Combining Words and Compound Terms for Monolingual and Cross-Language Information Retrieval

Jian-Yun Nie, Jean-François Dufort
Dept. IRO, University of Montreal,
CP. 6128, succursale Centre-ville
Montreal, Quebec, H3C 3J7 Canada
nie@iro.umontreal.ca

Abstract:
Most existing systems of Information retrieval (IR) use single words as index to represent the contents of documents and queries. One of the consequences is the low recall level. In this paper, we propose to integrate compound terms as additional indexing units because terms are more precise representation units than words. Terms are recognized through the use of a terminology database and an automatic term extraction tool, which is based on syntactic templates and statistical analysis. In this paper, we first show that the use of compound terms is greatly beneficial to monolingual IR. Then compound terms are incorporated in statistical translation models trained on a large set of parallel texts. Our experiments on cross-language information retrieval (CLIR) show that such a translation model leads to a much better CLIR effectiveness when compound terms are integrated.

Keywords: information retrieval, compound term, translation model, cross-language information retrieval, query translation.

1. Introduction

Most information retrieval (IR) systems currently in use are based on simple words, which are used as indexes for documents and queries. The estimation of a document's relevance to a query is based on a sharing of keywords between them. For example, in a Boolean IR system, for a query represented by (*a* and *b*), the documents retrieved by the system must contain both the keywords *a* and *b*.

The word-based approach has been criticized in a number of studies. Much criticism is focused on the imprecision of word-based representation: The content of a document (or a query) cannot be captured precisely by a set of words. For example, a document describing "search engine" will be represented by the words {*search, engine*}. However, these same words also represent the meanings of "… search for used engines of cars …", "… search … ecologic engines …", "economical engines … search …", and so on, which are not related to "search engine". This fact leads to a high noise ratio or low recall ratio. It is due to word ambiguity (e.g. for the word "engine") and the lack of inter-word relationship in the representation (between "search" and "engine").

To solve this problem, both word disambiguation [14] and semantic representation [3] approaches have been proposed. Word disambiguation tries to recognize the exact meaning of each word. The recognized word sense, instead of word, is used to represent the contents of the document. The recognition of semantic relationships goes even further: it also tries to recognize the semantic relationship between words or the concepts they represent (e.g. the "engine" is *for_the_purpose_of* "search"). Unfortunately, the previous research results show that it is difficult to arrive at a satisfactory disambiguation rate: very often it is well below 70% [14, 15]. This means that about 1/3 of the senses assigned to words may be wrong. Both approaches can be applied only in limited areas. It is known that they can hardly scale up.

A more modest approach to arrive at a more precise representation is the one that uses compound terms. It is usually assumed that compound terms are less ambiguous than single words, and they represent a more precise meaning. For example, "search engine" as a term represent an unambiguous meaning, and implicitly, the semantic relationship between "search" and "engine" is encoded within the term.

Previous studies have suggested two approaches to identify compound terms: one is through the use of a dictionary of compound terms that is build manually [9]; another is through an automatic syntactic and statistical analysis [4, 5]. However, the impact of the addition of compound terms has not always been positive [13].

We notice that there are basically two problems to solve when one tries to use compound terms in IR:

- the recognition of compound terms;
- the way that compound terms are integrated into IR process.

Most previous research has focused on the first problem, while using a straightforward way to integrate the compounds in IR. In most of the cases, compounds are used to replace single words, or they are added into the same vector as single words. In the first case, one usually obtained much lower recall ratio because only a part of the document contents is represented by compound terms. In the second case, the global effectiveness is almost unchanged.

In this paper, we will show that with a more reasonable integration of compound terms, the effectiveness of IR can be substantially improved.

The second problem we address is the use of compound terms in cross-language information retrieval (CLIR). CLIR tries to retrieve documents with a query written in a different

language. The most critical problem, in addition to those of monolingual IR, is the translation of the query.

There are three possible ways to translate a query:

- by a machine translation (MT) system;
- by exploiting a bilingual dictionary;
- or by exploiting a set of parallel texts.

In the previous experiments [10] it is shown that the second approach used in a direct way does not lead to a satisfactory result. With a good MT system the first approach can lead to a high effectiveness. It is usually around 80-90% of that of monolingual IR. The third approach can be as good as the first one if the parallel texts are large enough and that they are exploited correctly [7, 10]. In comparison with the first approach, the third one has the advantage that there is no need for a huge amount of manual preparation. The translation tool is trained automatically from the parallel texts. So in our study, we use the third approach.

In some sense, the extraction of compound terms in CLIR is even more crucial than in monolingual IR. In fact, if a query is translated word-by-word, many possible translation words will be suggested, some of them being unrelated to the given sense. This is again due to the great ambiguity of single words. For example, if one tries to translate the query "search engine" word by word, very likely, we will also obtain the translation of "engine" as "mechanical engine". If the term can be translated as a whole, this unrelated meaning can be eliminated, or its weight will be much lower.

In this paper, we will show that we can obtain a better CLIR effectiveness if the translation model incorporates translations of compound terms.

The paper will be organized as follows. In Section 2, we will describe our approach for monolingual IR, which incorporates compound terms. Significant improvements will be shown in our experiments on two test collections in English and French. In Section 4, compound terms will be integrated into translation models. Again, significant improvements will be shown. Finally, Section 5 gives some conclusions.

2. Compound terms in monolingual IR

As we mentioned earlier, there are basically two approaches for their recognition: using a man-built terminology database or a dictionary of compound terms; using an automatic syntactico-statistical analysis. In the following subsections, we will first describe the two approaches. Then experimental results on monolingual IR will be presented.

2.1. Using a terminology database

A terminology database contains a large set of terms used in different specializations. In addition, several relations between terms are also created between terms, e.g. synonymy, hypernymy and hyponymy. The assumption of using a terminology database to recognize terms is that the database contains most of the important compound terms. Therefore, we can simply extract the stored terms from texts to form additional indexes for IR. In our case, we use a large database that contains over 1 million terms in both English and French. This database is the union of two large databases created by the Governments of Canada and Quebec for the purposes of translation and normalization of technical terminology in French. An English term is translated into French and vise versa. A certain number of them are long idiomatic expressions. Such expressions will likely not appear in our documents to be searched. Even if they do, their frequency will be very low, and their impact on IR will be small. So we do not consider the expressions whose length is more than 20 characters. Once the filtering is done, more than half of the terms are removed from the database. The following table contains some statistics of the remaining database we used (after filtering). Among the terms, there are respectively 57% and 75% compound terms in English and French.

Table 1. Statistics on the terminology database

English	# terms	527 549
	# compounds	300 025
French	# terms	395 302
	# compounds	295 683

The terms stored in the database are supposed to be in a standard form. However, there still may be slight form differences between the terms in the database and those in documents. For example, the database may contain a singular form of the term (e.g. database system), whereas in a document, it is in plural form (e.g. database systems). Such differences are not meaningful for IR. If the expressions in the database and in the documents are not unified somehow, the extraction process will recognize only a part of the terms. Therefore, the following term standardization process is carried out:

- Nouns in plural are transformed into singular form (e.g. systems → system);
- Verbs are changed into infinitive form (e.g. retrieves → retrieve, retrieving → retrieve);
- Articles in a term is removed (e.g. the database system)

The first two transformations are done with a statistical tagger [6]. The English tagger is trained on Penn Tree-bank, and the French tagger is trained on an equivalent in French. The tagger tries to determine the most probable POS tag for each word in a sentence such that the global tagging of the sentence receives the maximum probability.

Once the POS tags are determined, the corresponding morphological rules are applied to transform the word into the standard form (called the citation form).

For example, the expression "adjusted the earnings" will be transformed into "adjust earning".

Once the preprocessing is done, the size of the terminology database is further reduced, as shown in the following table.

Table 2. Statistics on the processed terminology database

English	# terms	392 962
	# compounds	292 375
French	# terms	384 208
	# compounds	289 500

The same preprocessing is carried out on the documents. Then the extraction process is quite straightforward. A document text is linearly scanned from the beginning to the end. At each position, we determine what terms of the database appear at the beginning of the word sequence. These terms are extracted, and added to the original text.

For example, suppose a preprocessed text as follows:

```
<text>
     arm dealer prepare relief supply
     to soviet union
```

From this segment, we can extract two stored terms "arm dealer" and "soviet union" when the scanning arrives at the positions "arm" and "soviet". So the text is extended into the following form:

```
<text>
     arm dealer prepare relief supply
     to soviet union
<term>
     arm_dealer soviet_union
```

2.2. Extraction of terms by a syntactico-statistical analysis

Another method to extract compound terms uses syntactic structures, together with a statistical analysis. First, word sequences corresponding to predefined syntactic templates are extracted as candidates. If the frequency of occurrences of a candidate is above a certain threshold, then the sequence is considered as a compound term.

The first problem is the definition of the syntactic templates. This is done manually according to the general knowledge on syntactic structures of a language. Usually the extraction is restricted to noun phrases. For example, the following template is used in the tool we used - Exterm:

```
((NC|AJ) )*((NC|AJ)|NC PP) ((NC|AJ) )*NC
```

Of course, a POS tagging is necessary in order to recognize the syntactic category of each word. Again, we use the statistical tagger mentioned earlier.

A statistical analysis follows, which ensures that a sequence is relatively frequent in a text. The higher we set the threshold, the more the terms extracted are precise; however, the more likely we will also miss good terms. The setting of the threshold may have a great impact of the resulting term candidate. The best threshold should be found through a series of experiments. As the goal of this study is to carry out a preliminary test on whether the terms extracted by such a program can be useful for IR, we do not test different values of the threshold. The threshold is fixed at 2 for our experiments.

2.3. IR system

In our experiments we use the SMART system. SMART is an IR system, developed in Cornell University [2]. The indexing process considers every token as an index. Indexes are weighted according to the *tf***idf* weighting scheme[1]. This is a common way to weigh the importance and uniqueness of a term in a document. The principle is as follows: 1) The more a word occurs in a document, the more it is important. This is the *tf* factor. On the other hand, the more there are documents containing the word, the less the word is specific to one particular document. In other words, the word does not allow to distinguish a document from the others. Therefore, the weight of the word is lowered. This is the *idf* factor. More precisely, the two factors are measures as follows:

$$tf(t, D) = \log(freq(t, D) + 1);$$

$$idf(t) = \log(\frac{N}{n(t)})$$

where *freq*(*t, D*) is the frequency of occurrences of the word/term *t* in the document *D*; *N* is the total number of documents in the collection; *n*(*t*) is the number of documents containing *t*.

The retrieval process follows the vector space model [11]. In this model, a vector space is defined by all the tokens (words or terms) encountered in the documents. Each word/term represents a distinct dimension in this space. Then a document, as well as a query, is represented as a vector in this space. The weight in a dimension represents the importance of the corresponding word/term in the document or query (the *tf***idf* weight). The degree of correspondence between a document and a query is estimated by the similarity of their vectors. One of the commonly used similarity measures is as follows:

$$sim(D,Q) = \frac{\sum_i d_i \times q_i}{\sqrt{\sum_i d_i^2 \times \sum_i q_i^2}}$$

SMART also has the flexibility of indexing different fields of the text separately. For example, we can put the indexes encountered in <text> filed and <term> field in two separate vectors. If both the document and the query are represented by two separate vectors, then the global similarity between the document and the query is calculated as follows:

$$sim(D,Q) = \sum_j I_j \times sim(D_j, Q_j)$$

[1] *tf* = term frequency, and *idf* = inversed document frequency.

where D_i and Q_j are respectively the separated vectors for the document and the query; and I_j the relative importance for the vector j. In other words, we are able to assign a relative importance to each filed in the process of retrieval. In our incorporation of compound terms, we will make use of this flexibility.

3. Experiments on monolingual IR

Our experiments have been conducted on the two corpora used in TREC6 and TREC7 [8]. The English AP collection contains 242 918 documents and the French SDA collection 141 656 documents. 25 queries have been manually evaluated queries. They are provided in both French and English.

3.1. Adding terms as additional indexes

One of the possibilities is to use the terms identified to replace words. This means that we only consider the <term> filed added during the term extraction process. However, as compound terms only covers part of the contents of the document or the query, the indexes will not have a full coverage. Therefore, we use the identified terms as additional indexes to words identified by the traditional indexing approach.

In our first experiment, we add the identified terms into the same vector as words. This approach is similar to the previous studies. The following table shows the resulting retrieval effectiveness[2].

Table 3. Effectiveness of monolingual IR by adding compounds in the same vector.

Average precision	Trad. IR	TermDB (change)	Exterm (change)	TermDB +Exterm
English AP	0.2520	0.2432 (-3.5%)	0.2523 (+0.1%)	0.2478 (-1.7%)
French SDA	0.2356	0.2358 (0.1%)	0.2469 (+4.8%)	0.2470 (+4.9%)

As we can see the effectiveness is only changed marginally. This result is similar to those of the previous studies, that merging compound terms with words is not an effective approach.

We observe that, despite the large size of our terminology database, the incorporation of its terms is not very helpful. In comparison the terms identified by Exterm have a better impact on IR effectiveness.

In the following experiments, we will simply test with all the terms identified by both approaches (i.e. TermDB+ Exterm).

3.2. Separating terms from words

We observed in the combined vector of words and compound terms that in many cases, the weights of compound terms are unduly high, in comparison with those of simple words. The reason is as follows: As compound terms appear much more rarely in the document collection, their *idf* factor is much higher than simple words. As a consequence, if a compound term is identified in a query, it often plays a dominant role in the retrieval process. As compound terms only correspond to a part of the query contents, this means that this part is overstressed.

In order to better balance the weights of compound terms and single words, we separate the two types of element into two vectors. Each vector is assigned a relative importance I_j. In such a way, by assigning a lower importance to the vector of compound terms, we can create a better balance.

While the relative importance for the single-word vector is fixed at 1, we experimented with a series of values for the importance of the compound-term vector, 0.1, 0.2, …, and 1. The best figure is obtained when the compound-term vector is assigned an importance of 0.2-0.3. The following table shows the best results we obtained on the two test collections:

Table 4. Effectiveness of monolingual IR by separating compounds and words in two vectors.

	Traditional IR	TermDB + Exterm
English AP	0.2520	0.2827 (+12.2%)
French SDA	0.2356	0.3859 (+63.8%)

We can observe that with a reasonable assignment of relative importance, we can greatly improve the effectiveness of IR. In our case, the improvement for the French collection is particularly large (63.8% better). This may show that it is particularly important to recognize compound terms in French documents.

Globally, our experiments show that the detection of compound terms may greatly contribute in IR effectiveness. However, one also has to care about the way that compounds are used in combination with simple words. A naïve addition does not bring a significant impact. Significant impact may obtain with a more reasonable utilization of compounds.

4. Using compounds in cross-language IR

As we mentioned earlier, one of the effective approach to query translation for CLIR is the use of a statistical translation model trained on a large set of parallel texts. There are a few manually prepared parallel corpora. The best known is the Canadian Hansard, which contains the debates of the Canadian parliaments during 7 years, in both French and English. It contains dozens of millions words in each language. Such a parallel corpus is a valuable resource that contains word/term translations. The question is how to extract the translations from it. The training of a translation model aims to extract the translation relations between words in two languages.

The training of statistical translation model aims to obtain a probability function $P(t|s)$ that gives the probability of

[2] Retrieval effectiveness is measured in terms of average precision – a standard measure in IR [11]

translation of a source words *s* by a target word *t*. This is the result of IBM model 1 [1]. The training process is usually broken down into the following steps.

The first step segments parallel texts into sentences, and then to align sentences between the two languages [12]. A pair of aligned sentences means that one sentence is the translation of another. Note that beside the 1-1 alignment, there may also be 1-n and 1-0 alignments. However, the training of statistical model usually only considers the 1-1 alignments.

In our case, we use the IBM model 1. The training is based on the following principle (for a detailed description, see [1]):

We consider that a co-occurrence of a source word and a target word in a pair of aligned sentences as evidence of translation. Such evidence is gathered through all the alignments. The more the translation from one word to another is supported by such evidence, the higher it is assigned a probability. The final probabilities assigned should be such that maximizes the expectation of the given sentence alignments.

Concretely, the training is a process that repeats the following two steps:

- Assign an (initial) probability to each pair of words
- Using EM (Expectation Maximization) algorithm to maximize the expectation of the alignments. This algorithm iteratively modified the probability assignments so that the global expectation can be improved.

The resulting function $P(t|s)$ can be used directly for query translation in CLIR as follows:

- For each query word, we determine a set of target words with the highest probabilities;
- Among all the suggested translation words for the query, those with the highest probabilities are kept as the query "translation".

In our experiments, we keep the 30 best translations. This is not the most sophisticated and most principled utilization of the translation model, but it has been shown to be quite effective in our previous tests [10].

We observe that in the previous studies, parallel texts have usually been exploited to find translations between single words. The most obvious problem we can see is that by taking words one by one, many of them become ambiguous. The translation model will then suggest several translations corresponding to different meanings of the word. For example, the word "information" (in French) will have many possible translations because 1) the word denotes several meanings; 2) it appears very frequently in the parallel corpus. Among the possible translations, there are "information", "intelligence", "espionage", etc. However, if the term we intend to translate is "système d'information" (information system), and if the term is translated as a whole, then many of the meanings of "information" can be eliminated. The most probable translation of this term will be the correct term "information system". Through this example, we can see that a translation model that integrates the translation of compound terms can be much more precise. This is the goal of our utilization of compounds during query translation.

To do this, we have to train a translation model that incorporates compound terms as additional translation units to words. So compound terms are first extracted from the training parallel corpus, and added to the original sentences. Then the same translation process is launched. The resulting model contains now the translations for both single words and compound terms.

For the purpose of comparison, we also trained word-translation models (without compounds). The following table shows the CLIR results with both types of translation model:

Table 5. The CLIR effectiveness with different models.

	Word	Compounds (change)
F-E on AP	0.1465	0.2591 (+76.86%)
E-F on SDA	0.2257	0.2860 (+26.72%)

In this table, "F-E on AP" means that French queries are used to retrieve English documents in the AP collection.

Again the above results are obtained with two separate vectors to represent each document and query. The compound-term vector is assigned a relative importance of 0.3., while the word-vector is assigned 1.

We can see a great improvement in CLIR effectiveness once the translation model incorporates compound terms, especially for the F-E case.

Table 6 shows the comparisons with monolingual IR. In comparison with the traditional IR approach based on words, the CLIR using compound-term translation is even better. In particular, in the case of SDA, the difference is quite large. In comparison with the best performances we obtained on monolingual IR that uses compound terms, the percentage of the CLIR effectiveness is lower. This is normal. In particular, the SDA case represents a significant drop. The reason is that the monolingual IR on SDA has been boosted by the use of compound terms. It is difficult to catch up the same performance in CLIR. Nevertheless, the numbers shown in the third colon are comparable to the typical CLIR case, which is around 80% of that of monolingual IR effectiveness.

Table 6. Comparison with the monolingual effectiveness.

	Trad. Mono-IR	Mono-IR with compounds
F-E on AP	102.8%	91.7%
E-F on SDA	121.4%	74.1%

5. Conclusion

In this paper, we proposed to use compound terms in order to improve the precision of document and query representation. As a consequence, the retrieval effectiveness can also be improved.

Previous studies have suggested two approaches to identify compound terms from a text: using a manually

constructed dictionary of compounds, or using a syntactic/statistical analysis. However, the experiments have not always shown significant impact on IR effectiveness. We argue here that another important factor is the appropriate integration of compounds in the retrieval process. Different from the previous approaches, we proposed to separate compounds and single words in document and query representations, and assign a lower importance to the compound part in order to better balance their weights. This approach has been shown to be effective. On two text collections, the effectiveness of monolingual IR with compounds has been greatly improved.

For CLIR, we exploit a large set of parallel texts (the Hansard). In order to integrate compounds in query translation, we first extracted compound terms from the Hansard. The model trained on the modified Hansard naturally incorporates the translation of compound terms (in addition of that for single words). The translation accuracy is greatly improved. As a consequence, we observe significant improvements in CLIR effectiveness.

This preliminary study successfully shows the utility of compound terms in both monolingual IR and CLIR. We have shown that another key in using compound terms is their appropriate integration in IR process.

There are still several questions to be investigated. For example, we have not examined the impact of frequency threshold set in Exterm. The default threshold value (2) we used is not necessarily adapted to our task. Another question is the combination of the translations suggested by the translation model and those suggested by a bilingual dictionary (e.g. our terminology database). In our preliminary tests, this combination has not been found useful. However, it is too early to conclude on this. These problems will be further investigated in our future research.

References

[1] Brown, P. F., Pietra, S. A. D., Pietra, V. D. J., and Mercer, R. L., The mathematics of machine translation: Parameter estimation. *Computational Linguistics*, vol. 19, pp. 263-312, 1993.

[2] Buckley, C. *Implementation of the SMART information retrieval system*, Technical report, #85-686, Cornell University, 1985.

[3] Chiaramella Y. and Nie J.-Y., "A retrieval model based on an extended modal logic and its application to the RIME experimental approach," presented at *Research and Development on Information Retrieval - ACM-SIGIR Conference*, Brussels, pp. 25-43, 1990.

[4] Grefenstette, G. The Problem of Cross-Language Information Retrieval. In *Cross-language Information Retrieval*. Kluwer Academic Publishers. pages 1-9, 1998.

[5] Fagin, J., *Experiments in Automatic Phrase Indexing for Document Retrieval: A Comparison of Syntactic and Non-Syntactic methods*, PhD thesis, Computer Science, Cornell University, 1988.

[6] Foster, George F. *Statistical Lexical Disambiguation*, M.Sc thesis, McGill University, School of Computer Science, 1991.

[7] Franz, M., McCarley, J.S., Roukos, S., Ad hoc and multilingual information retrieval at IBM, *The Seventh Text Retrieval Conference (TREC-7),* NIST SP 500-242, pp. 157-168, 1998 (http://trec.nist.gov).

[8] Harman D. K. and E. M. Voorhees (eds.) *The Sixth Text REtrieval Conference (TREC-7).* Gaithersburg, NIST SP 500-242, 1998 (http://trec.nist.gov).

[9] Miller, G., Wordnet: an on-line lexical database, in *International Journal of Lexicography*, vol. 3, 1990.

[10] Nie, J.Y., Isabelle, P., Simard, M., Durand, R., Cross-language information retrieval based on parallel texts and automatic mining of parallel texts from the Web, *ACM-SIGIR conference*, Berkeley, CA, pp. 74-81, 1999.

[11] Salton, G. and McGill, M. J., *Introduction to Modern Information Retrieval*: McGraw-Hill, 1983.

[12] Simard, M., Foster, G., Isabelle, P., Using Cognates to Align Sentences in Parallel Corpora, *Proceedings of the 4th International Conference on Theoretical and Methodological Issues in Machine Translation*, Montreal , 1992.

[13] Sparck-Jones, K., Notes and references on early automatic classification work, *SIGIR Forum*, vol. 25, pp. 10-17,1991.

[14] Voorhees, E. M. Using Wordnet to disambiguate word senses for text retrieval. *Research and Development on Information Retrieval - ACM-SIGIR*, Pittsburgh, pp. 171-180, 1993.

[15] Yarowsky, D. Unsupervised word sense disambiguation rivaling supervised methods, *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pp. 189-196, Cambridge, MA., 1995.

Jian-Yun Nie received PhD from University of Grenoble, France. He is an associate professor in University of Montreal, Canada. His research area is information retrieval.

Jean-François Dufort is currently a B.Sci. student in University of Montreal. This work was carried out during summer 2001 when he obtained an Undergraduate Student Research Award from NSERC.