

A Tutorial on Okapi Simple BM25F

Dr. E. Garcia

admin@miislita.com

Abstract – This is a tutorial on the Okapi Simple Best Match 25 with Extension to Multiple Weighted Fields, also known as Okapi Simple BM25F. Unlike the classic BM25, the model is applicable to structured documents consisting of multiple fields. The model preserves term frequency nonlinearity and removes the independence assumption between same term occurrences.

Keywords: simple bm25f, bm25, weighted fields, structured documents, nonlinearity, independence assumption, term occurrences

All Rights Reserved © 2011 E. Garcia; First Published: August 2, 2011.

Introduction

In a recent tutorial, we discussed Okapi Best Match 25, also known as the BM25 model. (Robertson & Zaragoza, 2009, Garcia, 2011). This new tutorial discusses a derivative model: Simple BM25 with Extension to Multiple Fields (Robertson, Zaragoza, & Taylor, 2004). Known as Simple BM25F, this model incorporates the structure of documents into the scoring process.

Background

Robertson and co-workers have developed the family of BM25 models in stages over a period of nearly 30 years (Robertson & Zaragoza, 2009). The most common implementations is given by an expression of the form

$$w_{i,j} = L_{i,j}G_i = \left\{ \frac{f_{i,j}}{k_1 \left[(1-b) + b \left(\frac{dl_j}{dl_{ave}} \right) \right] + f_{i,j}} \right\} \log \left\{ \frac{[(r_i + k)/(R - r_i + k)]}{[(n_i - r_i + k)/(N - n_i - R + r_i + k)]} \right\} \quad (\text{Eq 1})$$

where $w_{i,j}$ is the weight of term i in document j . $L_{i,j}$ is the local weight of term i in document j , and G_i is the global weight of term i in the collection of documents. In Equation 1,

$f'_{i,j} = \frac{f_{i,j}}{B}$	= normalized occurrence frequency of term i in document j .
$f_{i,j}$	= occurrence frequency of term i in document j .
dl_j	= length of document j .
dl_{ave}	= average document length.
B	= normalization factor, defined as $B = \left[(1-b) + b \left(\frac{dl_j}{dl_{ave}} \right) \right]$.
b	= tuning parameter for adjusting B and achieving full, soft, or zero normalization.
k_1	= tuning parameter for adjusting saturation of term frequencies.
k	= smoothing parameter for using the model predictively.
r_i	= number of relevant documents in the collection containing term i .
$n_i - r_i$	= number of nonrelevant documents in the collection containing term i .
n_i	= number of documents in the collection containing term i .
$R - r_i$	= number of relevant documents that do not contain term i .
$N - n_i - R + r_i$	= number of nonrelevant documents in the collection that do not contain term i .
$N - n_i$	= number of documents in the collection that do not contain term i .
R	= number of relevant documents in the collection.
$N - R$	= number of nonrelevant documents in the collection.
N	= number of documents in the collection.

where the length of a document, dl_j , is defined as the sum of index term frequencies.

$$dl_j = \sum_i^m f_{i,j} \quad (\text{Eq 2})$$

Index terms are terms from an inverted index that are present in a document. Since frequently used terms are poor discriminators of document relevance, these are usually excluded from an inverted index. Thus, if l is the total number of terms in a document, $l > m$ and $\sum_i^l f_{i,j} > \sum_i^m f_{i,j}$.

Since in practice we don't know which documents are relevant to an arbitrary query, we may assume that none of them are relevant. So for $R = r = 0$ and $k = 0.5$, Equation 1 reduces to

$$w_{i,j} = L_{i,j} G_i = \left\{ \frac{f_{i,j}}{k_1 \left[(1-b) + b \left(\frac{dl_j}{dl_{ave}} \right) \right] + f_{i,j}} \right\} \log \left[\frac{(N - n_i + 0.5)}{(n_i + 0.5)} \right] \quad (\text{Eq 3})$$

A common modification of Equations 1 and 3 consists in multiplying local weights ($L_{i,j}$) by $k_l + 1$

$$w_{i,j} = L_{i,j} G_i = \left\{ \frac{(k_l + 1) f_{i,j}}{k_1 \left[(1-b) + b \left(\frac{dl_j}{dl_{ave}} \right) \right] + f_{i,j}} \right\} \log \left[\frac{(N - n_i + 0.5)}{(n_i + 0.5)} \right] \quad (\text{Eq 4})$$

Since this is the same for all terms, it does not affect the ranking produced. The reason for using Equation 4 is to make the final scores more compatible with RSJ weights. So for a single occurrence of a term and $b = 0$, Eq 4 reduces to an RSJ weight; in this case, to $\log \left[\frac{(N - n_i + 0.5)}{(n_i + 0.5)} \right]$.

Now suppose a document is to be scored with a query q based on the presence of x number of query terms in it. This is done by computing a dot product score S of the form

$$S(d_j, q) = \vec{d}_j \cdot \vec{q} = \sum_i^x w_{i,j} w_{i,q} \quad (\text{Eq 5})$$

How do we apply this ranking function, designed for unstructured documents, to structured ones? Trying to combine all document fields into a non-structured form is certainly out of the question. How about treating each field type as collections of unstructured *document fields*?

For example, for scientific papers we may construct collections of titles, abstracts, and body sections. For Web documents, we may include anchor text fields. This is the text of links pointing to the document to be scored. We could then apply Equation 5 to each field collection and then form a linear combination of these scores. Unfortunately, this destroys the nonlinearity relationship between term weights and term frequencies. This is not a trivial issue. Most weighting functions based on term frequencies are and should be nonlinear in this parameter. According to Robertson, et al., (2004, emphasis added):

“This is desirable because of the **statistical dependence of term occurrences**: the information gained on observing a term **the first time** is greater than the information gained on subsequently seeing the same term.

For this reason, term weights **should not grow linearly** with term frequency but rather should saturate after a few occurrences.”

In other words, subsequent changes in the frequency of a term by a given factor should not change by the same factor the weight of the term and the relevance of a document to that term (Garcia, 2011).

Problem

Before scoring terms by combining field types, we need to keep in mind that preserving term frequency nonlinearity removes the independence assumption between same term occurrences, while destroying nonlinearity restores term independence.

The second problem is that scoring term weights by combining field types opens the question of how to collect global field statistics like IDF values for the individual fields. Since titles are short fields, frequently used terms in body fields and that receive a low weight (e.g., stopwords) may occur rarely in titles and should receive a high weight in the title score. Since a frequently used term is defined in relation to a field type, the result would be very unstable IDF statistics.

The third problem is that there would be no easy way of interpreting the meaning of merging evenly weighted field types. For instance due to the nonlinearity nature of term frequencies, setting all field weights to 1 does not restore the unstructured scenario of equivalently merging all fields into a large unstructured field.

A fourth problem that arises from constructing collections of field types is how to normalize the length of the fields. As noted by Robertson, et al. (2004, 2009), the initial reason for document length normalization in BM25 was to account for the verbosity and scope of the documents. It is not that clear whether normalization based on verbosity and scope should apply to the different fields, or if the whole document length should be used.

Finally, there is the question of optimization. In BM25 the two most important tuning parameters are k_1 and b . These parameters need to be optimized for each field type.

Solution

A simple workaround consists in weighting term frequencies accordingly to their field importance, combining them, and then using the resulting pseudo-frequency in Equations 3 or 4.

To illustrate, let s be a field or stream and v its weight. Suppose a document is decomposed into two fields or streams: title and body. If we assign a weight of 6 to terms in the title and a weight of 2 to terms in the body, this is equivalent to replacing the document by itself but this time repeating the title six times and the original body twice. The resulting pseudo-frequency of a term is now a linear combination of these weighted fields

$$\widetilde{f}_{i,j} = \sum_{s=1}^S v_s f_{i,s} \quad (\text{Eq 6})$$

Applying this to all terms, the new document length is

$$\widetilde{dl}_j = \sum_i^m \widetilde{f}_{i,j} \quad (\text{Eq 7})$$

The new average document length over this pseudo-collection is

$$\widetilde{dl}_{ave} = \frac{\sum_{j=1}^N \widetilde{dl}_j}{N} \quad (\text{Eq 8})$$

Equations 6 – 8 can then be used in Equations 3 or 4. This makes term weights nonlinear with the pseudo-frequencies, preserving the statistical dependence of terms and their IDF values in the original collection. Finally, the unstructured scenario is restored by setting $v = 1$ for all fields, without compromising statistical dependence.

Conclusion

Robertson and co-workers have developed the family of BM25 models in stages over a period of nearly 30 years, with BM25 being the best known of these. Unlike vector space models found in the IR literature, these models account for the verbosity and scope of documents and their lengths, with BM25F accounting for the structure of documents. When combined with other models the result is a superior model. For instance, Najork, Zaragoza, and Taylor (2007) found that a combination of BM25F and simple in-degree link analysis outperforms the combination of BM25F with PageRank or HITS authority scores. Scores were also much easier and faster to compute. In addition, Najork (2007) found that the combination of SALSA and BM25F outperforms the combination of HITS and BM25F.

Robertson, et al (2009) have made a distinction between the Simple BM25F and a modified version presented at TREC-13 (Zaragoza, Craswell, Taylor, Saria, & Robertson, 2004) in which the several free parameters of BM25F are allowed to vary between fields. This *variable* BM25F presents new challenges. To address the problem of optimizing k_1 and b , and of convergence around a global optimum, Robertson et al. have tried some heuristic techniques and tricks, including scaling of k_1 according to changes in the average term frequencies and document lengths. They have also developed what they call a Robust Linear Search optimization technique (Robertson, et. al, 2004, 2009).

Although problems associated with the optimization of ranking functions is the focus of new IR research (Joachims, Li, Liu, & Zhai, 2007; Li, Liu, & Zhai, 2008), these types of problems can be examined with some of the robust algorithms described in Sequential Simplex Optimization (SSO). Developed in the '60s by Nelder and Mead and since its introduction to Chemometrics by Deming and Morgan in the early '70, SSO has been used extensively in many Natural Science disciplines, including IR, in the presence or absence of boundary conditions or more than one local optimum (<http://www.chem.sc.edu/faculty/morgan/pubs/xd.html>).

References

Garcia, E. (2011). *A Tutorial on Okapi BM25*

<http://www.miiisita.com/information-retrieval-tutorial/okapi-bm25-tutorial.pdf>

Joachims, T., Li, H., Liu, & Zhai, C. (2007). *Learning to rank for information retrieval* (LR4IR 2007). SIGIR Forum, vol. 41, no. 2, pp. 58–62, 2007.

Li, H., Liu, T. Y., & Zhai, C. (2008). *Learning to rank for information retrieval* (LR4IR 2008). SIGIR Forum, vol. 42, no. 2, pp. 76–79, 2008.

Najork, M., Zaragoza, H., & Taylor M. (2007). *HITS on the Web: How does it Compare?* SIGIR, 2007.

<http://research.microsoft.com/apps/pubs/default.aspx?id=65139>

Robertson, S. E., & Zaragoza, H. (2009). *The Probabilistic Relevance Framework: BM25 and Beyond*.

Foundations and Trends in Information Retrieval, Vol. 3, No. 4 (2009) 333–389.

http://www.soi.city.ac.uk/~ser/papers/foundations_bm25_review.pdf

Robertson, S. E., Zaragoza, H., & Taylor, M. (2004). *Simple BM25 extension to multiple weighted fields*.

Proceedings of the 2004 ACM CIKM International Conference on Information and Knowledge Management, pp. 42–49, ACM, 2004.

Zaragoza, H., Craswell, N., Taylor, M., Saria, S., & Robertson, S. (2004). *Web and HARD track*. Microsoft Cambridge at TREC-13 (2004). In Proceedings of 13th Annual Text Retrieval Conference.

<http://trec.nist.gov/pubs/trec13/papers/microsoft-cambridge.web.hard.pdf>