XRCE's Participation to CLEF 2007 Domain-specific Track

Stephane Clinchant and Jean-Michel Renders
Xerox Research Centre Europe, 6 ch. de Maupertuis, 38240 Meylan, France
FirstName.LastName@xrce.xerox.com

Abstract

Our participation to CLEF07 (Domain-specific Track) was motivated this year by assessing several query translation and expansion strategies that we recently designed and developed. One line of research and development was to use our own Statistical Machine Translation system (called Matrax) and its intermediate outputs to perform query translation and disambiguation. Our idea was to benefit from Matrax' flexibility to output more than one plausible translations and to train its Language Model component on the CLEF07 target corpora. The second line of research consisted in designing algorithms to adapt an initial, general probabilistic dictionary to a particular pair (query, target corpus); this constitutes some extreme viewpoint on the "bilingual lexicon extraction and adaptation" topic that we are investigating since now more than 6 years. For this strategy, our main contributions lie in a pseudo-feedback algorithm and an EM-like optimisation algorithm that realize this adaptation. A third axis was to evaluate the potential impact of "Lexical Entailment" models in a cross-lingual framework, as they were only used in a monolingual setting up to now. Experimental results on CLEF-2007 corpora (domain-specific track) show that the dictionary adaptation mechanisms appear quite effective in the CLIR framework, exceeding in certain cases the performance of much more complex Machine Translation systems and even the performance of the monolingual baseline. In most cases also, Lexical Entailment models, used as query expansion mechanisms, turned out to be beneficial.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: H.3.1 Content Analysis and Indexing; H.3.3 Information Search and Retrieval; H.3.4 Systems and Software; H.3.7 Digital Libraries; H.2.3 [**Database Managment**]: Languages—*Query Languages*

General Terms

Measurement, Performance, Experimentation

Keywords

Domain-specific IR, Lexicon Extraction, Query Translation and Disambiguation, Dictionary Adaptation

1 Introduction : Query Translation and Disambiguation

We can distinguish at least two families to perform query translation. The first one is to use Machine Translation systems (such as Babylon, Systran, etc.); the second one is to rely on multilingual

dictionaries or lexicons. Machine Translations systems aims at translating a source sentence into a target sentence. MT systems are built to produce well formed grammatical sentences. However, most information retrieval models (or user's queries) do not rely today on proper syntax: this is the bag of words hypothesis. A query is a set of terms and no use is made about the order or the syntax in the query, if it exists. One need not translate properly the query into a correct sentence, a rough term-to-term translation can be sufficient to capture the concept of a query. Hence, term-to-term translations rely on bilingual dictionaries and cross-lingual information retrieval has been concerned with the extraction of bilingual dictionaries on the one hand, and with algorithms to obtain the best translation of a query from a dictionary on the other hand.

The first and naive use of a dictionary, is to use all translations — possibly weighted — of a query word. Albeit simple, this approach does not address the *polysemy* of words. A classical example is the translation of the english word *bank*. Bank can refer either to a financial institution or to the edge of a river. Choosing the right translation of a query term can be obvious with the context of the complete query. If one was to translate the word *bank* in a query and also observe the word *account*, then the translation is no longer ambiguous. Note though that the retrieval process is a disambiguating process in itself, in that spurious translations are generally filtered out simply by the fact that it is very unlikely that they co-occur with other translations. Several approaches [15, 12, 8, 13, 14, 9] resolve the translation of query with the notion of *coherence*. Each query term has candidate translation terms and a co-occurrence statistics can be computed between all the candidate translation terms; then an optimisation algorithm is used to solve some maximum coherence problem. The idea is that the query defines a lexical field. The more likely a candidate belongs to the lexical field, the better it is for translation.

2 Cross-lingual Information Retrieval and Language Modelling

We will first introduce the standard monolingual language modeling approach to information retrieval. Then, we will present the classical extensions to cross-lingual information retrieval.

The core idea of language models is to determine the probability $P(q|d)$ — the probability that the query would be generated from a particular document. Formally, given a query q, the language model approach to IR [17] scores documents d by estimating $P(q|d)$, the probability of the query according to some language model of the document. Using some independence assumption, for a query $q = \{q_1, \dots q_\ell\}$, we get:

$$P(q|d) = \prod_{i=1}^{\ell} P(q_i|d). \tag{1}$$

We assume that for each document there exists some parameter θ_d, which is a probability distribution over words — a language model. Abusively we note $P(q|d) \equiv P(q|\theta_d)$. Standard language models in information retrieval are multinomial distributions : the language model of a document is defined by its parameter vector θ_d, whose dimension is the size of the vocabulary. As this multinomial parameter is normalized (the sum of its components sums up to one), another notation is used : $\theta_{dw} = P(w|d)$.

For each document d, a simple language model could be obtained by considering the frequency of words in d, $P_{ML}(w|d) \propto \#(w,d)$ (this is the Maximum Likelihood, or ML, estimator). The probabilities are smoothed by the corpus language model $P_{ML}(w|\mathcal{C}) \propto \sum_d \#(w,d)$. The resulting language model is:

$$P(w|d) = \lambda\, P_{ML}(w|d) + (1-\lambda)\, P_{ML}(w|\mathcal{C}). \tag{2}$$

The reasons of smoothing are twofold: first a word can be present in a query but absent in a document. However this fact does not make it impossible and the document should give it a probability. The second reason is to play a role like the Inverse Document Frequency. Smoothing enables implicitly to renormalize the frequency of one word in a document with respect to its occurrence in the corpus. Others smoothing methods could be applied (Dirichlet smoothing,

Absolute Discounting, ...) and can be found in [20]. The *Query Likelihood* approach above gives an intuitive view of how language models works in information retrieval. Others equivalent ranking functions can be considered and lead to the same ranking function as the *Query Likelihood* formulation. For example the *KL-divergence* and the *Cross-Entropy* functions can also be used in information retrieval. Let θ_q be a multinomial parameter for the language model of a query q, θ_d the language model for a document d, the cross-entropy function between these two objects is:

$$CE(\theta_q|\theta_d) = \sum_w P(w|q)\log(P(w|d)) = \sum_w \theta_{qw}\log(\theta_{dw}) \tag{3}$$

As far as cross-lingual IR is concerned, the core idea remains the same: modeling the probability of the query given the document. Let q_s be the query in some source language, w_s a word in the source language, d_t a document in the target language, w_t a word in the target language, $P(w_t|w_s)$ the probability that word w_s is translated into w_t. We can distinguish two methods:

The first method, we will refer as **CL_LM1**, translates the query into a query language model in the target language [12]. Then a monolingual search is performed, using a ranking criterion such as the Cross-Entropy:

$$\begin{aligned} CE(q_s|d_t) &= \sum_{w_t} P(w_t|q_s)\log P(w_t|d_t) \\ &= \sum_{w_t,w_s} P(w_t|w_s,q_s)P(w_s|q_s)\log P(w_t|d_t) \\ &\cong \sum_{w_t,w_s} P(w_t|w_s)P(w_s|q_s)\log P(w_t|d_t) \end{aligned} \tag{4}$$

The second model, we will refer as **CL_LM2**[2, 11], models the translation from the document side : a language model of the document is built in the source language and compared to the query:

$$\begin{aligned} CE(q_s|d_t) &= \sum_{w_s} P(w_s|q_s)\log P(w_s|d_t) \\ &\cong \sum_{w_s} P(w_s|q_s)\log(\sum_{w_t} P(w_s|w_t)P(w_t|d_t)) \end{aligned} \tag{5}$$

Both models are based on probabilistic dictionaries, but the first model uses a dictionary from source language to target language, whereas the second model uses a dictionary from target to source. In CL_LM1, the translation process in independent of the document, whereas, in CL_LM2, one tries to model the probability that a particular document is translated and "distilled" into the original query.

In the following part of this report, we will adopt the viewpoint of model CL_LM1 for two reasons: first it is simpler to use because it just requires a monolingual retrieval system, unlike CL_LM2 which need a devoted cross-lingual system. The second reason is a benchmarking one : we wanted to compare our results with Machine Translation tools, which operate in that direction (translate the query from source to target) for obvious practical reasons.

3 Dictionary Adaptation

The main idea of dictionary adaptation is to be able to adapt the entries of a dictionary to a query and a target corpus. Formally, let $q_s = (w_{s1}, \ldots, w_{sl})$ be the query in source language. Ideally, we are looking for $P(w_t|q_s)$, the probability of a target term given the source query. As we adopt the **CL_LM1** model, this leads us to focus on $P(w_t|w_s, q)$, which is the probability that source term w_s tranlates to w_t, given the context of the query. Computing this probability would need

to clearly define the context of a query, or its associated "concept". The next question is how can we find the context of the query in the target language? We argue that relevant documents in target language contain such an information. In other words, the *coherence* is implicitly present in relevant documents. Even if relevant documents are obviously not known in advance, they can be found by active relevance feedback or pseudo-relevance feedback (PRF). Hence, our algorithm will adapt the probabilities in the dictionary based on the set of (pseudo) relevant documents. Before going into the details of this adaptation mechanism, let us first review monolingual PRF techniques in the framework of Language Modelling-based retrieval. Their extension to the cross-lingual case will provide us with the adaptation method.

3.1 Monolingual PRF within the language modeling framework

Traditional methods, such as Rocchio's algorithm, extract terms from feedback documents and add them to the query. The language modeling approach to information retrieval goes beyond this approach: it extracts a probability distribution over words from the feedback documents. We shall first present the general setting for pseudo-feedback with monolingual language models.

- Let $\mathcal{C}$ be a corpus, d_k a document of the corpus.
- Let n the number of top documents selected after a first retrieval.
- $\mathbf{F} = (d_1, \ldots, d_n)$ the feedback documents.
- Let θ_F, a multinomial parameter, standing for the distribution of relevant terms in $\mathbf{F}$: in other words θ_F is a probability distribution over words peaked on relevant terms.

Feedback methods have two aspects: first extracting relevant information (identification of θ_F) and, secondly, enriching the query.

3.1.1 Estimation of θ_F

To estimate θ_F from feedback documents $\mathbf{F}$, we present as an example the method of Zhai and Lafferty [21]. They propose the following generative process for $\mathbf{F}$:

- For i from 1 to n, draw document d_i following the distribution:
  - $d_i \sim Multinomial(l_{d_i}, \lambda\theta_F + (1-\lambda)p(.|\mathcal{C}))$

so that we have the following global likelihood:

$$P(\mathbf{F}|\theta) = \prod_k \prod_w (\lambda\theta_{Fw} + (1-\lambda)P(w|\mathcal{C}))^{c(w,d_k)} \tag{6}$$

$P(w|\mathcal{C})$ is word probability built upon the corpus, λ is a fixed parameter, which can be understood as a noise parameter for the distribution of terms. $c(w, d_k)$ is the number of occurence of term w in document d_k. Finally θ_F is learned by optimising the data loglikelihood with an Expectation Maximization algorithm.

3.1.2 Updating the original query

Now suppose the relevant language model θ_F has been estimated; how can we add the information from the feedback to the query? Within the language model approach to IR, a query is represented as a probability distribution over words (in practice a multinomial distribution which is estimated from maximum likelihood). If θ_Q is the multinomial parameter for a query Q, then the ML-estimation of θ_{Qw} is equal to the proportion of words w in the query Q. To come back with the

initial question of how to combine information from the initial query and feedback documents, a simple method is to simply mix the parameters of their distributions:

$$\theta_{new\_query} = \alpha\theta_{old\_query} + (1-\alpha)\theta_F \tag{7}$$

In practice, we restrict θ_F to their top N words, by considering all other values of this vector as null.

We can note that more elaborated techniques exists in [18]. Setting the value of α is done experimentally and adapted to collections. The robustness of the estimation of θ_F has a significant impact on the value of α. Lastly, the value of α could be understood as a trade off between precision and recall.

3.2 Extension to the Cross-lingual case: Dictionary Adaptation

We generalize the monolingual mixture model for feedback to the case of CLIR: the input data are an initial source query language model $p(w_s|q_s)$ and a first dictionary $p(w_t|w_s)$. The monolingual mixture model can be interpreted as follows: for each term in a document, first choose between the relevant topic model or the corpus language model. Then generate the frequency of the term from the chosen mixture component. We extend this process, by choosing either a source query term w_s (instead of the relevant topic model), or the target corpus ($\mathcal{C}$) language model, for each term in a feedback document. If a query term w_s has been chosen, then a target term w_t is generated with some unkown (ideal) probabilistic dictionary. Mathematically, this gives:

- For i from 1 to n, draw document d_i with:
  - $d_i \sim Multinomial(l_{d_i}, \lambda \sum_{w_s} \theta_s p(w_s|q_s) + (1-\lambda)p(.|\mathcal{C}))$

where l_{d_i} is the length of document d_i.

In this framework, θ_s can be interpreted as an adapted probability of translation : $\theta_{st} \equiv p(w_t|w_s, q_s)$. But is can be interpreted too as a probability distribution (multinomial parameter) over the vocabulary of target terms; it is like a language model, but associated to a specific word w_s. To understand the connections between the monolingual model and the bilingual model, we can make an analogy of this form : $\theta_F \equiv \sum_{w_s} \theta_s p(w_s|q_s)$. Note that the same algorithm realizes both the query enrichment and the dictionary adaptation. Note also that the translation/adaptation is limited to the words of the query (w_s) if we adopt a simple maximum likelihood language model for the query (what is assumed in the following). Lastly, but importantly, the role of the initial (probabilistic), non-adapted dictionary relies in providing the algorithm with a good starting candidate solution for θ_s.

From this generative process, it remains to solve the problems of estimating the parameters $(\theta_s)_{w_s \in Q}$ and of generating the new query language model (on the target side).

3.2.1 Estimation of adapted translation probabilities

We now proceed to the estimation of the parameters $(\theta_s)_{w_s \in Q}$ with maximum likelihood approach using an EM-like algorithm. Recall that, as in the monolingual setting, λ is a fixed parameter and $p(w_s|q_s)$ is also known since it represents the distribution of words in a particular query.

First, the model likelihood can be written in the equivalent form:

$$P(\mathbf{F}|\theta) = \prod_k \prod_{w_t} \left(\lambda(\sum_{w_s} \theta_{st} p(w_s|q_s)) + (1-\lambda)P(w_t|\mathcal{C})\right)^{c(w_t,d_k)} \tag{8}$$

We can maximize the log-likelihood with an EM algorithm. Let t_{wd} the hidden random variable whose value is 1 if word w in document d has been generated by $p(.|\mathcal{C})$. Let r_{ws} be the indicator for which query word has been chosen. Let $\theta_{ts} = p(w_t|w_s, q_s)$ be the unknown parameter of this model.

The E-step gives:

$$p(t_{wd} = 1|\mathbf{F}, \theta^{(i)}) = \frac{(1-\lambda)p(w_t|\mathcal{C})}{\lambda(\sum_{w_s} \theta_{ts}^{(i)} p(w_s|q_s)) + (1-\lambda)P(w_t|\mathcal{C})} \tag{9}$$

$$p(t_{wd} = 0|\mathbf{F}, \theta^{(i)}) = 1 - p(t_{wd} = 1|\mathbf{F}, \theta^{(i)}) \tag{10}$$

Then, r_{ws} is only defined for $t_{wd} = 0$:

$$p(r_{ws} = k|\mathbf{F}, \theta^{(i)}, t_{wd} = 0) \propto p(w_s = k|q_s)\theta_{ts}^{(i)} \tag{11}$$

As usual, in the M-step, we try to optimize a lower bound of the expected log-likelihood :

$$\begin{aligned} Q(\theta^{(i+1)}, \theta^{(i)}) &= \sum_{d,w} c(w,d)\Big(p(t_{wd} = 1|\theta^{(i)}) \log((1-\lambda)p(w|\mathcal{C})) \\ &+ p(t_{wd} = 0|\theta^{(i)}) \sum_{w_s} p(r_{ws} = k|\theta^{(i)}) \log(p(w_s = k|q_s)\theta_{ts}^{(i+1)})\Big) \end{aligned} \tag{12}$$

Differentiating w.r.t. $\theta^{(i+1)}$ and adding Lagrange multiplier (for $\sum_{w_t} \theta_{ts} = 1$) gives the M-step:

$$\theta_{ts}^{(i+1)} \propto \sum_{d} c(w_t, d)p(t_{wd} = 0|\mathbf{F}, \theta^{(i)})p(r_{ws} = k|\mathbf{F}, \theta^{(i)}, t_{wd} = 0) \tag{13}$$

As already mentioned, $\theta^{(0)}$ is given by the corresponding part of an initial (probabilistic), non-adapted dictionary.

3.2.2 Query Update

When the algorithm converges giving some optimal $\theta^{(adapted)}$ parameters, a new query can be generated by using *all* entries in the adapted dictionary ($(\theta_s^{adapted})_{w_s \in Q}$), so no selection method, nor threshold is required to compute the new query. To make the analogy with monolingual IR we do not use a parameter like α or, in a sense, we use $\alpha = 1$, since we only use the dictionary learnt by feedback. The new query language model becomes:

$$P(w_t|q_s) = \sum_{w_s} \theta_{st}^{adapted} P(w_s|q_s) \tag{14}$$

In others words, model **CL_LM1** with $p(w_t|w_s) = \theta_{st}^{adapted}$ is used to perform the retrieval.

3.3 Remarks

The initial dictionary is used as the starting point for the EM algorithm. As a consequence, only non zero entries are used in this algorithm. During the iterations of the EM, the dictionary weights are adapted to fit the feedback documents and hence to choose the correct translations for a query.

In the introduction to dictionary adaptation we argued that one should model the probability $P(w_t|w_s, q)$. In the model represented by equation 4, we made an independence assumptions which discard the query q from this latter probability. However, the query q is implicitly present in the feedback documents, which enables to learn translation probabilities from the context of the query. [11] propose a feedback method for CL_LM2 relying also on dictionary adaptation. Our method is an extension of the classical monolingual mixture model for feedback to the cross-lingual case, which is also a natural feedback method for CL_LM1. However, Hiemstra and al. [11] experiments show that their model were unable to perform pseudo-relevance feedback, but was very good with active relevance feedback.

4 Lexical Entailment as Query Expansion Mechanism

Lexical Entailment (LE) [3, 10, 5] models the probability that one term entails another, in a monolingual framework. It can be understood as a probabilistic term similarity or as a unigram language model associated to a word (rather than to a document or a query). Let u be a term in the corpus, then lexical entailment models compute a probability distribution over terms v of the corpus $P(v|u)$. These probabilities can be used in information retrieval models to enrich queries and/or documents and to give a similar effect than the use of a semantic thesaurus. However, lexical entailment is purely automatic, by extracting statistical relationships from the considered corpus. In practice, a sparse representation of $P(v|u)$ is adopted, where we restrict v to be one of the N_{max} terms that are the closest from u using an Information Gain metric [1] .

We refer to [3] for all technical and practical details of the method. Still one important thing to be mentioned is that the LE models $P(v|u)$ are used as if this was a cross-lingual framework (for instance one of the **CL_LM**1 or **CL_LM2** models), i.e. as if $P(v|u)$ was a probabilitic translation matrix. If $q = (q_1, ..., q_l)$ and if **CL_LM2** is chosen, this gives using the CE criterion:

$$CE(q|d) = \sum_{q_i} P(q_i|q) \log(\sum_{w} P(q_i|w)P(w|d)) \tag{15}$$

$P(q_i|w)$ is the result of the Lexical Entailment model, and $P(w|d)$ is given by equation 2. We also used a slighty modified formula, introducing a background query-language smoothing $P(q_i|\mathcal{D})$. Instead of eq. 15, the document score is now computed as:

$$CE(q|d) = \sum_{q_i} P(q_i|q) \log(\beta \sum_{w} P(q_i|w)P(w|d) + (1-\beta)P(q_i|\mathcal{D})) \tag{16}$$

5 Experiments on GIRT - 2004 to 2006

We refer to the overview paper [1] for the description of the task, the corpora and the available resources (see aldo `http://www.gesis.org/en/research/information_technology/girt4.htm` for specific information).

In order to do some preliminary tunings and validations, we used the domain-specific corpus GIRT as available in 2006 from the CLEF Evaluation Forum, as well as the 75 queries and their relevance assessments collected from the years 2004, 2005 and 2006. In the next section, we will present the results on the test data, namely the new GIRT corpus (extended on the english side, by additional documents coming from the CSA corpus) and the corresponding new queries. We used Mean Average Precision (MAP) as retrieval performance measure.

For the whole collection and the queries, we used our home-made lemmatiser and word-segmenter (decompounder) for German. Classical stopword removal was performed. We used only the title and the description of the queries.

As multilingual resources, we used on the one hand the English-German GIRT Thesaurus (considered as domain-specific, but very narrow) and, on the other hand, a probabilistic one, called ELRAC, that is a combination of a very standard one (ELRA) and a lexicon automatically extracted from the parallel JRC-AC (Acquis Communautaire) Corpus (see `URL: langtech.jrc.it/JRC-Acquis.html`) using the *Giza++* word alignment algorithm.

As already mentioned, one goal of the experiments was to compare the query translation approach using dictionary adaptation with the use of our Statistical Machine Translation system (MATRAX). The latter needs two kinds of corpus: a parallel corpus for the alignment models, and a corpus in the target language to learn a "language model". We fed MATRAX with the JRC-AC (Acquis Communautaire) Corpus for the alignment models, and with out GIRT / CSA corpora (in the target language) for the language models. In this way, we can expect to introduce some bias or adaptation to our target corpus in the translation process, as the Language Model component of Matrax will favour translation and disambiguation consistent with this corpus.

[1] The Information Gain, aka Generalised (or average) Mutual Information [4], is used for selecting features in text categorisation [19, 7] or detecting collocations [6].

Table 1: Monolingual Experimental Results in MAP

Language	Before feedback	After Feedback
EN	0.33	0.37
GER	0.41	0.48

Table 2: Monolingual Lexical Entailment (LE) Experimental Results in map

Language	Simple LE	Double LE	Standard approach with PRF (baseline)
EN	0.38	0.41	0.38
GER	0.45	0.51	0.49

Table 3: Dictionary Adaptation Experimental Results in MAP

Translation	Initial Dictionary	Without adaptation	After adaptation	Rel. Improv.
EN to GER	Thesaurus	0.3054	0.3385	10%
EN to GER	ELRAC	0.2751	0.3502	29%
GER to EN	Thesaurus	0.3068	0.3516	16%
GER to EN	ELRAC	0.2089	0.3027	50%

5.1 Monolingual Experiments

Monolingual results enable to evaluate the performance of the cross-lingual results, being a reference to compete with. Table 1 shows the results of the monolingual experiments. A Dirichlet prior smoothing was used with a value of 200, and PRF was applied, using the TOP15 documents with the mixture model algorithm described in 3.1. We observe a significative difference between the behavior of the english corpus and the german one. English documents are sparser than german ones, which explains the retrieval deficiency.

Table 2 shows monolingual experiments using lexical entailment models. We used the top 20 entailed terms ($N_{max} = 20$) , for each german term, and the top 10 terms for english term ($N_{max} = 10$) since the english corpus is sparser than the german one. We applied LE first on the basic query (results are given in column 2). Then the mixture model algorithm for pseudo-feedback described in 3.1 is applied on the top15 documents, which provides a new query and once again the lexical entailment model is applied. The lexical entailment model using pseudo-relevance feedback will also be called *PRF+Lexical Entailment*, or *Double Lexical Entailment* (as actually the top15 documents are retrieved using a first lexical entailment step). Performance of this model is given in Column 3 of Table 1. One can see that lexical entailment models perform better than the baseline monolingual models without feedback and that lexical entailment techniques provide improvment comparable (and better than) to those obtain by pseudo-relevance feedback.

5.2 Cross-lingual Experiments

5.2.1 Baseline

Table 3 (column 3 - without adaptation) shows the experimental results using the dictionary adaptation algorithm. We tested the algorithm both for the English-to-German and German-to-English translations. We also used different initial dictionaries : the first one based on the GIRT thesaurus and the second one based on ELRAC. We used model CL_LM1 (cf. eq 4) for the retrieval. Recall that in model CL_LM1, the query words are translated with the dictionary and then some monolingual search is performed.

This baseline used all translation candidates: this makes the queries noisy and has the consequence that any traditional monolingual relevance feedback algorithm we tried did not boost the performance of the retrieval. As the query is already noisy, it is likely that expanding it make it unstable since feedback terms are mixed with irrelevant terms issued by the naive translation.

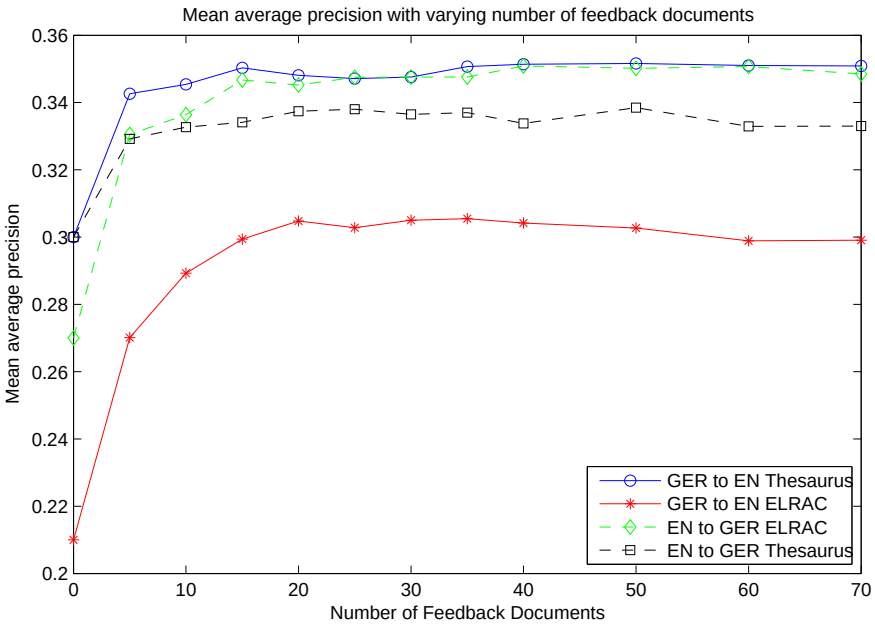

Figure 1: Influence of the number of pseudo-feedback documents

5.2.2 Dictionary Adaptation

Then, we perform a dictionary adaptation with parameters $\lambda = 0.5$ (in equation 8) and the number of feedback documents is set to 50 (Table 3). The results show that, with dictionary adaptation, we gain in performance for every dictionary and translation sense. We obtain a global improvement ranging from 3% to 10% , and a relative improvement from 10% to 50% and an average gain of 6% for both directions and both dictionaries.

The thesaurus used is the one provided by GIRT and already performs well since it is adapted to the corpus of GIRT: there is less ambiguity in this dictionary than in the standard ELRAC dictionary. Still, the method is able to gain in precision. The interesting fact is the improvement obtained by the ELRAC dictionary after adaptation. ELRAC is a general dictionary not at all adapted to social science corpus of GIRT. The initial performance of ELRAC is worst than using the GIRT thesaurus. However, the dictionary adaptation improves a lot the query translation process(8% avg increase in map on both direction). This shows that a general dictionary with the adequate adaptation mechanism can be used for a specialized corpus, without a huge loss compared to a domain specific dictionary. Of course, domain specific dictionaries work better but they require external resources, or comparable corpora to be extracted from, whereas general dictionaries are always more easily available. Beyond the feature of giving a more accurate translation, a second reason of these improvements is that dictionary often encodes some semantic enrichment. For example the word *area* can be translated in french into *region*, or *zone*.

Figure 1 shows the evolution of mean average precision with an increasing number of pseudo-feedback documents. This graph indicates that the algorithm seems to be very stable and robust to a large set of feedback documents. One can also notice , that much of the gain can be obtained using only the top 10 documents. We believe the stability is due to the initialization of algorithm with the previous dictionary, which make only non zeros entries serves as training data.

Figure 2 shows the influence of the λ parameter. This parameter can be interpreted as a noise parameter in the feedback documents. Since, we restrain ourself to non-zero entries, a better interpretation would be as a noise parameter in the dictionary. The conclusion we can draw from this graph, is that modeling the noise is useless when only non-zero entries are used. So, the algorithm should be used with $\lambda = 1$. This parameter could have more influence if we extend the

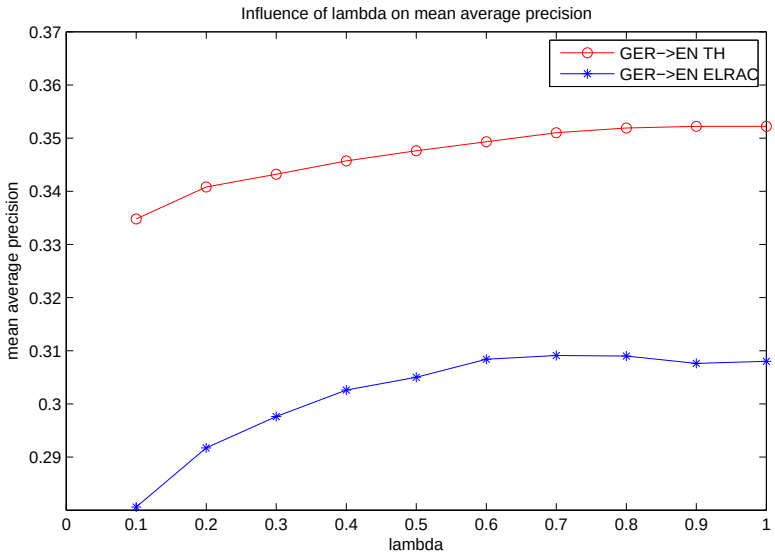

Figure 2: Influence of λ

Table 4: CLIR Results with Lexical Entailment in map

Translation	Method	baseline	Simple Lex. Ent.	PRF Lex. Ent.
EN to GER	DA Thesaurus	0.3385	0.36	0.39
EN to GER	DA ELRAC	0.3502	0.38	0.41
GER to EN	DA Thesaurus	0.3516	0.37	0.39
GER to EN	DA ELRAC	0.3027	0.33	0.36

number of feedback documents to a larger value. Then, the data would be noisier. The results around the influence of the number of top documents show that they are sufficient to disambiguate the query. However, if we were to "smooth" zero entries of the dictionary (and then allow new translation candidates that were not present in the initial dictionary), this noise parameter would influence much more the performance. There are two problems acting at the same time : query translation and query enrichment. Enriching the query amounts to smoothing non zeros entries in the dictionary. We believe it is more important to solve the query translation problem first and enrich the query later (possibly with another monolingual mechanism). Hence, the λ parameter can be set to 1 without loss of performance.

Table 4 shows the results of lexical entailment model after a first step of dictionary adaptation. To sum up, the original query is first roughly translated with an initial dictionary, then a first retrieval is done and the dictionary is adapted to the query: a new translation of the query is obtained. The baseline model is the model CL_LM1 using the new translated query. Instead of using CL_LM1, the others models rely on a Lexical Entailment model. As before, *Simple Lex Entailment* names the model CL_LM2 with the lexical entailment model based on the information gain. *PRF Lex Ent* denotes the same model, but with a step a pseudo-feedback with the mixture model introduced previously. Once again, the lexical entailment model outperforms the baseline. One can argue that, both models CL_LM1 and CL_LM2 are alternatively use in the same retrieval process. This comes from historical reasons : we first developped lexical entailment model a few years ago, and dictionary adaptation model later on (for CLEF07). These two models were combined afterwards. Theoretically, it could be interesting to develop a single model tackling the multilinguality and the use of monolingual thesaurus in a single framework.

6 Experimental results on GIRT 07

We now proceed to our participation to the Domain Specific Task in CLEF 2007, on the GIRT and CSA corpora. Once again, we refer to [16] for a precise description of the task, the corpora, and the available resources. We submitted monolingual runs as well as bilingual runs, restricted to English and German. Our monolingual runs mainly rely on lexical entailment models. The bilingual runs are issued from two techniques: either query translation with our home-developed Statistical Machine Translation System called Matrax, or query translation through dictionary adaptation.

6.1 Parameters, Nomenclature and Monolingual Runs

Table 5: Monolingual pseudo-feedback Parameters

Value	Notation in this report
	GERMAN
15	n of section 3.1
0.85	α in eq. 7
20	Take the top N words from θ_F (cf section 3.1.2)
0.6	λ in eq. 6
	ENGLISH
10	n cf section3.1
0.8	α in eq. 7
20	Take the top N words from θ_F (cf section 3.1.2)
0.6	λ in eq. 6

Table 6: Lexical Entailment IR Model Parameters

Name	Value	Reference
λ	0.9	eq. 2
β	0.125	eq. 16

Tables 5 and 6 show the main parameters of our system. If a run contains *prf* (respectively *le*), in its name then it used parameters described in table 5 (respectively table 6). The item list below describes the nomenclature of our retrieval models.

- **Language Model+ PRF**: The standard query likelihood (or equivalently cross-entropy) approach, with the mixture model for pseudo-feedback (as explained in section 3.1);
- **Lexical Entailment** : the lexical model with Information Gain used in conjunction with the CL_LM2 model
- **Language Model + PRF + Lexical Entailment** : After a first retrieval with Language Model and PRF (as in bullet 1), the enriched query is scored with a Lexical Entailment model
- **PRF Lexical Entailment** : this is the *Double Lexical Entailment* model explained before, where a first lexical entailment model is used to provide the system with an initial set of TOPn documents, from which a mixture model for pseudo-feedback is built, and a second retrieval is performed based once again on the lexical entailment model applied to the enriched query.

Table 7 shows our official runs with their result in mean average precision and their associated information retrieval model.

Table 7: Official Monolingual Runs with their underlying model and results in MAP

Model	MAP	Run Name
GERMAN		
Lexical Entailment Simple	0.3475	xrcelede
Language Model + PRF	0.4465	xrceprfde
Language Model + PRF + Lexical Entailment	0.5014	xrceprfdele
PRF Lexical Entailment	0.5051	xrceprflede
ENGLISH		
Lexical Entailment Simple	0.2722	xrceleen
Language Model + PRF	0.2934	xrceprfde
Language Model + PRF + Lexical Entailment	0.3237	xrceprfenle
PRF Lexical Entailment	0.3051	xrceprfleen

6.2 Bilingual Runs

The bilingual retrieval model adopts the same nomenclature as in previous sections.

As allready explained, all our bilingual runs follow the same schema "query translation" followed by a monolingual search (most often with PRF or query expansion in the target language). For the first step —query translation —, we used either our Statistical Machine Translation system (MATRAX), either one (initial standard) dictionary adapted following the strategy described in this paper. The monolingual search component obeys the same nomenclature as in the previous section.

In order to increase the recall of what can be obtained with MATRAX, we intentionally kept the TOP5 most plausible translations given by MATRAX and concatenated them to obtain the new query in the target language (this indeed significantly increased the performance of the retrieval).

In order to perform lexicon adaptation, the choice of the initial dictionary is crucial to the task. We used two initial dictionaries that were at our disposal: the first one, CsaGirt, has been extracted from the concatenation of the GIRT and CSA thesauri. The second dictionary was ELRAC, composed as described before. Hence, to benefit from both sources, the dictionaries were merged hierarchically : an entry of the dictionary is added to the other one, if this entry is not already present in the master dictionary. The dictionary named Hier-CsaGirtElrac (abbreviation: hcge) is the dictionary obtained by giving priority to the dictionary CsaGirt and then adding any Elrac entry not already present in CsaGirt. The dictionary named Hier-ElracCsaGirt (abbreviation: hecg) is the dictionary obtained by giving priority to the dictionary Elrac and then adding the dictionary CsaGirt.

Table 8 shows the result of our bilingual runs with their mean average precision and the model used for translation and retrieval. If no other query expansion (in the target language) is done beyond the lexical entailment model, Matrax offers the best results (but recall that Matrax is significantly harder and more time-consuming to train than our simple dictionary extraction and adaptation). However, it seems that, once we want to adopt more complex PRF techniques after translation, there is a substantial advantage to use our dictionary adaptation method that, presumably, gives less noisy translations. Consequently, the best absolute performance are obtained by combining (1) the hierarchical building of the inital dictionary (the order in the hierarchy is dependent of the source and target languages, (2) adapting this initial dictionary with the proposed algorithm and (3) performing a rather sophisticated (PRF+Lexical Entailment) query expansion/enrichment in the target language. Note that, when English is the target language, bilingual performances are even better than monolingual ones.

Table 9 shows the results of some experiments that we performed after the submission to CLEF, but using the CLEF 2007 queries and relevance assessments. The table intent is to better understand the individual effect of the basic components of our official runs. We can observe that the monolingual pseudo-relevance feedback algorithm improves a lot the results: for German, it boosted the mean average precision form 0.30 to 0.44. We can also see that the dictionary

Table 8: Official Bilingual Runs with their underlying model and results in MAP

Bilingual Model	MAP	Run Name
ENGLISH to GERMAN		
Matrax + Language Model + PRF	0.4	xrcee2dmatrax
Matrax+Lexical Entailment Simple	0.43	xrcee2dmatraxle
Matrax+ PRF Lexical Entailment	0.4298	xrcee2dmatraxprfle
Adapt Dico Hier-CsaGirtElrac + Lexical Entailment	0.3905	xrcee2dhcgele
Adapt Dico Hier-CsaGirtElrac + PRF Lexical Entailment	0.4447	xrcee2dhcgeprfle
Adapt Dico Hier-ElracCsaGirt + PRF Lexical Entailment	0.4568	xrcee2dhecgprfle
GERMAN to ENGLISH		
Matrax + Language Model + PRF	0.2468	xrced2ematrax
Matrax+Lexical Entailment Simple	0.2757	xrced2ematraxle
Matrax+ PRF Lexical Entailment	0.2873	xrced2ematraxprfle
Adapt Dico Hier-ElracCsaGirt + Lexical Entailment	0.2338	xrced2ehecgle
Adapt Dico Hier-CsaGirtElrac + PRF Lexical Entailment	0.3341	xrced2ehcgeprfle
Adapt Dico Hier-ElracCsaGirt + PRF Lexical Entailment	0.2923	xrced2ehecgprfle

Table 9: Unofficial runs with their underlying model and results in MAP

Model Description	MAP	MAP
Matrax Language Model without PRF		
english to german	0.2911	
german to english	0.2083	
Adaptation of Dictionary	Before	After
english to german : Hier-CsaGirtElrac	0.2768	0.3541
english to german : Hier-ElracCsaGirt	0.2127	0.3050
german to english : Hier-CsaGirtElrac	0.2072	0.2454
german to english : Hier-ElracCsaGirt	0.154	0.207
Monolingual		
english Language Model	0.2511	
german Language Model	0.3016	

adaptation also works for queries of this year. Finally, there is still a deficiency when the target corpus is the english corpus: we still believe this is due to the unbalanced nature of the documents (german documents are longer in average and, consequently, more reliable, because they most often contain the abstract field).

7 Conclusion to GIRT Participation

Our main goal this year was to validate two query translation and disambiguation strategies. The first one relies on the use of our Statistical Machine Translation tool, especially taking benefit from its flexibility to output more than one plausible translations and to train its Language Model component on the CLEF07 target corpora. The second one relies on a pseudo-feedback adaptation mechanism that performs simultaneously dictionary adaptation and query expansion.

Experimental results on CLEF-2007 corpora (domain-specific track) show that the dictionary adaptation mechanisms appear quite effective in the CLIR framework, exceeding in certain cases the performance of much more complex Machine Translation systems and even the performance of the monolingual baseline. The pseudo-feedback adaptation method turns out to be robust to the number of feedback documents and relatively efficient since we do not need to extract co-occurence statistics. It is also robust to the noise in feedback documents, contrary to several traditional

monolingual feedback methods that decreased their performances in our experiments. Lastly, it enables to use general dictionaries in domain specific context with almost as good performance as domain specific dictionaries.

We believe that the concept of *adaptation of lexicon* has other applications in cross-lingual information access tasks. For instance, if there is some underlying class or category system (built in a supervised or unsupervised way), lexicons could be adapted to a particular category/cluster. Moreover, the adaptation model could be useful to adapt a dictionary to a user profile: from feedback sessions, one can learn an bilingual lexicon adapted to a particular user, which has significant applications. Our further works will focus on such aspects.

Aknowledgments

This work was partly supported by the IST Programme of the European Community, under the SMART project, FP6-IST-2005-033917. The authors also want to thank Francois Pacull for his greatly appreciated help in applying the MATRAX tools in CLEF07 experiments.

References

[1] S. Baerisch and M. Stempfhuber. Domain-specific track clef 2006 : Overview of the results. In *CLEF 2006: Proceedings of the Workshop of the Cross-Language Evaluation Forum, Alicante, Spain, September 20 - 22, 2006.* Springer, 2006.

[2] A. L. Berger and J. D. Lafferty. Information retrieval as statistical translation. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 222–229. ACM, 1999.

[3] S. Clinchant, C. Goutte, and É. Gaussier. Lexical entailment for information retrieval. In M. Lalmas, A. MacFarlane, S. M. Rüger, A. Tombros, T. Tsikrika, and A. Yavlinsky, editors, *ECIR*, volume 3936 of *Lecture Notes in Computer Science*, pages 217–228. Springer, 2006.

[4] B. Colin. Information et analyse des données. *Pub. Inst. Stat. Univ. Paris*, XXXVII(3–4):43–60, 1993.

[5] I. Dagan, O. Glickman, and B. Magnini. The pascal recognising textual entailment challenge. In *PASCAL Challenges Workshop for Recognizing Textual Entailment*, 2005.

[6] T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74, 1993.

[7] G. Forman. An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research*, 3:1289–1305, 2003.

[8] J. Gao, J.-Y. Nie, E. Xun, J. Zhang, M. Zhou, and C. Huang. Improving query translation for cross-language information retrieval using statistical models. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 96–104, New York, NY, USA, 2001. ACM Press.

[9] J. Gao, J.-Y. Nie, and M. Zhou. Statistical query translation models for cross-language information retrieval. *ACM Transactions on Asian Language Information Processing (TALIP)*, 5(4):323–359, 2006.

[10] O. Glickman, I. Dagan, and M. Koppel. A probabilistic classification approach for lexical textual entailment. In *Twentieth National Conference on Artificial Intelligence (AAAI-05)*, 2005.

[11] D. Hiemstra, W. Kraaij, R. Pohlmann, and T. Westerveld. Translation resources, merging strategies, and relevance feedback for cross-language information retrieval. In C. Peters, editor, *CLEF*, volume 2069 of *Lecture Notes in Computer Science*, pages 102–115. Springer, 2000.

[12] W. Kraaij, J.-Y. Nie, and M. Simard. Embedding web-based statistical translation models in cross-language information retrieval. *Comput. Linguist.*, 29(3):381–419, 2003.

[13] Y. Liu, R. Jin, and J. Y. Chai. A maximum coherence model for dictionary-based cross-language information retrieval. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 536–543, New York, NY, USA, 2005. ACM Press.

[14] C. Monz and B. J. Dorr. Iterative translation disambiguation for cross-language information retrieval. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 520–527, New York, NY, USA, 2005. ACM Press.

[15] J.-Y. Nie and M. Simard. Using statistical translation models for bilingual ir. In *CLEF '01: Revised Papers from the Second Workshop of the Cross-Language Evaluation Forum on Evaluation of Cross-Language Information Retrieval Systems*, pages 137–150, London, UK, 2002. Springer-Verlag.

[16] V. Petras, S. Baerisch, and M. Stempfhuber. The domain-specific track at clef 2007. In *CLEF 2007: Proceedings of the Workshop of the Cross-Language Evaluation Forum, Budapest, Hungary, September 19 - 21, 2007.*, page forthcoming. Springer, 2007.

[17] J. Ponte and W. Croft. A language modelling approach to information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 275–281. ACM, 1998.

[18] T. Tao and C. Zhai. Regularized estimation of mixture models for robust pseudo-relevance feedback. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, New York, NY, USA, 2006.

[19] Y. Yang and J. O. Pedersen. A comparative study on feature selection in text categorization. In *Proceedings of ICML-97, 14th International Conference on Machine Learning*, pages 412–420, 1997.

[20] C. Zhai and J. lafferty. A study of smoothing methods for language models applied to ad hoc to information etrieval. In *Proceedings of SIGIR'01*, pages 334–342. ACM, 2001.

[21] C. Zhai and J. D. Lafferty. Model-based feedback in the language modeling approach to information retrieval. In *CIKM*, pages 403–410. ACM, 2001.