ITC-irst at CLEF 2002: Using N-best query translations for CLIR

Nicola Bertoldi and Marcello Federico

ITC-irst - Centro per la Ricerca Scientifica e Tecnologica
I-38050 Povo, Trento, Italy.

Abstract

This paper reports on the participation of ITC-irst in the Italian monolingual retrieval track and in the bilingual English-Italian track of the Cross Language Evaluation Forum (CLEF) 2002. A cross-language information retrieval systems is proposed which integrates retrieval and translation scores over the set of N-best translations of the source query. Translations are computed by a statistical translation model, based on an hidden Markov model, and trained over a bilingual dictionary and the target document collection. Retrieval scores result as a combination of a statistical language model and a standard Okapi model.

1. Introduction

This paper reports on the participation of ITC-irst in two Information Retrieval (IR) tracks of the Cross Language Evaluation Forum (CLEF) 2002: the monolingual retrieval task, and the bilingual retrieval task. The language of the queries was Italian for the monolingual track and English for the bilingual track; Italian documents were searched in both tracks. With respect to the 2001 CLEF evaluation (Bertoldi and Federico, 2002), the Cross Language IR (CLIR) system was modified in order to work with multiple translations of queries, and with source and target languages in the reverse order.

The basic IR engine, used for both evaluations, combines scores of a standard Okapi model and of a statistical language model. For CLIR, a light-weight statistical model for translating queries was developed, which also computes the list of N-best translations for each query. In this way, the basic IR engine is used to integrate retrieval and translation scores over multiple translations (Federico and Bertoldi, 2002). Remarkably, training of the system just requires a bilingual dictionary and the target document collection.

This paper is organized as follows. Section 2 introduces the statistical approach to CLIR. Sections 3 to 5 describe, respectively, the query-document model, the query-translation model, and the CLIR algorithm. Section 6 presents and discusses experimental results.

2. Statistical CLIR Approach

From a statistical perspective, the CLIR problem can be formulated as follows. Given a query $\mathbf{f}$ in the source language, by convention French, one would like to find relevant documents d in the target language, English, within a collection $\mathcal{D}$. More formally, documents should be ranked according to the posterior probability:

$$\Pr(d \mid \mathbf{f}) \propto \Pr(\mathbf{f}, d) \tag{1}$$

where the right term of formula (1) follows from the application of Bayes formula and from the constancy of $\Pr(\mathbf{f})$ with respect to the ranking of documents.

To fill the language difference between query and documents, the hidden variable $\mathbf{e}$ is introduced, which represents an English (term-by-term) translation of $\mathbf{f}$. Hence, the following decomposition is derived:

$$\begin{aligned} \Pr(\mathbf{f}, d) &= \sum_{\mathbf{e}} \Pr(\mathbf{f}, \mathbf{e}, d) \\ &\approx \sum_{\mathbf{e}} \Pr(\mathbf{f}, \mathbf{e}) \Pr(d \mid \mathbf{e}) \\ &= \sum_{\mathbf{e}} \Pr(\mathbf{f}, \mathbf{e}) \frac{\Pr(\mathbf{e}, d)}{\sum_{d'} \Pr(\mathbf{e}, d')} \end{aligned} \tag{2}$$

In deriving formula (2), one makes the assumption (or approximation) that the probability of document d given query $\mathbf{f}$ and translation $\mathbf{e}$, does not depend on $\mathbf{f}$. Moreover, the main summation in (2) is taken over the set of possible translations of $\mathbf{f}$. As terms of $\mathbf{f}$ may typically admit more than one translation, the size of this set can grow exponentially with the length of $\mathbf{f}$. Finally, the denominator in formula (2) requires summing over all document in $\mathcal{D}$ and should be computed for every possible translation $\mathbf{e}$.

In the following, statistical models will be described which, through suitable approximations, permit to efficiently compute formula (2). In particular, we will show how:

- probability $\Pr(\mathbf{e}, d)$ is computed by the query-document model,
- probability $\Pr(\mathbf{f}, \mathbf{e})$ is computed by the query-translation model,
- formula (2) is computed and documents are ranked by the CLIR algorithm.

Q, T, D	random variables of query, translation, and document
$\mathbf{f}, \mathbf{e}, d$	instances of source query, query translation, and document
w, f, e	generic term, term in the source language, term in the target language
$\mathcal{D}$	collection of documents
$\mathcal{V}$,$\mathcal{V}(d)$	set of terms occurring in $\mathcal{D}$, and in document d
$N, N(d)$	number of term occurrences in $\mathcal{D}$, and in a document d
$N(w), N(d,w), N(\mathbf{e},w)$	frequency of term w in $\mathcal{D}$, in document d, and in query $\mathbf{e}$
N_w	number of documents in $\mathcal{D}$ which contain term w
$\lvert\cdot\rvert$	size of a set

Table 1: List of often used symbols.

3. Query-Document Model

The query-document model computes the joint probability of a query $\mathbf{e}$ and a document d, written in the same language. Two query-document models were considered in the experiments. The former is based on a statistical model, the latter on the standard Okapi scoring function.

3.1. Witten-Bell Query-Document Model

The joint probability of a query $\mathbf{e}$ and a document d can be factored out as follows:

$$\Pr(\mathbf{e}, d) = \Pr(\mathbf{e} \mid d)\Pr(d) \tag{3}$$

where $\Pr(\mathbf{e} \mid d)$ represents the likelihood of $\mathbf{e}$ being generated by d, and $\Pr(d)$ the a-priori probability of d. In the following, no a-priori knowledge about the documents will be assumed, hence a uniform *a-priori* distribution is taken[1]. For what concerns the probability of $\mathbf{e}$ given d, an order-free multinomial (bag-of-word) model is assumed. Hence, assuming $\mathbf{e} = e_1, \ldots, e_n$, we have[2]:

$$\Pr(\mathbf{e} = e_1, \ldots, e_n \mid d) = \prod_{k=1}^{n} p(e_k \mid d) \tag{4}$$

The probability of term q being generated by document d is estimated by the statistical LM:

$$p(e \mid d) = \lambda \frac{N(d,e)}{N(d)} + (1-\lambda)\, p(e) \tag{5}$$

where $p(e)$, the word probability over $\mathcal{D}$, is estimated by interpolating the smoothed relative frequency with the uniform distribution over the vocabulary $\mathcal{V}$ of $\mathcal{D}$:

$$p(e) = \mu \frac{N(e)}{N} + (1-\mu)\frac{1}{|\mathcal{V}|} \tag{6}$$

Parameters λ and μ are estimated according to (Witten and Bell, 1991).

[1]However, this model permits to apply any available prior distribution on documents.

[2]Notice the use of $p(\cdot)$ to indicate a probability computed by a statistical model.

3.2. Okapi Query-Document Model

The query-document model can also be based on a generic scoring function $s(\mathbf{e}, d)$. In order to obtain a distribution over queries and documents scores have to be normalized as follows:

$$\Pr(\mathbf{e}, d) = \frac{s(\mathbf{e}, d)}{\sum_{\mathbf{e}', d'} s(\mathbf{e}', d')} \tag{7}$$

The denominator of the above formula is considered only for the sake of normalization, but can be disregarded in the computation of equation (2).
In the experiments the following scoring function was used, whose logarithm corresponds to the standard Okapi formula:

$$s(\mathbf{e} = e_1, \ldots, e_n, d) = \prod_{k=1}^{n} idf(e_k)^{W_d(e_k)} \tag{8}$$

where:

$$W_d(w) = \frac{N(d,w)(k_1+1)}{k_1(1-b) + k_1 b \frac{N(d)}{l} + N(d,w)} \tag{9}$$

scores the relevance of w in d, and:

$$idf(w) = \frac{N - N_w + 0.5}{N_w + 0.5} \tag{10}$$

is the inverted document frequency. As in previous work, the setting $k_1 = 1.5$ and $b = 0.4$ were used. An explanation of the involved terms can be found in (Robertson et al., 1994) and other papers referred in it.

3.3. Combined Query-Document Model

Previous work (Bertoldi and Federico, 2001) showed that Okapi and the statistical model rank documents almost independently. Hence, information about the relevant documents can be gained by integrating the scores of both methods. Combination of the two models is implemented by just taking the sum of scores. Actually, in order to adjust scale differences, scores of each model are normalized in the range $[0, 1]$ before summation. The resulting query-document model was also applied to the monolingual IR track.

4. Query-Translation Model

The query-translation model computes the probability of any query-translation pair. This probability is modelled by an HMM (Rabiner, 1990) in which the observable variable is the Italian query $\mathbf{f}$, and the hidden variable is its English translation $\mathbf{e}$. According to the HMM, the joint probability of a pair $(\mathbf{f}, \mathbf{e})$ is decomposed as follows:

$$\begin{aligned} Pr(\mathbf{f} = f_1, \ldots, f_n, \mathbf{e} = e_1, \ldots, e_n) \\ = p(e_1) \prod_{k=2}^{n} p(e_k \mid e_{k-1}) \prod_{k=1}^{n} p(f_k \mid e_k) \end{aligned} \quad (11)$$

Formula (11) puts in evidence two different conditional probabilities: the term translation probabilities $p(f \mid e)$ and the target LM probabilities $p(e \mid e')$.

Given a query-document model and a query $\mathbf{f}$, the most probable translation $\mathbf{e}^*$ can be computed through the well known Viterbi search algorithm, while N-best translations of $\mathbf{f}$ can be computed with the tree-trellis based algorithm (Federico and Bertoldi, 2002).

Probabilities $p(f \mid e)$ are estimated from a bilingual dictionary as follows:

$$\Pr(f \mid e) = \frac{\delta(f, e)}{\sum_{f'} \delta(f', e)} \quad (12)$$

where $\delta(f, e) = 1$ if the term e is one of the translations of term f and $\delta(f, e) = 0$ otherwise. In Section 6, it will be explained how out-of-dictionary words are processed.

Probabilities $p(e \mid e')$ are estimated on the target document collection, through an order-free bigram LM, which tries to compensate for different word positions induced by the source and target languages. Let

$$p(e \mid e') = \frac{p(e, e')}{\sum_{e''} p(e'', e')} \quad (13)$$

where $p(e, e')$ is the probability of e co-occurring with e', regardless of the order, within a text window of fixed size. Smoothing of this probability is performed through absolute discounting and interpolation as follows:

$$p(e, e') = \max\left\{\frac{C(e, e') - \beta}{N}, 0\right\} + \beta p(e) p(e') \quad (14)$$

where $C(e, e')$ is the number of co-occurrences appearing in the corpus, $p(e)$ is computed according to equation (6), and the absolute discounting term β is equal to the estimate proposed in (Ney et al., 1994). Absolute discounting was chosen for its good performance and suitability to the order-free case.

5. CLIR Algorithm

The CLIR algorithm is in charge of computing formula (2) and sorting documents according to the posterior probability (1). The algorithm relies on two approximations in order to limit the set of possible translations and documents to be taken into account:

- external summation is taken over $\mathcal{T}_N(\mathbf{f})$, the set of the N-best translations of $\mathbf{f}$, and
- internal summation is over $\mathcal{I}(\mathbf{e})$, the set of documents in $\mathcal{D}$ containing at least a word of $\mathbf{e}$.

This corresponds to approximating formula (2) by:

$$\Pr(\mathbf{f}, d) \approx \sum_{\mathbf{e} \in \mathcal{T}_N(\mathbf{f})} \Pr(\mathbf{f}, \mathbf{e}) \frac{\Pr(\mathbf{e}, d)}{\sum_{d' \in \mathcal{I}(\mathbf{e})} \Pr(\mathbf{e}, d')} \quad (15)$$

1. Input $\mathbf{f}$
2. Compute $\mathcal{T}_N(\mathbf{f})$ and scores $P[\mathbf{f}, \mathbf{e}]$
3. For each $\mathbf{e} \in \mathcal{T}_N(\mathbf{f})$
4. $\quad N = 0$
5. $\quad$ For each $d \in \mathcal{I}(\mathbf{e})$
6. $\quad\quad$ Compute $P[\mathbf{e}, d]$
7. $\quad\quad$ Update $N = N + P[\mathbf{e}, d]$
8. $\quad$ For each $d \in \mathcal{I}(\mathbf{e})$
9. $\quad\quad$ Update $P[d] = P[d] + P[\mathbf{e}, d] * P[\mathbf{f}, \mathbf{e}]/N$
10. Order documents according to $P[d]$

Table 2: CLIR algorithm.

The CLIR algorithm is illustrated in Table 2. Briefly, given an input query $\mathbf{f}$, the N-best translations $\mathcal{T}_N(\mathbf{f})$ and their probabilities $P[\mathbf{f}, \mathbf{e}]$ are computed first. Then, for each translation $\mathbf{e}$, the addenda in formula (2) are computed only for documents containing at least one term of $\mathbf{e}$. This requires one additional loop over the documents in order to compute the normalization term.

6. Experimental Evaluation

Four runs were submitted to CLEF 2002: one for the Italian monolingual track (`IRSTit1`) and 3 for the bilingual English-to-Italian track, using 1-best, 5-best, and 10-best translation (`IRSTen2it1`, `IRSTen2it2`, and `IRSTen2it3`), respectively. The tracks consisted of 49 topics, for a total of 1072 documents to be retrieved, inside a collection of 108,578 Italian newspaper article from *La Stampa* and *Swiss News Agency*, both of 1994. All runs used only title and description parts of the topics.

6.1. Preprocessing

Text preprocessing was applied on the target documents before indexing, and on the queries before retrieval. More specifically, the following preprocessing steps were carried out:

- *Tokenization* was performed on documents and queries to isolate words from punctuation marks, to recognize abbreviations and acronyms, correct possible word splits across lines, and discriminate between accents and quotation marks.
- *Base forms* were computed for Italian words by means of morphological analysis and POS tagging.
- *Stemming* was performed on English words by using the Porter's algorithm (Frakes and Baeza-Yates, 1992).
- *Stop-terms removal* was applied on the documents by removing terms with a low inverted document frequency (Frakes and Baeza-Yates, 1992).
- *Proper names and numbers* in the query were recognized in order to improve coverage of the dictionary.
- *Out-of-dictionary terms* which have not been recognized as proper names or numbers were removed from the query.

6.2. Blind Relevance Feedback

After document ranking, Blind Relevance Feedback (BRF) can be applied. BRF is a well known technique that allows to improve retrieval performance. The basic idea is to perform retrieval in two steps. First, the documents matching the source query $\mathbf{e}$ are ranked, then the B best ranked documents are taken and the R most relevant terms in them are added to the query, and the retrieval phase is repeated. In the CLIR framework, R terms are added to each single translation of the N-best list and the retrieval algorithms is repeated once again.
In this work, new search terms are selected from the top B documents according to:

$$r_w \frac{(r_w + 0.5)(N - N_w - B + r_w + 0.5)}{(N_w - r_w + 0.5)(B - r_w + 0.5)} \quad (16)$$

where r_w is the number of documents, among the B top documents, which contain term w. In all the performed experiments the values $B = 5$ and $R = 15$ were used (Bertoldi and Federico, 2001).

6.3. Official Results

Table 6.3. reports official results of the submitted runs, and Table 6.3. compares them with the worst, median, and best results of competitors.
With respect to previous evaluations, this year we had to perform query translation in the reverse order, i.e. from English to Italian. Given the large gap between monolingual and cross-lingual results, i.e. about 15% absolute, future work will be devoted to investigate possible weakness in the preprocessing phase, poor coverage of the dictionary with respect to the terms in the queries, and any other possible causes of poor translation.

Official Run	N-best	mAvPr
IRSTit1		.4920
IRSTen2it1	1	.3444
IRSTen2it2	5	.3531
IRSTen2it3	10	.3552

Table 3: Results of the official runs.

Official Run	$<$ mdn	$>$ mdn	wrs	bst
IRSTit1	11	37	0	7
IRSTen2it1	21	24	3	5
IRSTen2it2	19	26	2	2
IRSTen2it3	16	26	2	6

Table 4: Results of the official runs against the worst, median and best values.

7. References

Bertoldi, N. and M. Federico, 2001. ITC-irst at CLEF 2000: Italian monolingual track. In Peters (ed.), *Cross-Language Information Retrieval and Evaluation*, vol. 2069 of *LNCS*, pages 261–272. Springer Verlag.

Bertoldi, N. and M. Federico, 2002. ITC-irst at CLEF 2001: Monolingual and bilingual tracks. In Peters, et al. (eds.), *Cross-Language Information Retrieval and Evaluation*, vol. 2406 of *LNCS*, pages 94–101. Springer Verlag.

Federico, M. and N. Bertoldi, 2002. Statistical cross-language information retrieval using n-best query translations. In *Proc. of 25th International ACM SIGIR*. Tampere, Finland.

Frakes, W. B. and Ricardo Baeza-Yates (eds.), 1992. *Information Retrieval: Data Structures and Algorithms*. Prentice Hall.

Ney, H., U. Essen, and R. Kneser, 1994. On structuring probabilistic dependences in stochastic language modelling. *Computer Speech and Language*, 8:1–38.

Rabiner, L. R., 1990. A tutorial on hidden Markov models and selected applications in speech recognition. In Weibel and Lee (eds.), *Readings in Speech Recognition*, pages 267–296. Morgan Kaufmann.

Robertson, S. E., S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford, 1994. Okapi at TREC-3. In *Proc. of 3rd TREC*. Gaithersburg, MD.

Witten, I. H. and T. C. Bell, 1991. The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression. *IEEE Trans. Inform. Theory*, IT-37(4):1085–1094.