A probabilistic approach to crosslingual Information Retrieval

CIIR internal report

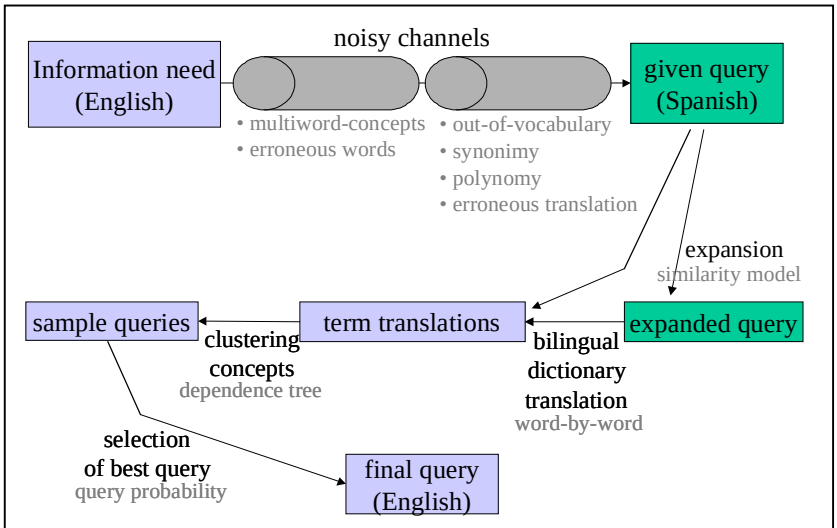
Philip Gröting June 2001

Abstract:

We present a method to translate queries from an arbitrary source language to retrieve documents in a destination language merely with easily obtainable instruments such as a machine readable dictionary and monolingual corpora in both languages. The key is to infer probabilistical information about the query and structuring the destination language terms accordingly. Though the results compare unfavourably with those obtained with more sophisticated but difficult to obtain IR-methods using Part-of-Speech-Tagging and/or Phrase dictionaries, our work shows the successful deployment and combination of related work to crosslingual Information Retrieval.

Introduction

Our model is based on a noisy-channel model, assuming that the users information need was expressed in the destination language and only through some noisy channel transformed into the source language. In fact, we consider the noisy channel to be split into two stages, one representing the loss of explicitness (information need to query) while the second represents the change of language, introducing among others errors through synonymy and polynomy. Accordingly, our method will decode the query to the expected information need in a two steps, of which each consists of two sub stages. The bilingual dictionary



graph 1: noisy channel concept

translation addresses the noise introduced by the second noisy channel. Additionally, though in the current work not successfully deployed, an expansion step could be taken here. Afterwards, an algorithm based on the dependence tree structure is used to decrease ambiguity, producing several sample queries. The last step is to choose the query which most probably represents the information need. The latter two steps address the noisy channel assumed to exist even in monolingual information retrieval, trying to identify multiword-concepts and filtering out erroneous words.

For simplicity we will refer throughout the remainder of this work to the destination language as English and the source language as Spanish. Therefore, the query (which represents incompletely the information need) is given in Spanish and has to be transformed such that it may separate relevant from irrelevant documents if passed on to a standard retrieval system, as was INQUERY in our experiments.

All experiments using INQUERY are based on the following two corpora: for English, the AP collection (years '88 through '90) within the TREC-9 corpus, consisting of 243,000 documents, was used. For Spanish the database ISM_ALL of 208 MB was used. Tests are performed using 21 queries from the TREC crosslingual IR set with provided relevance judgements. Using these corpora allowed for comparison with results obtained by Lisa Ballesteros [Ball98].

Step 1 A: Expansion

Optionally, a source language expansion step can be performed before the translation process. Implemented as a call to the INQUERY function get_modified_query with one of the given query words at a time, frequently occurring words, i.e. expected to be relevant to the query, are found. Those terms that occur more often than 10% with each other (an arbitrarily found threshold) are added to the query. This method reveals only few additional words, those found are in most cases very closely related, but on the other hand very biased towards relevant topics during the years '88 through '90. For example, the term *pope*[1] leads to the additional words *Vatican*, *II* and *Paul*, which should be useful for further disambiguation, if speaking about the pope. Unfortunately, these terms are also added, if in fact we meant *la papa*, the *potato*. Another problem is, once again, the short time period over which the corpus is created: the word *peace* is extended by *PLO*, *Israel*, *Palestinia*, *Bosnia* and *Serbia*, which is related during the years mentioned, but inappropriate if considering a longer time period.

[1] All words translated to English for understandability, although process works in Spanish

Statistics for expansion-step

Original Spanish terms	140		
Terms expanded	62	% of original terms	44%
Spanish terms added	152	Average # of expansions / term	2.45
English terms added	623	Avg. # new definitions / expanded term	10.0

We tried to use the words added during expansion in the disambiguation step in one of the following ways:

1. Using them like an original term, i.e., their English translation could occur in the final output, although this was made less likely by reducing their probability
2. Using them only as a bridge for matching other terms with its origin. An example for this would be that *pope* was not co-occurring with *tourism*, but *Vatican* did. Through this link our algorithm inferred that *pope* should be correlated with *tourism* as well.

Although the added terms were good expansions, i.e. adding meaningful words that are closely related to the term they are extending, the expansion step did not lead to the results expected. Contrarily, as the badly chosen example shows, it even decreased the precision by inferring links where a human would never expect one to exist, as between *pope* and *tourism*.

The result of the expansion compared with the non-expanded algorithm reads as follows:

Method	Non-expanded	Expanded I (adding to result)	Expanded II (only links)
Relevant retrieved:	609	414 (68.0%)	567 (93.1%)
Average precision:	0.1778	0.1424 (80.0%)	0.1392 (78.3%)

We mentioned the expansion step here for the sake of completeness, while the results shown below were created without the expansion step.

Step 1 B: Translation

Translation was performed word-by-word using a machine readable dictionary provided by Collins. All possible translations were taken into account, regardless of the word's part-of-speech. Thus, e.g. it is not possible to distinguish among the male word *el papa* (pope) and the female word *la papa* (potato). Additionally, the dictionary includes a large number of uncommon translations as they appear in sayings, so that *el papa* is also translated to *soft job* or stab and *la papa* gains the additional meaning *baby food*.

On average, one Spanish term is translated to 6.0 English terms. The key idea during this phase is to minimize bias against rare translations, to avoid ignoring a correct though uncommon translation whose terms have never been taken into account. Nevertheless, in order to decrease computational complexity, terms with an absolute document frequency of less than 10 times were filtered out[2]. Contrarily, if a Spanish term was not found in the dictionary, it could be included as English term, if it appeared in the English corpus, too. These words are most probably names or abbreviations, which still carry some information although used in another language.

In more detail, this step reads as follows: check for the occurrence of the unaltered word in the dictionary. If not successful, check for a small number of regularities to transform plural words into the singular form or to find the infinitive of conjugated verbs. Only if not even this search returns a result, the word to search for is stemmed, since this often injects errors, but in most of the cases finds possible translations.

As an example, consider searching for the word *paises*, which means countries. The given plural form of the word does not appear in the dictionary, so the end is stripped off, resulting in *pais*. This is found as country, while the stemmed version *pai* would also wrongly result in *pie*.

Step 2 A: Disambiguation

We developed different approaches concerning the disambiguation of words: (1) is based on building a complete dependence tree among all English terms as translated from the query terms, while (2) reduces the complexity by computing a dependence tree for the Spanish terms and searching for Mutual Information only among those terms, which were directly connected in the tree for the source language Spanish.

1. Complete Destination Language Dependence Tree

In a first step the EM-measure is computed for all pairs of terms, cf. [Ball98] for more detail:

$$em(a,b) = \max\{\frac{n(a,b) - \frac{n(a) * n(b)}{n}}{n(a) + n(b)}, 0\}$$

The components are as following: n is the number of documents in the corpus, n(a) the document frequency of term a, n(a,b) the co-occurrence of terms a and b. The co-occurrence statistics are derived by the INQUERY command evaluate_query(#uw100(a b)), thus the co-occurrence is measured over a text window

[2] 10 occurrences in a corpus of 243,000 documents, thus containing several million words, we consider as negligible

of 100 terms[3]. This formula, slightly differing from the more often used EMIM measure has the advantage of not being biased toward frequent terms.

A second step chooses a root node for the tree which is to be computed. Experiments showed that a good selection strategy is as simple as follows, if several possible translations can be computed: for each translation alternative choose consecutively the most frequent term within one category, which we define as all English terms derived from the same Spanish query term. That is, for alternative I, the English term translated from Spanish term 1 with the highest document frequency is chosen, in alternative II terms translated from Spanish term 2 are considered, etc. This method is based on the observation that correct translations almost always include at least one of the most frequent terms of one category.

In a third step we compute a dependence tree similar to the approach proposed by van Rijsbergen [vR77]. As algorithms to model the tree we used exchangeably the Minimal Spanning Tree algorithm by Whitney[4] and a simpler greedy approach. Both were shown to lead to equal results, given problems with a depth of less than 100 nodes (i.e. English terms) and relatively sparse EM-information. It is important to note that about half of the term pairs did not possess any co-occurrence, so that it is not uncommon that several words did not appear in the tree, due to the lack of any connection to another word.

Finally, the tree is pruned back, so that any translation of a category is dismissed, as long as there remains at least one other translation and the node in the tree is not necessary to connect the root with the last occurrence of another category. This ultimately results in the possibility of including several translations for one Spanish term. These multiple translations are later combined with the INQUERY #syn operator.

This algorithm is summarized in figure 1. In the following we refer to query terms in the source language as *qs* and to query terms in the target language as *qt(qs)* or shorter as *qt*, if the source term, from which the translation is taken is of no concern.

[3] experimentally, other window sizes of 50 and 250 terms have been tested without changing the results significantly

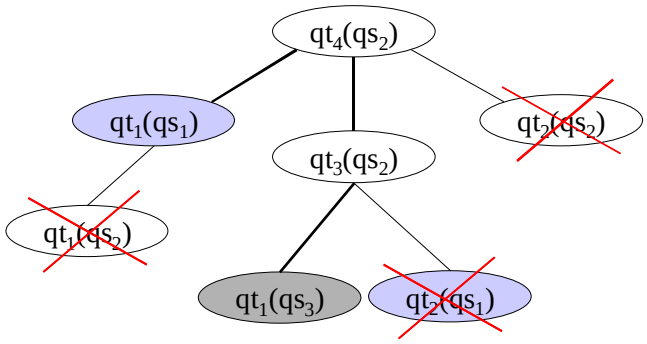
[4] pseudocode free of charge available on the internet

Algorithm "Disambiguation with destination-language dependence tree":

```
for each query word qs1, qs2, ...
translate to qt1(qs1), qt2(qs1), ..., qt1(qs2), ...
for each word pair (qt1(qs1), qt1(qs2)) compute EM co-occurrence
for each qs {
    root = qti(qsk), such that n(qti(qsk)) > n(qtj(qsk))
    compute dependence_tree(root)
    prune tree bottom to top as long as categories are not deleted
    give out nodes remaining
}
```

figure 1: algorithm for complete destination language dependence tree

Example:



The dependence tree depicted on the left will be pruned (marked by the crossed out terms) and eventually lead to the query „$qt_1(qs_1)$ #syn($qt_3(qs_2)$ $qt_4(qs_2)$) $qt_1(qs_3)$".
Thus, two terms from category 2 are used, because $qt_3(qs_2)$ cannot be pruned without loosing a translation of qs_3.

Figure 2: example for pruning a dependence tree according to categories

2. Using Dependence Tree of Source Language

Yet another chosen approach does not compute the full dependence tree for the destination language, but rather uses information gained by processing the query to decrease the complexity in the destination language computations. Thus, the steps performed are:

(1) Building a dependence tree for the Spanish source query, using the same mechanism described in the preceding paragraph. Since on average the problem size for the dependence tree for Spanish is 5.6 words, while for the English we face an average problem size of 33.6 words, we can assume that the probability of asserting wrong interdependencies is much smaller regarding the Spanish query.

(2) The knowledge gained out of the first step can then be used to search the limited space of those English terms whose Spanish origins have been found interconnect ed, i.e. shared a link in the dependence tree. As an example, in the query *automóviles de energía solar* (cars powered by solar energy) the term *energía* is translated to *energy*, *drive*, *power* and *current*. While *drive* and automobile co-occur most often, the term

energy is better. This translation can be found, when the Spanish dependence tree is taken into account, in which *energía* is associated with *solar*, and *energy* co-occurs more often with *solar* than *drive*.
(3) Since not all terms can be added to the tree with this method (often there are no co-occurrences among the terms from the categories associated with each other in the Spanish dependence tree), we attempt to add terms for the other categories using the English dependence-tree algorithm described before and finally with a backup method that adds the terms with the highest document frequency, if no co-occurrence is significant.

Step 2 B: Selection by query probability

Following the computation of sample queries, one for each non-stopword in the original query, one has to be chosen as result. The query probability P(Q) is computed by a mixture model, with the two components conditional probability of node given the parent node and individual probability, which is the best estimate for that word, given all possible translations within the respective category:

$$P(Q) = \prod_{qt(qs)} (\lambda * \frac{n(qt_i \wedge qt_j)}{n(qt_j)} + (1-\lambda) * \frac{n(qt_i)}{\sum_{i'} n(qt_{i'})})$$

s.t. (1) qt_j is parent node of qt_i
(2) all qs are covered; note: for simplicity in the formula qt was written instead of qt(qs)
(3) $qt_{i'}$ are all English terms from the same category as qt_i

This approach needs refinement, since it could be shown experimentally that taking always the first of he sample queries led to slightly higher results than selecting by the highest query probability P(Q).

Giving individual weight to terms

The INQUERY system allows with the #wsum function to assign individual weight to each of the terms in the query. Passing the respective term-probability, being equal to one component within the multiplicative term of the P(Q) formula as shown in the preceding paragraph, results in higher recall and reduced precision. No significance tests have been done yet, but experimentally results of 5-10% increased recall and a only slightly reduced precision (less than 1% difference) can be reported and suggest to make use of this technique.

Results

Although we can find several examples in which the described algorithms are promising, the overall performance is poor. Table 1 summarizes recall and precision results of the two disambiguation techniques described in this paper as well as two baselines for comparison. The first column shows a crosslingual baseline, for which the Spanish queries were translated with the method described under step 1B and then tagged together with #syn-operators including all words within one category. The two following columns represent the techniques shown in step 2A: the English dependence-tree algorithm and the algorithm also taking into account the Spanish dependence-tree. The rightmost column is a monolingual baseline; here English queries (manually translated from the Spanish source) were run on the INQUERY retrieval system.

Our first technique, the English dependence tree algorithm, achieves a slight gain over the crosslingual baseline. Taking additionally the Spanish dependence tree increases the result to 56% of the monolingual result, but is still worse than the 60% Lisa Ballesteros reported for word-by-word translation using a part-of-speech tagger and the synonym operator. Compared to the results presented in [Ball98], which achieved 79% of monolingual retrieval by means of a phrase dictionary and co-occurrence statistics, our results cannot compete.

	baseline #syn(all)	***English dependence***	***Spanish dependence***	***mono-lingual***
Relevant retrieved	498	510	609	972
% of optimal	*51,2*	*52,5*	*62,7*	
% over baseline		*2,4*	*22,3*	*95,2*
Interpolated Recall - Precision				
at 0.00	0.3719	0.4007	0.4361	0.7356
at 0.10	0.2801	0.3114	0.2956	0.6294
at 0.20	0.2379	0.2387	0.2540	0.5347
at 0.30	0.2119	0.2020	0.2229	0.4410
Average precision (non-interpolated) over all rel docs				
	0.1612	0.1700	0.1778	0.3125
% of optimal	*51,6*	*54,4*	*56,9*	
% over baseline		*5,5*	*10,3*	*93,9*
Precision:				
5 docs:	0.2571	0.2762	0.2952	0.5238
10 docs:	0.2619	0.2619	0.2571	0.4905
100 docs:	0.1195	0.1157	0.1467	0.2619
500 docs:	0.0371	0.0379	0.0469	0.0831
1000 docs:	0.0237	0.0243	0.0290	0.0463
R-Precision (precision after R (= num_rel for a query) docs retrieved):				
Exact:	0.1968	0.1851	0.1921	0.3298

Table 1: experimental results

Discussion of results

The poor results in practice may partly be explicable with some deficiencies of the English database:

1. Our database consists of American English while the dictionary is British English. Therefore, e.g. the Spanish word *basura* is translated to *rubbish* and *garbage*, which hardly occur in the database. There the words *trash* or *junk* would be found, but do not exist in the dictionary.
2. Some common term combinations one would expect to identify easily are in fact not detected: *segunda guerra mundial* should be translated to *second world war*, but this combination of words is uncommon, more often *World War II* is used – and the algorithm fails. Additionally, due to the short period of only two years from which the database documents are collected, the bias towards topics relevant during that topic seems to be a greater constraint than expected: during the years '88 – '90 it seems as if the second world war was not much a topic, so that even the terms *world* and *war* do not show a high EM-score. Another example is demonstrated by the query the *popes journey to Mexico*. Since the pope did not visit Mexico during the specified years, there is no co-occurrence of the terms, and the algorithm heads into another direction: *tourism* as an alternative translation to *journey* occurs more often with *Mexico*, as do *potatoes*, which are, as mentioned before, an alternative to *pope*, due to a lacking part-of-speech recognition. Therefore the *viaje del papa a mexico* becomes <*potato tourism Mexico*>.

Acknowledgement

This material is based on work supported in part by the Library of Congress and Department of Commerce under cooperative agreement number EEC-9209623 and in part by SPAWARSYSCEN-SD grant number N66001-99-1-8912. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsor.

References

[Ball96] Lisa Ballesteros and Bruce Croft. "Dictionary methods for crosslingual IR"

[Ball98] Lisa Ballesteros and Bruce Croft. "Resolving ambiguity for cross-language Retrieval"

[Ponte98] Jay Ponte and Bruce Croft. "A language modelling approach to Information Retrieval"

[Brown90] Peter Brown et al. "A statistical approach to Machine Translation"

[Berger99] Adam Berger and John Lafferty. "Information Retrieval as statistical translation"

[Carb97] Jaime Carbonell et al. “Translingual Information Retrieval: A comparative evaluation”

[Ball00] Lisa Ballesteros. “Cross-language retrieval via transitive translation”

[vR77] C.J. van Rijsbergen. “Information Retrieval” (especially chapter 6)