ENSM-SE at CLEF 2005 : Uses of fuzzy proximity matching function

Annabelle MERCIER, Amelie IMAFOUO and Michel BEIGBEDER
Ecole Nationale Superieure des Mines de Saint Etienne (ENSM-SE)
158 cours Fauriel 42023 Saint Etienne Cedex 2 FRANCE
{annabelle.mercier,imafouo,mbeig}@emse.fr

August 19, 2005

Abstract

Based on the idea that the closer the query terms in a document are, the more relevant this document is, we propose a information retrieval method based on a fuzzy proximity degree of term occurences to compute document relevance to a query. Our model is able to deal with Boolean queries, but contrary to the traditional extensions of the basic Boolean information retrieval model, it does not explicitly use a proximity operator. A single parameter allows to control the proximity degree required. We explain how we construct the queries and we report the results of the experiments of the CLEF 2005 campaign before the conclusion.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.4 Systems and Software

General Terms

Experimentation

Keywords

Proximity, fuzzy set theory, fuzzy proximity, term density

1 Introduction

In information retrieval domain, systems are founded on three basic ones models: The Boolean model, the vector model and the probabilistic model which were derived within many variations (extended Boolean models, models based on fuzzy sets theory, generalized vector space model,...) [1]. Though they are all based on weak representations of documents: either sets of terms or bags of terms. In the first case, what the information retrieval system knows about a document is if it contains or not a given term. In the second case, the system knows the number of occurence – *term frequency*, tf – of a given term in each document. So whatever is the order of the terms in the documents, they share the same index representation if they use the same terms. The worthy of note exceptions are most of the Boolean model implementations which propose a NEAR operator [10]. This operator is a kind of AND but with the constraint that the different terms are within a window of size n, where n is an integral value. The set of retrieved documents can be restricted with this operator, for instance, it is possible to discriminate documents about "data structures" and those about "data about concrete structures". The result is an increase in precision of the system [5]. But the Boolean systems that implement a NEAR operator share the same limitation as any basic Boolean system : These systems are not able to rank the retrieved documents because with this model a document *is* or *is not* relevant to a query. In fact, different extensions were proposed to the basic Boolean systems to circumvent this limitation. These extensions represents the documents with some kind of term weights most of the time computed on a tf basis. Then they apply some combining formulas to compute the document score given the term weigths and the query tree. But these extensions are not compatible with the NEAR operator. So some works defined models that attempt to directly score the documents by taking into account the proximity of the query terms within them.

2 Many uses of proximity

Three methods were proposed to score the documents by taking into account some set of intervalls containing the query terms. These methods differ in the set of intervalls that are selected in a first step, and then in the formulas used to compute a score for a given interval. The method of Clarke and al. [2] selects the shortest intervals that contains all the query terms (This constraint is relaxed if there are not enough retrieved documents), so the intervals can not be nested. In the methods of Hawking and al. [4], for each query term occurence, the shortest interval containing all the query terms is selected, thus the selected intervals can nest. Rasolofo and al. [8] chose to select intervals only containing *two* terms of the query, but with the additionnal constraint that the interval is shorter than five words. Moreover, the passage retrieval methods use indirectly the notion of proximity. In fact, in several methods, document ranking is doing by selecting documents which have passages with high density of query terms that-is-to-say documents where the query terms are closed [11, 3, 6]. The next section presents our method based on term proximity to score the documents.

3 Fuzzy proximity interpretation of queries

To address the problem of scoring the documents by taking into account the relative order of the words in the document, we have defined a new method based on a *fuzzy proximity* between each position in the document text and a query. First, given a document d and a term t, we define a term proximity function $w_{d,t}$. We can use different types of kernel (hamming, rectangular, gaussian) for the function but a triangular one is computed. A k constant controls the support of the function and this support represents the extent of each term occurence influence. This function reaches its maximum (value 1) at each occurence of the term t in the document d and linearly decreases on each side down to 0. So for each query term t, we determine the fuzzy proximity at each position of the document d retrieved. When the zone of influence of two terms occurrences overlaps in a document position x the value of the nearest term occurrence is taken so:

$$w_t^d(x) = \max_{i \in Occ(t,d)} f(x-i)$$

where $Occ(t,d)$ is the set of occurrence positions of term t in the document d and f the influence function kernel.

The figures 1 and 2 show the fuzzy proximity function w_A (resp. w_B) for the term A (resp. B) in the document d_0 and d_1.

The query model is that of the classical Boolean model: A tree with terms on the leaves an OR or AND operators on the internal nodes. Given a query q, the term proximity functions located on the query tree leaves are combined in the query tree with usual formulas pertaining to the fuzzy set theory. We compute here the fuzzy proximity of the query. So the fuzzy proximity is computed by :

$$w_{q\,\mathrm{OR}\,q'} = \max(w_q, w_{q'})$$

for a disjunctive node and by

$$w_{q\,\mathrm{AND}\,q'} = \min(w_q, w_{q'}).$$

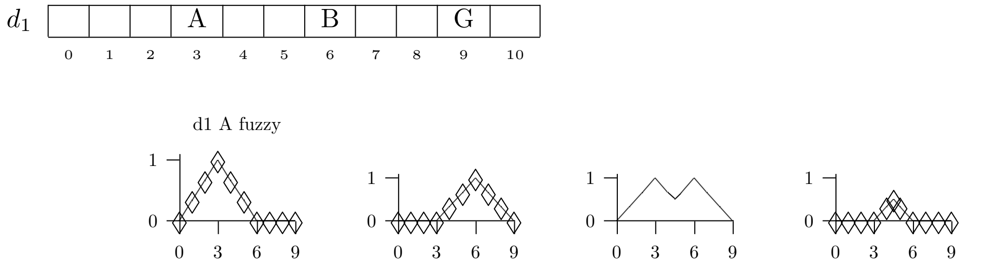

Figure 1: Document 1 – In order, we show w_A^{d1}, w_B^{d1}, $w_{A\,OR\,B}^{d1}$ and $w_{A\,AND\,B}^{d1}$.

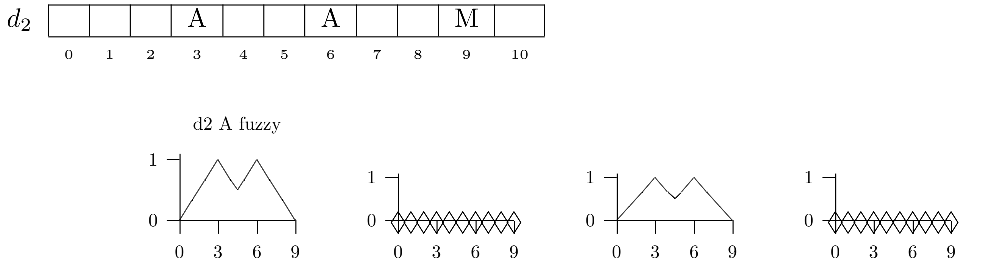

Figure 2: Document 2 – In order, we show w_A^{d2}, w_B^{d2}, $w_{A\,OR\,B}^{d2}$ and $w_{A\,AND\,B}^{d2}$.

for a conjunctive node.

So we obtain a function $w_{d,q}$ from the set of positions in the document text to the interval $[0, 1]$. The result of the integration of this function is used as the score of the document :

$$s(q,d) = \int_{-\infty}^{+\infty} w_q^d(x)\ \mathrm{d}x,$$

Finally, the computed score $s(q, d)$ depends on fuzzy proximity function and allows to rank document according to query term proximity.

4 Experiments and evaluation

We carried out experiments on the CLEF 2004 evaluation campaign [1] test collection. We use the retrieval tool Lucy that which is based on the Okapi BM-25 information retrieval model [9]. to index this collection. This tool is adapted to our method because it keeps in the index the terms positions of the documents. thus, we extend the tool to compute similarity values for our fuzzy proximity matching function.

Documents in the CLEF 2005 test collection are newspapers articles in XML format *SDA* and *Le Monde* of the years 1994 and 1995. For each document (tag <DOC>), we keep the fields <DOCNO> with the tag and the document number by Lucy, the textual contents of the tags <TX>, <LD>, <TI>, <ST> for *SDA French* and <TEXT>, <LEAD1>, <TITLE> for *Le Monde* 1995. We used the topics and the relevance judgements to evaluate the different methods by the trec_eval program.

4.1 Building the queries

Each topic has three tags: <FR-title>, <FR-desc>, <FR-narr>. We built three set of queries for our experiments. Queries are either manually or automatically built from the textual contents of the "title" and the "description" tags.

[1] http://clef.isti.cnr.it/

For automatic built queries (*two* sets): For the first set, a query is made of terms from the "title" field; for the second set, a query is made of terms from the "description" field, stop words[2] are removed. Below we give the results for the first set of queries. Let show the steps for building an automatic query using the "title" by giving an example with the topic 278. The original topic is expressed by :

```
<top>
<num> 278 </num>
<FR-title> Les moyens de transport pour handicaps </FR-title>
<FR-desc> A quels problmes doivent faire face les personnes handicapes
 physiques lorsquelles empruntent les transports publics et quelles
 solutions sont proposes ou adoptes? </FR-desc>
<FR-narr> Les documents pertinents devront dcrire les difficults
auxquelles doivent faire face les personnes diminues physiquement
lorsquelles utilisent les transports publics et/ou traiter des progrs
 accomplis pour rsoudre ces problmes. </FR-narr>
</top>
```

First, the number and the title fields are extracted so we have : `<num> C278 </num>` `<FR-title> Les moyens de transport pour handicaps </FR-title>` And we compact like this : `278 moyens transport handicapes`

From this query, we make some derivations "automatically" :

Lucy `278 moyens transport handicapes`

conjunctive fuzzy proximity `249 moyens & transport & handicapes`

disjunctive fuzzy proximity `249 moyens | transport | handicapes`

Manual built queries, (*one* set): are made of terms from the "title" field and additionnaly terms from the "description" field. Moreover, we add the plurial form of the terms and the terms derivation to compensate the LUCY tool lack of stemming. We thus obtain queries that are conjunction of disjunctions of the different derivations of the terms. On the other hand, the evaluation by the LUCY tool uses flat queries that are of different derivations of the terms. We give an example with the topic 278 as previously: **Lucy** `278 moyen moyens transport transports handicap handicape handicapes`

stemming fuzzy proximity `278 (moyen | moyens) & (transport | transports) & (handicap | handicape | handicapes)`

4.2 Building the result lists

We compare the Okapi model and our fuzzy method with different values of k. As we know on one hand that the Okapi method is one of the best performing one and on the other hand a previous study showed that the proximity based methods improve retrieval [7], we decide to merge the Okapi results list with the results lists provided by proximity based methods. Consequently, if one of the proximity based method does not retrieve enough documents, then its results list is supplemented by the documents from the Okapi results list that have not yet been retrieved by proximity based methods; the maximum number of documents retrieved is $1,000$.

4.3 Differents runs

In the officials runs, the queries are constructed :

1. automatically with terms conjunction of title field and test with $k = 20$ (run RIMfuzzET020) and $k = 50$ (run RIMfuzzET050),

2. manually with terms of three fields and test with $k = 50$ (run RIMfuzzLemme050) and $k = 80$ (run RIMfuzzLemme080).

[2] stop words removed: à, aux, au, chez, et, dans, des, de, du, en, la, les, le, par, sur, uns, unes, une, un, d', l'

Recall	Lucy	fuzzET050	fuzzET020
0	**62**	*59*	57
10	**45**	*44*	44
20	**33**	32	*33*
30	**26**	25	*25*
40	21	*21*	**21**
50	19	*19*	**19**
60	**14**	14	*14*
70	*11*	11	**11**
80	7	**8**	*8*
90	4	4	4
100	1	1	1

Figure 3: Automatic runs

Recall	LucyLemme	fuzzLemme080	fuzzLemme050
0	68	**70**	*68*
10	**49**	*49*	48
20	39	**41**	*41*
30	31	*33*	**33**
40	25	*28*	**28**
50	21	**22**	*21*
60	17	**18**	*18*
70	13	**14**	**14**
80	8	**10**	**10**
90	5	**6**	*6*
100	1	*1*	**1**

Figure 4: Manual runs

For the runs RIMLucyET and RIMLucyLemme, the queries are flat (bag of terms) and these runs provide two baselines produced by using basic Lucy search engine. The recall precision results are provided in the figure 4.3 for the automatic runs and in the figure 4.3 for manual runs.

With the values chosen for the officials runs, unfortunally, the Lucy method performs better than the fuzzy proximity ones but when manuals queries are used the result are better or equal to the Lucy ones.

Amount the unofficial runs, we change the value of the k constant to enlarge the area of influence of a term occurrence. In the figure we notice that the largest the area is the better the results are. The fuzzy proximity method perform better with manual queries (run RIMLemme*) because we retrieved more documents with our method so the proximity between query terms is the main factor to select and rank documents.

5 Conclusion

We have presented and experimented our information retrieval model which takes into account the position of the term occurences in the document to compute a relevance score on the CLEF 2005 Ad-Hoc french test collection. We notice that the higher the area of influence of term is the better the results are. In futher experiments, we are going to use another influence function more flexible which allows to adapt the value of k constant to the number of retrieved documents. We think also that the results can be improved by using a stemming step before indexing and by use

Recall	fuzzET100	fuzzET200	Lucy	fuzzLemme100	fuzzLemme200	LucyLemme
0	0.5950	0.6105	**0.6210**	**0.7224**	0.7127	0.6808
10	0.4372	0.4348	**0.4496**	0.5014	**0.5079**	0.4904
20	0.3254	0.3277	**0.3341**	0.4043	**0.4071**	0.3925
30	0.2561	0.2553	**0.2638**	0.3284	**0.3407**	0.3082
40	0.2117	**0.2125**	0.2110	0.2756	**0.2771**	0.2517
50	0.1856	0.1851	**0.1914**	**0.2160**	0.2157	0.2080
60	0.1434	0.1405	**0.1462**	0.1827	**0.1829**	0.1746
70	0.1119	0.1095	**0.1123**	**0.1443**	0.1441	0.1275
80	**0.0789**	0.0774	0.0726	0.1019	**0.1023**	0.0804
90	0.0441	0.0437	**0.0442**	**0.0626**	0.0616	0.0463
100	**0.0121**	0.0121	0.0121	**0.0130**	0.0130	0.0123

Figure 5: Unofficial automatic and manual runs

a thesaurus to retrieved more documents with our fuzzy proximity method.

References

[1] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval.* ACM Press / Addison-Wesley, 1999.

[2] Charles L. A. Clarke, Gordon V. Cormack, and Elizabeth A. Tudhope. Relevance ranking for one to three term queries. *Information Processing and Management*, 36(2):291–311, 2000.

[3] Owen de Kretser and Alistair Moffat. Effective document presentation with a locality-based similarity heuristic. In *SIGIR '99: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 113–120. ACM, 9 1999.

[4] D. Hawking and P. Thistlewaite. Proximity operators - so near and yet so far. In D. K. Harman, editor, *The Fourth Text REtrieval Conference (TREC-4)*, number 500-236. Department of Commerce, National Institute of Standards and Technology, 1995.

[5] E. M. Keen. Some aspects of proximity searching in text retrieval systems. *Journal of Information Science*, 18:89–98, 1992.

[6] Koichi Kise, Markus Junker, Andreas Dengel, and Keinosuke Matsumoto. Passage retrieval based on density distributions of terms and its applications to document retrieval and question answering. In Andreas Dengel, Markus Junker, and Anette Weisbecker, editors, *Reading and Learning: Adaptive Content Recognition*, volume 2956 of *Lecture Notes in Computer Science*, pages 306–327. Springer, 2004. No electronic version.

[7] A. Mercier. Etude comparative de trois approches utilisant la proximit entre les termes de la requte pour le calcul des scores des documents. In *INFORSID 2004*, pages 95–106, mai 2004.

[8] Yves Rasolofo and Jacques Savoy. Term proximity scoring for keyword-based retrieval systems. In *25th European Conference on Information Retrieval Research*, number 2633 in LNCS, pages 207–218. Springer, 2003.

[9] S. E. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. Okapi at trec-3. In D. K. Harman, editor, *Overview of the Third Text REtrieval Conference (TREC-3)*, number PB95-216883, pages 109–. Department of Commerce, National Institute of Standards and Technology, 1994.

[10] Gerard Salton and M. J. McGill. *Introduction to Modern Information Retrieval.* McGraw-Hill Book Company, 1983.

[11] Ross Wilkinson. Effective retrieval of structured documents. In *SIGIR 94 proceedings*, pages 311–317. Springer-Verlag New York, 1994.