as in this study, is the average document frequency of $w$ with respect to the total number of documents in the collection, $y_w = \frac{1}{N}$. 

3. HEURISTIC IR CONSTRAINTS

Fang et. al. [3] proposed a set of hypothetical constraints which lays a guideline of how a good IR scoring function should behave. The constraints are categorized into two groups, four form constraints and two adjustment constraints.

3.1 Form Constraints

Four form constraints define the general form of the scoring function $g$. These constraints are expressed in the following analytical forms [1]:

$$\frac{\partial g}{\partial t_w^*} > 0; \quad \frac{\partial^2 g}{\partial t_w^*} < 0; \quad \frac{\partial g}{\partial l_d} < 0; \quad \frac{\partial g}{\partial t_w} < 0$$

Considering DFR term frequency normalization and $y = \frac{1}{N}$, [5] has shown that it is sufficient for a scoring function $g$ to satisfy the following three conditions, denoted by C1, C2 and C3 respectively:

$$\frac{\partial g}{\partial t_w^*} > 0, \quad \frac{\partial^2 g}{\partial t_w^*} < 0, \quad \frac{\partial g}{\partial l_d} < 0$$

For RTC normalization it is also trivial to show that these three conditions are sufficient for any scoring function to satisfy all the form constraints.

During function generation, it is hence ensured that the generated scoring functions must satisfy C1, C2 and C3. As these constraints are same for both DFR and RTC normalization, all the generated scoring functions will satisfy the constraints for any of the two normalization schemes.

3.2 Adjustment Constraints

Two adjustment constraints aim to adjust the function $g$ satisfying the form constraints by regulating the interaction between term frequency $t_w^*$ and document length $l_d$. These two constraints are:

**C4** Let $q$ be a query. $\forall k > 1$, if $d_1$ and $d_2$ are two documents such that $l_{d_1} = k \times l_{d_2}$ and for all terms $w$, $t_w^{d_1} = k \times t_w^{d_2}$, then $RSV(q | d_1) \geq RSV(q | d_2)$.

**C5** Let $w = w$ be a single term query, for two documents $d_1$ and $d_2$ if $t_w^{d_1} > t_w^{d_2}$ and $l_{d_1} = l_{d_2} + (t_w^{d_1} - t_w^{d_2})$, then $RSV(q | d_1) \geq RSV(q | d_2)$.

Document length effect (fourth form constraint) penalizes longer documents, whereas the first adjustment constraint C4 avoids over-penalizing long documents. The second adj-justment constraint C5 ensures that a longer document must not be penalized over a shorter document if the excess length is due to the occurrences of the query term.

We present here two properties which will help to study the effect of adjustment constraints over the function discovery framework.

**Property 1.** If a function generated by the function discovery approach using RTC and DFR normalization satisfies C1, C2 and C3, then the function also satisfies C4.

**Property 2.** If a function generated by the function discovery approach using RTC normalization satisfies C1 and C3, then the function satisfies C5. If the function is generated using DFR normalization and satisfies C1, C2 and C3, then it satisfies C5 when $t_w^{d_2} \leq \frac{p \cdot f_1(p)}{f_1(0) - f_1(p)}$, where $f_1(u) = \log \left( \frac{l_{d_2} \alpha + \beta}{l_{d_2} \alpha} \right)$ and $\beta = c \cdot l_{avg}$, where $t_w^{d_2}, l_{d_2}$ are as explained in the definition of C5.

We now proceed to prove these properties. We do so by first proving the following lemmas.

**Lemma 1.** For a term $w$ if there are two documents $d_1$ and $d_2$ such that for any $k > 0$, their normalized term frequencies are $x_w^{d_1} = NTF(t_w^* , p, l_{d_1})$ and $x_w^{d_2} = NTF(t_w^* , l_{d_2})$ respectively, then:

- for RTC normalization $x_w^{d_2} \geq x_w^{d_2}$
- for DFR normalization $x_w^{d_1} \geq x_w^{d_2}$ when $t_w^{d_2} \leq \frac{p \cdot f_1(p)}{f_1(0) - f_1(p)}$ where $f_1(u) = \log \left( \frac{l_{d_2} \alpha + \beta}{l_{d_2} \alpha} \right)$ and $\beta = c \cdot l_{avg}$

**Proof.** Assuming RTC normalization, one has $x_w^{d_1} = x_w^{d_2}$, thus proving the property.

For DFR normalization it can be shown that:

$$x_w^{d_1} - x_w^{d_2} = t_w^* \log \left( \frac{(k + \alpha)^k}{k \cdot (1 + \alpha)} \right)$$

**Lemma 2.** For a term $w$ if there are two documents $d_1$ and $d_2$ such that for any integer $p > 1$, their normalized term frequencies are $x_w^{d_1} = NTF(t_w^* + p, l_{d_1})$ and $x_w^{d_2} = NTF(t_w^* , l_{d_2})$ respectively, then:

- for RTC normalization $x_w^{d_2} \geq x_w^{d_2}$
- for DFR normalization $x_w^{d_1} \geq x_w^{d_2}$ when $t_w^{d_2} \leq \frac{p \cdot f_1(p)}{f_1(0) - f_1(p)}$ where $f_1(u) = \log \left( \frac{l_{d_2} \alpha + \beta}{l_{d_2} \alpha} \right)$ and $\beta = c \cdot l_{avg}$

**Proof.** For RTC normalization $x_w^{d_1} - x_w^{d_2} = \frac{p \cdot (l_{d_1} - l_{d_2})}{l_{d_2} \cdot (l_{d_2} + \alpha)} \geq 0$ since $l_{d_2} \geq l_{d_2}$, which proves the property.

For DFR normalization it can be derived that:

$$x_w^{d_1} - x_w^{d_2} = (t_w^* + p) \log \left( \frac{l_{d_2} + p}{l_{d_2} + p} \right) - x_w^{d_2} \log \left( \frac{l_{d_2} + \beta}{l_{d_2} + \beta} \right)$$

(assuming $\beta = c \cdot l_{avg}$)

$$= (t_w^* + p) \cdot f_1(p) - t_w^{d_2} \cdot f_1(0)$$

Let $f_2(t_w^{d_2}) = (t_w^{d_2} + p) \cdot f_1(p) - t_w^{d_2} \cdot f_1(0)$, then $f_2(t_w^{d_2})$ is a strictly decreasing function with $t_w^{d_2}$ as $f_2(t_w^{d_2}) < 0$. We have $f_2(0) = p \cdot f_1(p) > 0$, but $f_2(t_w^{d_2}) \rightarrow -\infty$ as $t_w^{d_2} \rightarrow -\infty$. Thus $f_2(t_w^{d_2})$ crosses zero at $t_w^{d_2} = \frac{p \cdot f_1(p)}{f_1(0) - f_1(p)}$. So $x_w^{d_1} - x_w^{d_2} \geq 0$ when $t_w^{d_2} \leq \frac{p \cdot f_1(p)}{f_1(0) - f_1(p)}$, thus proving the property for DFR normalization.
Since queries are considered as set of terms and the order is not considered, \( RSV(q, d_1) \geq RSV(q, d_2) \) is equivalent to \( g(w, d_1) \geq g(w, d_2) \) (here we used the original form of the scoring functions as in Eq. 2). As \( g \) is satisfying \( C_1 \), i.e. \( \frac{\partial g}{\partial w} > 0 \), one has \( g(x_u^w, y) \geq g(x_u^w, y) \) if \( x_u^w \geq x_u^w \). Hence the adjustment constraint \( C_4 \) boils down to the Lemma 1, which is true, as shown above, for all the scoring functions generated using the function discovery approach with both DFR and RTC normalization. Thus all generated scoring functions are satisfying \( C_4 \) proving Property 1.

Suppose \( p > 0 \) is an integer constant such that \( t_2^d = t_2^d + p \). Then this constraint can be rewritten as, if \( l_d = t_2^d + p \) then \( RSV(q, d_1) > RSV(q, d_2) \). Again as \( g \) is satisfying \( C_1 \), one has \( g(x_u^w, y) \geq g(x_u^w, y) \) if \( x_u^w \geq x_u^w \). Thus the adjustment constraint \( C_5 \) becomes Lemma 2 and is always satisfied by the generated functions if RTC normalization is used. But for DFR normalization \( C_5 \) is satisfied only when \( t_2^d \leq \frac{p f_i(y)}{\log p} \) where \( f_i(u) = \log \left( \frac{\beta w + u}{\beta w + u} \right) \) and \( \beta = c \cdot \log w \). This proves Property 2. So for DFR normalization a generated function satisfies \( C_5 \) for not so high \( t_2^d \) values which is the case in most practical scenarios.

4. EXPERIMENTAL EVALUATION

Here we examine the effect of each constraint separately. Experiments are performed on six IR collections (Table 2), five from TREC (trec.nist.gov) and one from CLEF (www.clef-campaign.org) campaigns. These collections are indexed using Terrier IR Platform v3.5 (terrier.org). Preprocessing steps in creating an index include stemming using Porter stemmer and removing stop-words using the stop-word list provided by Terrier. Generated functions are also implemented in Terrier. We specify by \( C_V \), respectively by \( C_N \), the set of functions which satisfy all the constraints, respectively none of the constraints of a given length, and by \( C_{N_{\text{V}}} \) the set of functions which only satisfy constraint \( C_1 \).

Experiments are performed on six IR collections (Table 2), five from TREC (trec.nist.gov) and one from CLEF (www.clef-campaign.org) campaigns. These collections are indexed using Terrier IR Platform v3.5 (terrier.org). Preprocessing steps in creating an index include stemming using Porter stemmer and removing stop-words using the stop-word list provided by Terrier. Generated functions are also implemented in Terrier. We specify by \( C_V \), respectively by \( C_N \), the set of functions which satisfy all the constraints, respectively none of the constraints of a given length, and by \( C_{N_{\text{V}}} \) the set of functions which only satisfy constraint \( C_1 \).

Performances of \( C_V \) and \( C_{N_{\text{V}}} \) are compared to empirically justify the usefulness of the heuristic IR constraints as a whole.

An initial intuition can be made by the sizes of the sets \( C_N^1, C_N^2 \) and \( C_N^3 \). Figure 1 shows the number of functions in each of the sets till length 8. Clearly the number of functions satisfying \( C_2 \) is the minimum, whereas the number of functions satisfying \( C_1 \) is the maximum. Thus the constraint \( C_2 \) is the harshest one, whereas \( C_1 \) is the loosest one. Another trivial yet interesting observation is that \( C_N \) is the biggest set and \( C_V \) is the smallest one among all the five sets.

From each of the sets \( C_V \), \( C_{N_{\text{V}}} \), \( C_N^1 \), \( C_N^2 \) and \( C_N^3 \), 10 subsets are created. Each subset contains 100 randomly selected sample functions chosen from the initial set. When creating a subset, 100 functions are selected without replacement. When creating another different subset, again all functions are considered for selection. Thus a function may be repeated in different subsets but never within the same subset.

These samples are tested on CLEF-3 and TREC-3,5,6,7,8. For each function MAP is noted and it is averaged over all 100 functions within a single sample set. Finally, average

![Figure 1: Number of functions in the sets \( C_N \), \( C_N^1 \), \( C_N^2 \) and \( C_V \) till length 8.](image)

![Figure 2: Average MAP of the sets \( C_N \) (■), \( C_N^1 \) (●), \( C_N^2 \) (■) and \( C_V \) (■) till length 8 with (a) DFR and (b) RTC normalization.](image)
performance over 10 sample sets is reported.

Figure 2 shows a plot of average MAP of 10 sample sets from all five sets \( C_N, C_1^N, C_2^N, C_3^N \) and \( C_V \) with DFR and RTC normalization. As expected \( C_V \) is always best and \( C_N \) is always worst among the five sets. Performance of other three sets \( C_1^N, C_2^N \) and \( C_3^N \) are in between \( C_V \) and \( C_N \). Both for DFR and RTC, \( C_2 \) is best performing on 4 out of 6 collections. But for TREC-3 and TREC-5 \( C_3 \) is slightly better than \( C_2 \). There is no deterministic comparative pattern between \( C_1 \) and \( C_3 \). All possible relative orders in terms of performance between \( C_1 \) and \( C_3 \) are visible. As for example in case of DFR (Figure 2(a)) \( C_1 \succ C_3 \) on CLEF-3, \( C_1 \prec C_3 \) on TREC-3,5 and \( C_1 \approx C_3 \) on TREC-6,7,8. Though for RTC \( C_1 \prec C_3 \) for 4 out of 6 collections (Figure 2(b)). In summary the general trend is that \( C_2 \) is the most effective among three constraints although the plots display an inconclusive pattern. Thus it can be said that the relative effectiveness of the constraints is highly dependent on the collection in hand.

Above experiments are performed to study the effects of each constraint. But these experiments also revealed that combination of all the constraints (i.e. set \( C_V \)) always performs best. Hence for all practical purposes it is always better to utilize all the constraints together.

5. CONCLUSION

In this paper we showed that the first adjustment constraint is satisfied by all the functions generated using the approach proposed in [5] with both DFR and RTC normalization. However, the second adjustment constraint is always satisfied by all the generated functions for RTC normalization, but it is satisfied only for not so high \( t_{w}^d \) values for DFR normalization. We experimentally studied the effects of each form constraint separately and found that \( C_2 \) is the harshest among the three as it allows minimum number of functions. According to performances, for both DFR and RTC normalization, on most collections \( C_2 \) is more effective than \( C_1 \) and \( C_3 \) and there is no deterministic pattern between \( C_1 \) and \( C_3 \).

Here we have studied the constraints for DFR and RTC normalization, as three constraints \( C_1, C_2 \) and \( C_3 \) takes the same form with these two normalization schemes. For Okapi, the other popular normalization scheme, the forms of these constraints changes thus generating entirely different sets of valid functions.

6. REFERENCES


