Adapting the tf idf Vector-Space Model to Domain Specific Information Retrieval

Claire Fautsch
Computer Science Department
University of Neuchatel, Switzerland
claire.fautsch@unine.ch

Jacques Savoy
Computer Science Department
University of Neuchatel, Switzerland
jacques.savoy@unine.ch

ABSTRACT

The default implementation in Lucene, an open-source search engine, is the well-known vector-space model with *tf idf* weighting. The objective of this paper is to propose and evaluate additional techniques that can be adapted to this search model, in order to meet the particular needs of domain-specific information retrieval (IR). In this paper, we suggest certain specificity measures derived from either information theory or corpus-based linguistics. As an additional feature we suggest accounting for the number of search terms that a query and retrieved documents have in common. To integrate these methods we design and implement four extensions to the classical *tf idf* model and then evaluate the new IR models by applying them to four different domain-specific collections and comparing them to results found by a probabilistic retrieval model. The results tend to demonstrate that the adapted vector-space models clearly outperform the baseline approach (*tf idf*) and that performance levels obtained even surpass those found in the Okapi model.

Keywords

Domain Specific, Vector-Space Model, Term Specificity

1. INTRODUCTION

Due to the current growth in electronic resources, an increasing number of scientific journals and web sites and the emergence of blogs, efficient domain-specific information retrieval is more and more important. Biologists looking for interactions between DNA sequences and a given disease will limit their searches to the bio-medical domain, and eventually even to the most recent publications in this domain. Other users looking for specifications and opinions regarding a new cell phone would probably rather search in technical blogs. Among these users, the common need is relevant, domain-specific information. Given the extensive range of scientific publications and their use of technical language and formulae, collections on cooking recipes for example and other specialized subjects tend to use simple but specific terminology.

To make domain-specific information retrieval more efficient, the given domain is usually studied to unveil its underlying properties. For example Yu & Agichtein [1] showed orthographic variants found in the bio-medical domain are an important issue. Examples that might be encountered include spelling variants (e.g., "ecstasy", "extasy", or "ecstacy"), alternative punctuation and tokenization (e.g., "Nurr 77", "Nurr77" or "Nurr-77") or alternative names (e.g., the same protein could be named as "LARD", "Apo3", "DR3", "TRAMP", "wsl" or "TnfRSF12"). In order to improve retrieval effectiveness and account for the underlying characteristics of the corresponding domain these variations may be incorporated into the search strategy (e.g., extending the query with spelling variations extracted from a dedicated database). Such approaches cannot be easily applied to other fields.

Moreover, previous studies [2] in domain-specific IR tended to demonstrate that it is important to assign higher rankings to retrieved documents having many terms in common with the submitted query. In fact when a term occurs rarely its presence in a document surrogate may promote this document to the top of the ranked list, a phenomenon that may occur even if the document does not include additional search terms, able to more precisely specify the meaning of user's information need. Additionally, this problem tends to appear more frequently in domain-specific IR due to the fact that the precise meaning is given by a sequence of terms while a single term or a bigram may be too ambiguous (e.g., "algorithm" vs. "parallel sorting algorithm").

In this paper our intention is to propose and evaluate a variety of methods that can be applied to all domains in the same way, through accounting for both the specificity of these search terms and on the number of terms that the query and the retrieved documents have in common. Finally we also take account for the constraint to work with the classes defined in the Lucene open-source search engine[1] [3]. It is easier to extend the implemented weighting scheme base on *tf idf* weighting then to implement more complex search models such as probabilistic models.

The rest of the paper is organized as follows. Section 2 presents related work while Section 3 describes the test collections used to evaluate our proposed models. In Section 4 we expose our extended information retrieval models and describe the evaluation methodology used. Finally in Section 5 we analyze the results and draw some conclusions.

[1]http://www.lucene.apache.org/

SAC'10 March 22-26, 2010, Sierre, Switzerland.

2. RELATED WORK

In this section, we describe the previous research done on term specificity estimation and its use in domain-specific information retrieval.

Three approaches for measuring term specificity are presented in [4]. These measures are based on measures found in information theory and adapted for use in automatic hierarchy construction. As a method of identifying the domain-specific vocabulary, in a related paper Drouin [5] suggested comparing a domain-specific corpus to a more general reference corpus.

Over the last years domain specific information retrieval has become an important issue and thus has been the subject of various evaluation campaigns and tracks such as TREC[2] (Text REtrieval Conference) or CLEF[3](Cross Language Evaluation Forum). In domain-specific information retrieval, an approach currently being evaluated is document or query enhancement, using a controlled vocabulary based on a domain-specific thesaurus. For the GIRT corpus for example Petras [6] suggested using manually assigned keywords to improve retrieval results. In a second step she showed that combining pseudo-relevance feedback and query expansion by using a thesaurus could improve retrieval performance. Abdou *et al.* [7] described the impact of manually assigned descriptors taken from the MeSH thesaurus, illustrating how these descriptors could enhance retrieval performance by up to 13.5%. In the same paper they showed that extending the queries by applying automatically generated orthographic variants would slightly enhance overall retrieval effectiveness, although the outcome was less successful than expected. Other possibilities might include the use of a controlled vocabulary based on a domain-specific thesaurus to extend document representation [8] or even the generation of more specific indexing methods [2]. These suggested methods would however require various adaptations from domain to domain.

One of the goals in this paper is to go beyond extending documents and queries by means of specialized thesauri or other related lexical structures. In our opinion it would be preferable to enhance the overall retrieval performance by identifying specific search terms and thus enhancing their importance when matching queries and document surrogates.

3. TEST COLLECTIONS

To evaluate the retrieval models proposed in this paper, we use four different test corpora covering three different domains: biomedical, social sciences and blogosphere. We also consider two natural languages, namely English and German.

3.1 Genomics

The first collection was made available through the TREC evaluation campaign and had been used for the *ad hoc* Genomics retrieval track in 2004 and 2005. This corpus contains a 10-year subset from MEDLINE, a collection of abstracts and citations from publications in the bio-medical domain (containing 4,591,008 records or about 10.6 GB of compressed data), and includes a set of 100 topics as well as their relevance judgments. More information on these documents and topics can be found in [9].

[2]http://trec.nist.gov/

[3]http://www.clef-campaign.org/

3.2 Blog

The second collection was also used in the TREC evaluation campaign from 2006 to 2008 during the Blog tracks. This corpus was crawled between December 2005 and February 2006 and contains a total of 148 GB of data (or 4,293,732 documents), consisting of 753,681 feeds (38.6 GB), 3,215,171 permalinks (88.8 GB) and 324,880 homepages. In our evaluation we used only the permalink part and a total of 150 queries available for this collection. More information about this collection can be found in [10].

3.3 GIRT

The last two specific collections cover the social sciences domain. The GIRT (German Indexing and Retrieval Test database) was made available through the CLEF evaluation campaign. The original German collection contains 151,319 records taken from the social sciences while the English version is a translation of the German collection. For each language we applied a total of around 125 queries used in the CLEF domain-specific tasks from 2004 to 2008. For more information on this collection see [11].

3.4 General Corpora

Finally we needed a German and an English general reference corpus. The German reference collection was linked to a newspaper corpus containing 294,809 articles published in the *Frankfurter Rundschau* (1994), *Der Spiegel* (1994 and 1995) as well as articles from 1994 provided by the Swiss news agency (SDA).

The English corpus contains 169,477 articles extracted from the *Glasgow Herald* for 1995 as well as news articles extracted from the *Los Angeles Time* (1994). More information about both collections can be found in [12].

4. IR MODELS AND EVALUATION

The well-known vector-space model with *tf idf* weighting scheme has been adopted as the default IR model in Lucene, an open source search engine written in Java. Based on the implementation proposed, expanding this IR model is a rather straightforward procedure, but implementing the Okapi model requires substantial work. From several evaluation campaigns, e.g., TREC or CLEF, it is known that the retrieval effectiveness of the vector-space model with *tf idf* weighting is lower than certain implementations of the probabilistic model, such as Okapi [13]. As described in this section, we will extend the vector-space model to meet both the challenges of domain-specific information retrieval and those involved in improving its overall retrieval performance levels.

4.1 Vector-Space Model

As a baseline approach we used a standard *tf idf* weighting scheme with a cosine normalization. The score for the document D_i given the query Q_k was calculated by applying the following formula:

$$Score(D_i, Q_k) = \sum_{t_j \in Q_k} w_{ij} \cdot w_{kj}$$

where w_{ij} and w_{kj} respectively represent the weights of term t_j in the document D_i and in the query Q_k and are defined

as follows:

$$w_{ij} = \frac{tf_{ij} \cdot idf_j}{\sqrt{\sum_k (tf_{ik} \cdot idf_k)^2}}$$

where tf_{ij} is the frequency of the term t_j in the document D_i (or the query), idf_j the inverse document frequency computed as $\log(n/df_j)$, with n indicating the number of documents in the collection and df_j is the number of documents containing the term t_j.

4.2 Adapted Vector-Space Model

Our aim is to extend the previously described vector processing model to account for the particularities of domain-specific information retrieval, as well as an extension that would remain domain-independent. The underlying idea here is to discriminate between general and specific terms in topic formulation and as such attribute greater importance to more specific terms in the matching score. The *idf* measure can be viewed as a term specificity measure but only based on document frequency information (the number of documents in which a given term occurs) and not for example the occurrence frequency in the corpus, in a given document or compared to a general corpus.

Moreover, we wanted to assign more weight to documents having more than one search term in common with the submitted query. We therefore extended the *tf idf* model by using following formula:

$$Score(D_i, Q_k) = \sum_{t_j \in Q_k} (w_{ij} \cdot w_{kj} + spec_C(t_j))$$

where $spec_C(t_j)$ measures the specificity of the term t_j in the collection C. To measure this specificity we used the various methods described in the following paragraphs. In the proposed scoring function we adopted an addition operator to combine the *tf idf* model with the supplementary weight attached to the specificity of each search term. The advantage of this additive process is that it allows us to increase the matching score directly, according to the number of search terms that the query and the retrieved items have in common.

4.2.1 Mutual Information

Mutual information (MI) is widely used in Natural Language Processing (NLP) [14] to measure the association between two terms. We will use the MI measure presented in [15] estimating a relevance score of the term t across the collection, and calculated as follows:

$$MI(t) = \sum_i P(D_i) \cdot \log\left(\frac{P(t|D_i)}{P(t)}\right)$$

where $P(D_i) = 1/n$ is the probability of selecting the document D_i in the corpus, $P(t|D_i) = tf/l_i$ the probability of term t occurring in the document D_i, and $P(t) = cf/cl$ the probability of the term t in the collection, with cf the number of occurrences of t in the collection, cl the total number of indexing terms in the collection and l_i the length of D_i. In the current context, $spec_C(t)$ is then defined as $spec_C(t) = MI(t)$ and we denote this model as *tf idf* + MI.

4.2.2 Information Gain

Information Gain (IG) is a measure borrowed from information theory and NLP to estimate the relevance of a term t to a given document. We use the formula presented in [15] and defined as follows:

$$\begin{aligned} IG(t) \quad &= P(t) \textstyle\sum_i (P(D_i|t) \cdot \log\left(\frac{P(D_i|t)}{P(D_i)}\right) \\ &+ P(t^c) \textstyle\sum_i (P(D_i|t^c) \cdot \log\left(\frac{P(D_i|t^c)}{P(D_i)}\right) \end{aligned}$$

where $P(t^c)$ is the probability of the term t not occurring (i.e., $1 - P(t)$). The probabilities are calculated as defined in the previous paragraph. Finally we define $spec_C(t) = 1 - IG(t)$. We will reference to this model as *tf idf* + IG.

4.2.3 Relative Frequency Ratio

This third measure is based on the comparison of two corpora: a general one and a specific one. We assume here that domain-specific words are more frequent in a specific collection than in a general corpus, and based on this assumption, we calculate the specificity of the term t as follows:

$$spec_C(t) = \begin{cases} 1 & \text{if } freqR(t) \leq 1 \\ 2 & \text{if } 1 < freqR(t) < \infty \\ 3 & \text{if } freqR(t) = \infty \end{cases}$$

where $freqR(t) = \frac{f_{spec}}{f_{gen}}$ with f_{spec} and f_{gen} are the relative frequencies of the term t in the specific and the general corpus respectively (with similar size). If for a given term the relative frequency is greater than or equal to the general corpus compared to the specific one, the $spec_C(t)$ value will be one, while for the inverse it will have value of two. At the limit, when the frequency in the general corpus is null (the term does not occur), we get $freq(t) = \infty$ and thus $spec_C(t) = 3$. We reference to this measure as *tf idf* + RFR.

4.2.4 Index of Peculiarity

The final measure is based on 3-gram segmentation, normally used to detect spelling errors. In this case, each term is subdivided into tokens of length 3 (e.g., the word "house" generates the tokens "hou", "ous", and "use"). For each 3-gram (e.g., "xyz"), a *index of peculiarity* (IP) is calculated as follows:

$$IP(xyz) = \frac{\log(f(xy) - 1) - \log(f(yz) - 1)}{2} - \log(f(xyz) - 1)$$

where $f(xy)$ indicates the frequency of the bigram "xy" in the corpus, and $f(xyz)$ the frequency of the 3-gram "xyz".

In a spelling detection context, a large IP would indicate a misspelled word, while in our case a large IP means that a given term has a very specific meaning. For a given word t, the $spec_C(t)$ is calculated using the following formula.

$$spec_C(t) = \max_{xyz \in t} IP(xyz)$$

where "xyz" is a 3-gram extracted from word t. Finally, if the length of the given term is less than 3, $spec_C(t)$ is fixed at 0. We reference this model as *tf idf* + IP.

4.3 Okapi Model

To compare the results of the standard vector-space model to our new adapted models, we also used a probabilistic information retrieval model. To do so we implemented the Okapi (BM25) model proposed by Robertson *et al.* [13]. The document score was evaluated using the following formula:

$$Score(D_i, Q) = \sum_{t_j \in Q} qtf_j \cdot \log\left[\frac{n - df_j}{df_j}\right] \cdot \frac{(k_1 + 1) \cdot tf_{ij}}{K + tf_{ij}}$$

with $K = k_1 \cdot [(1 - b) + b \cdot \frac{l_i}{avdl}]$ where qtf_j denotes the frequency of term t_j in the query Q, df_j the number of documents in which the term t_j appears, l_i the length of the document D_i, and $avdl$ represents the average document length. To obtain the best retrieval performance the constants b and k_1 were set empirically according to the underlying collection.

4.4 Evaluation Methodology

To evaluate retrieval performance we used MAP [16] (Mean Average Precision), computed using the TREC_EVAL[4], using at most 1,000 retrieved documents per query to calculate MAP values.

To determine whether or not a given search strategy was statistically better than another, we applied the bootstrap methodology [17], with the null hypothesis H_0 stating that both retrieval schemes produced similar performance. In the experiments presented in this paper statistically significant differences were detected by applying a two-sided test (significance level $\alpha = 5\%$). Such a null hypothesis would be accepted if two retrieval schemes returned statistically similar means, otherwise it would be rejected.

5. EVALUATION

For the four collections we presented and tested four adapted vector-space models, as well as the classical *tf idf* and the Okapi model. Table 1 lists the results of our tests and Table 2 presents the mean improvement of each adapted model compared to the baseline approach.

The second column in Table 1 lists the results obtained on the Genomics collection, the third column those obtained for the Blog collection and the two last columns the results for the English and German GIRT corpora. Each column lists the results of the best performing model in bold print. Based on the statistical tests, we always found that there were statistically significant performance differences when comparing the classical *tf idf* and all other approaches. In all cases, the adapted vector space models perform better then the classical *tf idf* model (a difference that was always greater than +50%). When the Okapi model was used as a baseline, we used a "*" to denote those models showing statistically significant differences in the retrieval performances obtained.

For the Genomics collection, all adapted models except the IP model performed statistically at the same level as the Okapi model. To understand the effect of term specificity, we may analyze the performance of some queries. A closer look at the *tf idf* and the *tf idf*+IG models for example showed that for the query *"Comparison of Promoters of GAL1 and SUC1"* the MAP varied from 0.0323 to 1.0 when we accounted for term specificity. Indeed, this query retrieved only a single relevant document, and ranked it in position 31, when no specificity information was used (i.e., using the *tf idf* model). In total we obtained improvements for 89 of the 99 queries having at least one relevant document (he remaining query does not have any relevant document in the collection). The biggest decrease resulting from the application of specificity information was for the query *"Proteins involved in the nerve growth factor pathway"*. We noticed that except for proteins, this query did not contain any words which could be considered as belonging to biomedical domains.

[4]http://trec.nist.gov/trec_eval

	Mean Average Precision			
	Genomics	Blog	GIRT-EN	GIRT-DE
Queries	99	150	124	125
tf idf	15.58 *	19.33 *	20.79 *	23.92 *
tf idf + MI	29.41	30.91 *	31.34 *	**38.36**
tf idf + IG	**30.58**	30.82 *	32.82	38.05
tf idf+ IP	27.00 *	30.99 *	31.31 *	37.40
tf idf + RFR	30.25	30.94 *	31.66 *	37.52
Okapi	30.26	**33.57**	**33.70**	37.40

Table 1: MAP of the adapted *tf idf* vector-space models and Okapi probabilistic model

For the Blog collection, Okapi model resulted in the best performance. All other models resulted in statistically different retrieval performances when compared to Okapi. The adapted vector-space models showed considerable improvement over the standard vector-space model. When comparing the *tf idf* model and the *tf idf*+IP model for example, we observed that for 97 queries there were improvements while for 6 queries the classical *tf idf* produces a better performance. There was no change for the other 47 queries. The greatest improvement occurred with the query "Ruth Rendell" (from 0.0008 to 0.7737) in which the presence of both search terms improves the retrieved performance.

Upon an analysis of the results from the English GIRT collection, we observed that all models except the *tf idf*+IG model showed statistically significant performance differences when compared to Okapi. Even for this collection however the adapted models improved retrieval performance considerably. For the *tf idf*+IG model, the highest improvement was obtained for the query "Advertising and Ethics" where the MAP improved from 0.0374 to 0.7657 while for the query "Mortality rate" the MAP dropped from 0.5867 to 0.3702. For this model we obtained improvements for 107 queries, out of a total of 124 queries producing at least one relevant document.

Finally for the German GIRT collection we observed that all adapted vector-space models produced better performance than the Okapi model, yet these performance differences were not statistically significant. When comparing the *tf idf*+MI model to the classical *tf idf*, we observed improvements for 103 queries, with the highest improvement occurring for the query "Minderheitenpolitik im Baltikum" (MAP from 0.0957 to 0.7391) while for the query "Vaterrolle" we observed the greatest MAP decrease (from 0.7573 to 0.5545).

As depicted in Table 2, all four suggested approaches resulted in better performances than the classical *tf idf* vector-space model.

6. CONCLUSION

In this paper we presented four different extensions to the *tf idf* information retrieval model, that would allow it to be better adapted to specific domains. We have worked with the constraint to use the *tf idf* vector-space model because it is frequently used as well as implemented by the open-

Model	Mean MAP	%Change
tf idf	19.91	
tf idf + MI	32.51	+63.30%
tf idf + IG	33.07	+66.13%
tf idf + IP	31.68	+59.13%
tf idf + RFR	32.59	+63.74%

Table 2: Average improvement of the various adapted *tf idf* vector-space models compared to the baseline *tf idf*

source engine Lucene. Moreover this scheme does not have any parameters that need to be tuned.

Second, this work focused on detecting specific search terms and increasing their matching value. Various measures were proposed to identify specific terms, thus making it possible to derive various implementations. In the suggested additive scheme derived from the classical *tf idf* model, we also accounted for the number of terms that the query and the retrieved documents had in common.

We compared the suggested models to the Okapi model, a probabilistic approach, and then tested all models by applying them to four different collections written in English and German. The experiment showed that the adapted vector-space models significantly improved retrieval performances when compared to the classical *tf idf* approach. For the German collection the four adapted vector-space models even outperformed the Okapi model, while for the Genomics Collection at least three out of four adapted models produced retrieval performances that were statistically similar to those obtained by the Okapi approach.

We can thus conclude that for information retrieval in specific domains accounting for term the specificity is indeed worth the effort. One advantage of the adapted vector-space models is that no parameters are required, while this is not the case for the Okapi model. A second advantage is that these models can easily be adapted to each collection, regardless of the underlying domain or language.

Acknowledgments.

This research was supported in part by the Swiss National Science Foundation under Grant #200021-113273.

7. REFERENCES

[1] H. Yu and E. Agichtein, "Extracting synonymous gene and protein terms from biological literature," *Journal of Bioinformatics*, vol. 19, pp. 340–349, 2003.

[2] C. Fautsch and J. Savoy, "UniNE at TREC 2008: Fact and opinion retrieval in the blogsphere," in *The Seventeenth Text REtrieval Conference Proceedings (TREC 2008)*, 2008.

[3] E. Hatcher and O. Gospodnetic, *Lucene in Action (In Action series)*. Manning Publications, December 2004.

[4] P. M. Ryu and K. S. Choi, "Measuring the specificity of terms for automatic hierarchy construction," in *ECAI-2004 Workshop on Ontology Learning and Population*.

[5] P. Drouin, "Detection of domain specific terminology using corpora comparison," in *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC'04)*.

[6] V. Petras, "How one word can make all the difference - using subject metadata for automatic query expansion and reformulation," in *Working Notes for the CLEF 2005 Workshop, 21-23 September, Vienna, Austria*, 2005.

[7] S. Abdou and J. Savoy, "Searching in Medline: Query expansion and manual indexing evaluation," *Information Processing & Management*, vol. 44, pp. 781–789, 2008.

[8] C. Fautsch and J. Savoy, "Comparison between manually and automatically assigned descriptors based on a German bibliographic collection," in *Proceedings of the 6th International Workshop on Text-based Information Retrieval (TIR 2009)*, 2009.

[9] W. Hersh, A. Cohen, J. Yang, R. T. Bhupatiraju, P. Roberts, and M. Hearst, "TREC 2005 Genomics Track Overview," in *The Fourteenth Text REtrieval Conference Proceedings (TREC 2005)*, 2005.

[10] C. Macdonald and I. Ounis, "The TREC Blogs06 collection : Creating and analysing a blog test collection," *DCS Technical Report Series*, 2006.

[11] M. Kluck, "Die GIRT-Testdatenbank als Gegenstand informationswissenschaftlicher Evaluation," in *ISI* (B. Bekavac, J. Herget, and M. Rittberger, eds.), vol. 42 of *Schriften zur Informationswissenschaft*, pp. 247–268, Hochschulverband für Informationswissenschaft, 2004.

[12] C. Peters, P. Clough, J. Gonzalo, G. J. F. Jones, M. Kluck, and B. Magnini, eds., *Multilingual Information Access for Text, Speech and Images, 5th Workshop of the Cross-Language Evaluation Forum, CLEF 2004, Bath, UK, September 15-17, 2004, Revised Selected Papers*, vol. 3491 of *LNCS*, Springer, 2005.

[13] S. E. Robertson, S. Walker, and M. Beaulieu, "Experimentation as a way of life: Okapi at TREC," *Information Processing & Management*, vol. 36, pp. 95–108, 2000.

[14] P. M. Nugues, *An Introduction to Language Processing with Perl and Prolog*. Berlin: Springer, 2006.

[15] C. Orăsan, V. Pekar, and L. Hasler, "A comparison of summarisation methods based on term specificity estimation," in *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC2004)*, 2004.

[16] C. Buckley and E. M. Voorhees, "Retrieval system evaluation," in *TREC: Experiment and Evaluation in Information Retrieval*. (E. M. Voorhees and D. K. Harman, eds.), (Cambridge, MA), pp. 53–75, MIT Press, 2005.

[17] J. Savoy, "Statistical inference in retrieval effectiveness evaluation," *Information Processing & Management*, vol. 33, pp. 495–512, 1997.