Cross-language information retrieval in Twenty-One:
Using one, some or all possible translations?

Djoerd Hiemstra and Franciska de Jong

University of Twente, CTIT
P.O. Box 217, 7500 AE Enschede
The Netherlands
{hiemstra,fdejong}@cs.utwente.nl

ABSTRACT

This paper gives an overview of the tools and methods for Cross-Language Information Retrieval (CLIR) that were developed within the Twenty-One project. The tools and methods are evaluated with the TREC CLIR task document collection using Dutch queries on the English document base. The main issue addressed here is an evaluation of two approaches to disambiguation. The underlying question is whether a lot of effort should be put in finding *the* correct translation for each query term before searching, or whether searching with more than one possible translation leads to better results? The experimental study suggests that in terms of average precision, searching with ambiguities leads to better retrieval performance than searching with disambiguated queries.

Keywords: Cross-Language Information Retrieval, Statistical Machine Translation.

1 INTRODUCTION

Within the project Twenty-One a system is built that supports Cross-language Information Retrieval (CLIR). CLIR supports the users of multilingual document collections by allowing them to submit queries in one language, and retrieve documents in any of the languages covered by the retrieval system. For this type of functionality the translate option offered by some web search engines does not suffice, because it does not help the users to identify material that they might want to have translated. Since this approach presupposes that the users have already found the relevant document in its original foreign language, it fails to support exactly that part of a search in a mul-tilingual environment which is the most difficult one, viz., to formulate a query which will then take the user to the foreign language document in whose content he might be interested in. In order to support the retrieval of documents irrespective of the document language, either off-line document translation (DT), off-line index translation, or on-line query translation (QT) is required. From a practical point of view QT is enforced in environments where it would be impossible to produce translations for all documents in the document base and/or to produce translated indices for all languages. However, it has the disadvantage or at least restriction that the user must know the foreign languages at least up to the degree of a passive understanding of this language. The alternatives document translation (DT), or index translation do not necessarily presuppose even a passive knowledge of the foreign language.

Systems operating in a comparatively controlled environment, where the documents are limited either to a specific domain or to a limited number of data and document bases are likely to use DT. Twenty-One is a representative of this category. It has a clear target domain. viz. sustainable development, and with its strong focus on the disclosure of paper documents, which have to be scanned and OCRed, heavy preprocessing and storage has to be reckoned with anyhow. Off-line translation rather than translating query terms during retrieval has important advantages for the way the most critical part of the translation task is dealt with: disambiguation. As all index terms (NPs) are kept in their original position, contextual information is accessible for the disambiguation algorithms that are part of the translation software. Currently disambiguation in Twenty-One can be pursued in three ways:

- selection the dictionary preferred translation

- the use of domain specific dictionaries that are automatically generated on the basis of statistically processed parallel corpora (suited for specific applications only)

- disambiguation on the basis of the frequency of noun phrases in the document collection

This paper is organised as follows. Section 2 explores possibilities for the comparison of the DT approach with the QT approach. Section 3 introduces three basic methods for the QT approach to CLIR. Section 4 addresses heuristics and statistics for translation. Section 5 discusses the setup of our experiments and experimental results. Finally, section 6 contains concluding remarks.

2 EMPIRICALLY COMPARING DT TO QT

As said in the introduction DT has important advantages. Firstly, it can be done off-line. Secondly, if a classical machine translation is used, it is possible to present the user a high quality preview of a document. Thirdly, there is more context available for lexical disambiguation. This might lead to better retrieval performance in terms of precision and recall. For several types of applications, the first and second advantage may be a good reason to choose for DT. The third advantage however is more hypothetical. Does the DT approach to CLIR using classical machine translation really lead to better retrieval performance than the QT approach using a machine readable dictionary?

For a number of reasons it is very difficult to answer this question on the basis of empirical evidence. A first problem is that in the QT approach searching is done in the language of the documents while in the DT approach searching is done in the language of the query. But it is a well known fact that IR is not equally difficult for each language. A second problem is that, for a sound answer to the question, we need a machine translation system and a machine readable dictionary that have exactly the same lexical coverage. If the machine translation system misses vital translations that the machine readable dictionary does list, we end up comparing the coverage of the respective translation lexicons instead of the two approaches to CLIR. Within the Twenty-One project we have a third, more practical, problem that prevents us from evaluating the usefulness of the used translation system

(LOGOS) against the usefulness of the machine readable dictionaries available within the project (Van Dale). The Van Dale dictionaries are entirely based on Dutch head words, but translation from and to Dutch is not supported by LOGOS. All these considerations urge us to rephrase the the issue into a more manageable question.

A first, manageable, step in comparing DT with QT might be the following. What is, given a translation lexicon, the best approach for QT: using one translation for each query term or using more than one translation? Picking one translation is a necessary condition of the DT approach. For QT we can either use one translation for searching, or more than one. The question one or more translations also reflects the classical precision / recall dilemma in IR: picking one specific translation of each query term is a good strategy to achieve high precision; using all possible translations of each query term is a good strategy to achieve high recall.

3 METHODS FOR QT

As said in the previous section this paper compares CLIR using one translation per query term with CLIR using more than one translation per query term. We will report the results of retrieval experiments using the Dutch queries on the English TREC CLIR task collection. A Dutch query will be referred to as the source language query; the English query will be referred to as the translated query. The experiments can be divided into three categories:

1. QT using one translation per source language query term

2. QT using unstructured queries of all possible translations per source language query term

3. QT using structured queries of all possible translations per source language query term

3.1 USING ONE TRANSLATION PER QUERY TERM

If only one translation per query term is used for searching, the translation process must have some kind of explicit disambiguation procedure. This procedure might be based on an existing machine translation system, or alternatively, on statistical techniques or heuristics. After disambiguation, the translated query can be treated the way a

query is normally treated in a monolingual setting. A 'normal' monolingual setting in this context is retrieval on the basis of a statistical 'bag-of-words' model like e.g. the vector space model [10] or the classical probabilistic model [9]. In the next section, the use of a bag-of-words model will be referred to as the *unstructured queries*-option.

In section 4 a number of heuristics and statistics for disambiguation will be explored. As explained in section 2 we will not be able to actually use machine translation for disambiguation. It is however possible to define an upper bound on what is possible with the one-translation approach by asking a human expert to manually disambiguate the output of the machine readable dictionary. We hypothesise that QT using a machine translation system with the same lexical coverage as the machine readable dictionary will not result in better retrieval performance than QT using the manually disambiguated output of the same dictionary.

3.2   USING UNSTRUCTURED QUERIES

If more than one translation per source language query term is used for searching we might still treat the translated query as a bag-of-words. As we will see in section 5 the way of weighting the possible translations is crucial for unstructured queries. In particular it is important to normalise the possible translations in such a way that for each source language query term the weights of possible translations sum up to one. Not using normalisation will make source language query terms with a lot of possible translations unintentionally more important than source language query terms that have only less possible translations.

$$
\begin{aligned}
\text{similarity}(Q, D) &= \sum_{k=1}^{l} w_{qk} \cdot w_{dk} \\
w_{qk} &= tf(k, q) \\
w_{dk} &= \log(1 + \frac{tf(k, d)}{df(k) \sum_t tf(t, d)} \cdot \frac{0.15 \sum_t df(t)}{0.85})
\end{aligned}
$$

Figure 1: vector product weighting algorithm

Instead of using one of the bag-of-words models mentioned above, we will use a weighting algorithm based on a new model of information retrieval: the linguistically motivated probabilistic model [2, 5]. Figure 1 lists the weighting algorithm that was used to rank the documents given

a translated query. In this formula $tf(t, d)$ is the term frequency of the term t in the document d and $df(t)$ is the document frequency of the term t.

3.3   USING STRUCTURED QUERIES

If all possible translations are treated as one bag-of-words we ignore the fact that a document containing one possible translation of each source language query term is more likely to be relevant than a document containing all possible translations of only one source language query term. The boolean model or weighted boolean models (see e.g. [10]) can be used to retrieve only those documents that contain a translation of all or most of the source language query terms [6]. Disjunction can be used combine possible translations of one source language query term. Conjunction can be used in a way that the translated query reflects the formulation of the source language query.

Our structured query approach is based on the linguistically motivated model. A structured query has to be formulated in conjunctive normal form, which is the form in which it is automatically produced after dictionary based translation. The definition of the conjunction is simply the definition of the probability ranking function as introduced in [2] where $T_1, T_2, \cdots, T_n$ is a query of length n and D is a document id.

$$
P(T_1, T_2, \cdots, T_n | D) = \prod_{i=1}^{n} P(T_i | D)
$$

Disjunction of m possible translations of the source language query term on position i is defined as follows.

$$
P(T_{i1} \cup T_{i2} \cup \cdots \cup T_{im} | D) = \sum_{j=1}^{m} P(T_{ij} | D)
$$

The structured query weighting algorithm implicitly normalises the possible translations in a disjunction. Explicit normalisation as done for unstructured queries is no longer necessary. If there are no disjunctions in the query (that is, if there is only one translation per source language query term) then the structured ranking formula will produce exactly the same results as the weighting algorithm of figure 1. Structured queries are generated automatically by the translation module and may take relative frequencies of possible translations into account. A more detailed description of the algorithm will be published in the near future.

3.4 An example

Figures 2 and 3 give an example of an English query {*third, world*} that is used to search a French collection. It is assumed that the English term *third* has two possible French translations: *tiers* and *troisième* and that the English term *world* has three possible translations: *monde, mondial* and *terre*.

```
{third, world}
      ↓ translation
(tiers, troisième, monde, mondial, terre)
```
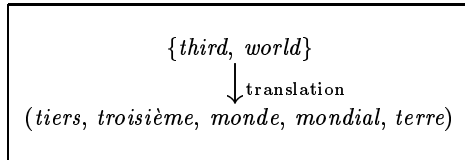
Figure 2: translation using an unstructured query

The result of figure 2 could be used directly for searching the French collection (see **run2a** in section 5), but this would make the term *world* in the source language query more important (because it has more possible translations) than the word *third*. The query weights of the weighting algorithm of figure 1 might therefore be used to make the contribution of *third* as high as the contribution of *world* by reweighting (normalising) the possible translations of *third* to 0.5 and the possible translations of *world* to 0.33 (see **run2c** in section 5). If one of the possible translations of one source language query term is more probable than the other(s), this possible translation might be weighted higher than the other(s) while keeping the normalisation in tact.

```
{third, world}
      ↓ translation
{(tiers ∪ troisième), (monde ∪ mondial ∪ terre)}
```
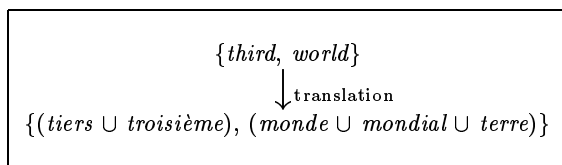
Figure 3: translation using a structured query

The structured query of figure 3 reflects the possible translations of the source language query terms in an intuitive way. Possible translations of one original query term might be weighted differently. Normalisation is an implicit feature of the weighting algorithm.

4 Heuristics and statistics for QT

This section lists a number of information resources that can be used to identify the proper translation or proper translations of a query term. The section briefly describes information that is explicitly or implicitly in the dictionary and information from other sources like parallel corpora and the document collection itself.

4.1 Dictionary preferred translation

The VLIS lexical database of Van Dale Lexicography list for each entry explicitly one *preferred* translation which is considered the most commonly used one. Replacing each query term with the preferred translation is a simple, but possibly effective, approach to CLIR.

4.2 Pseudo frequencies

The Van Dale database contains also explicit information on the sense of possible translations. Some Dutch head words carry over to the same English translation for different senses. For example the Dutch head word *jeugd* may be translated to *youth* in three senses: the sense of 'characteristic', 'time-frame' and 'person'. The 'person' sense has a synonym translation: *youngster*. As *youth* occurs in the dictionary under three senses and *youngster* under one sense, we assign *youth* a weight that is three times as high as the weight for *youngster*. The assumption made by weighting translations is that the number of occurrences in the dictionary may serve as rough estimates of actual frequencies in parallel corpora. In other words: the number of occurrences in the dictionary serve as *pseudo frequencies*. Ideally, if the domain is limited and parallel corpora on the domain are available, weights should be estimated from actual data as described in section 4.3.

4.3 Frequencies from parallel corpora

The Twenty-One system contains documents on the domain of sustainable development. Translation in Twenty-One is done using a general purpose dictionary (Van Dale) and a general purpose MT-system (LOGOS), but these resources are not very well suited for domain-specific jargon. Domain-specific jargon and its translations are implicitly available in parallel corpora on sustainable development. Translation pairs can be derived from parallel corpora using statistical co-occurrence by so-called word alignment algorithms. Within the Twenty-One project word

alignment algorithms were developed that do the job in a fast and reliable way [3, 4]. Domain specific translation lexicons were derived from Agenda 21, a UN-document on sustainable development that is available in most of the European languages including Dutch and English.

For the experiment we merged the automatically derived dictionary with the Van Dale dictionary in the following way. For each entry, we added the pseudo frequencies and the real frequencies of the possible translations. Pseudo frequencies are usually not higher than four or five, but the real frequencies in the parallel corpus may be more than a thousand for frequent translation pairs. Adding pseudo frequencies and real frequencies has the effect that for possible translations that are frequent in the corpus the real frequencies will be important, but for translations that are infrequent or missing the pseudo frequencies will be important.

Translation pairs that have a frequency of one or two in the parallel corpus may-be erroneously derived by the word alignment algorithm. If, however, such an infrequent translation pair is also listed in the machine readable dictionary, then the pair was probably correct. Therefore we added a bonus frequency of three to each possible translation that is both in the corpus and in Van Dale.

4.4 CONTEXT FOR DISAMBIGUATION

The techniques introduced so far do not resemble techniques that are actually used in machine translation systems. Traditionally, disambiguation in machine translation systems is based on (syntactic) context of words. In this section a statistical algorithm is introduced that uses context of the original query words to find the best translation. The algorithm uses candidate noun phrases (NPs) extracted from the document base to disambiguate the NPs from the query. NPs were extracted using the standard tools as used in the Twenty-One system: the Xerox morphological tools and the TNO parser. The NPs were sorted and then counted, resulting in a list of unique phrases with frequency of occurrence.

The introduction of NPs (or any multi-word expression) in the translation process leads to two types of ambiguity: sense ambiguity and structural ambiguity. Figure 4 gives an example of the French translation chart of the English NP *third world war*. Each word in this NP can have several translations that are displayed in the bottom cells

Figure 4: translation chart of *third world war*

of the chart, the so-called sense ambiguity. According to a list of French NPs there may be two candidate multi-word translations: *tiers monde* for the English NP *third world* and *guerre mondiale* for *world war*. These candidate translations are displayed in the upper cells of the chart. Because the internal structure of NPs was not available for the translation process, we can translate a full NP by decomposing it in several ways. For example *third world war* can be split up in the separate translation of either *third world* and *war* or in the separate translation of *third* and *world war*. The most probable decomposition can be found using techniques developed for stochastic grammars (see e.g. [1]). The probabilities of the parse trees can be mapped into probabilities, or weights, of possible translations. A more detailed description of the algorithm can be found in [8].

4.5 MANUAL DISAMBIGUATION

The manual disambiguation of the dictionary output was done by a native speaker of English. She had access to the Dutch version of the topics and to the English dictionary output consisting of a number of possible translations per source language (Dutch) query word. For each Dutch word, one of the possible English translations had to be chosen, even if the correct translation was not one of them.

4.6 OTHER INFORMATION

In the experiments described in this paper we ignored one important source of information: the multi-word entries in the Van Dale dictionaries. Multi-word expressions like for instance *world war* are explicitly listed in the dictionary. For the experiments described in this paper we only used word-by-word translations using the single word entries.

5 Experimental setup and results

In section 3 we identified three methods for QT: using one translation per query term, using a unstructured query of all translations per source language query term and using a structured query of all translations per source language query term. Each method is assigned a number 1, 2 or 3. In section 4 five sources of information were identified that may be used by these methods: dictionary preference, pseudo frequencies, parallel corpora, context in noun phrases and human expertise. Given the five information sources we identified seven (two experiments were done both with and without normalisation) basic retrieval experiments or runs that are listed in table 1. Each experiment is labelled with a letter from *a* to *g*.

run name	technique to weight translations / pick the best translation
run?a	no weights used / dictionary preferred translation.
run?b	weight by pseudo frequencies.
run?c	normalise weights of possible translations (run?a)
run?d	weight by normalised pseudo frequencies
run?e	normalised 'real' frequencies estimated from the parallel Agenda 21 corpus.
run?f	weight by using noun phrases from documents (including normalisation)
run?g	disambiguation by a human expert

Table 1: information to weight translations and / or pick the best translation

The combinations of seven information sources and three methods define a total number of 21 possible experiments. After removing combinations that are redundant or not informative 15 experiments remain.

In the remainder of this section we will report the results of 15 experiments on the TREC CLIR task test collection [11] topics 1-24 (excluding the topics that were not judged at the time of TREC-6 leaving 21 topics). The Dutch topics will be used to search the English documents. Experiments will be compared by means of their non-interpolated average precision, average precision in short. Additionally, the result of each experiment will be compared with the result of a monolingual base line run, which is the result of queries based on the English version of the TREC topics. The monolingual run performs at an average precision of 0.403. All experiments were done with the linguistically motivated experimental retrieval engine developed at the University of Twente.

5.1 One translation runs

Table 2 list the results of the one translation runs. Normalisation of translation weights is not useful for picking the best translation. Therefore the table does not list **run1c** and **run1d**. (**run1d** would give exactly the same results as **run1b**.)

run name	average precision	relative to baseline (%)
run1a	0.262	65
run1b	0.231	57
run1e	0.282	70
run1f	0.269	67
run1g	0.315	78

Table 2: results of 'one-translation' runs

Not surprisingly, the manual disambiguated run outperforms the automatic runs, but it still performs at 78 % of the monolingual run. Translation ambiguity and missing terminology are the two primary sources of error in CLIR [7]. We hypothesise that the loss of performance is due to missing terminology and possibly errors in the translation scripts. The 78 % performance of the monolingual base line is an upper bound on what is possible using a one-translation approach on the TREC CLIR collection.

The best automatic run is the run using corpus frequencies **run1e**. This is a surprise, because we used a relatively small corpus on the domain of the Twenty-One demonstrator which is *sustainable development*. Inspection of the topics however learns us that a lot of topics discuss international problems like air pollution, combating AIDS, etc. which fall directly in the domain of sustainable development.

The dictionary preferred run **run1a** performs reasonable well. The run using context from noun phrases **run1f** performs only a little better. Pseudo frequencies **run1b** are less useful in identifying the correct translation.

5.2 Unstructured query runs

Table 3 list the results of the unstructured query runs using all possible translations of each original query term. We experimented with all information sources except for the human expert.

run name	average precision	relative to baseline (%)
run2a	0.180	45
run2b	0.162	40
run2c	0.268	67
run2d	0.308	76
run2e	0.305	76
run2f	0.275	68

Table 3: results of 'unstructured query' runs

A first important thing to notice is that the normalisation of the term weights is a prerequisite for good performance if all possible translations per source language query term are used in an unstructured query. Not using the normalisation, as done in **run2a** and **run2b** will drop performance to a disappointing 40 to 45 per cent of the monolingual base line.

More surprisingly, the pseudo frequency run **run2d** and the real frequency run **run2e** now perform equally well and both approach the upper bound on what is possible with the one translation approach (**run1g**). Although the pseudo frequencies are not very useful for identifying the best translation, they seem to be as realistic as real frequencies if used for weighting the possible translations.

5.3 STRUCTURED QUERY RUNS

Table 4 lists the results of the structured query runs. Normalisation of term weights is implicit in the structured query, so **run3a** and **run3b** will give exactly the same results as **run3c** and **run3d** respectively.

run name	average precision	relative to baseline (%)
run3c	0.311	77
run3d	0.330	82
run3e	0.335	83
run3f	0.323	80

Table 4: results of 'structured query' runs

The four runs do not differ as much in performance as their unstructured equivalents, which suggests that the structured queries are more robust than the unstructured queries. Again, the pseudo frequency run **run2d** and the real frequency run **run2e** perform almost equally well. Three out of four runs perform better than the manually disambiguated 'one translation' run

run1g.

6 CONCLUSION

This paper gives an overview of methods and information sources that can be used for CLIR. Evaluation of these methods on the TREC cross-language collection indicates that using all possible translations for searching leads to better retrieval performance in terms of average precision than using just one translation. The results of the manually disambiguated run suggest that not much can be gained by putting a lot of effort in explicit disambiguation of possible translations. If proper weighting of possible translations is used, disambiguation is done implicitly during searching.

This paper briefly introduced a new method to rank document using structured queries. Mathematical details of the method will be published in the near future. In the cross-language experiments reported on here, structured queries outperform the unstructured queries.

REFERENCES

[1] Rens Bod. *Enriching Linguistics with Statistics: Performance Models for Natural Language.* Academische Pers, 1995.

[2] D. Hiemstra. A linguistically motivated probabilistic model of information retrieval. In C. Nicolaou and C. Stephanidis, editors, *Proceedings of the Second European Conference on Research and Advanced Technology*

for Digital Libraries, ECDL-2, pages 569–584, 1998.

[3] D. Hiemstra. Multilingual domain modeling in Twenty-One: automatic creation of a bi-directional translation lexicon from a parallel corpus. In *Proceedings of eightth CLIN meeting*, 1998.

[4] D. Hiemstra, F.M.G. de Jong, and W. Kraaij. A domain specific lexicon acquisition tool for cross-language information retrieval. In *Proceedings of RIAO'97 Conference on Computer-Assisted Searching on the Internet*, pages 255–266, 1997.

[5] D. Hiemstra and W. Kraaij. Trec-7 working notes: Twenty-One in ad-hoc and clir. In *Proceedings of the seventh Text Retrieval Conference, TREC-7*, (draft, to appear).

[6] D.A. Hull. Using structured queries for disambiguation in cross-language information retrieval. In *AAAI Symposium on Cross-Language Text and Speech Retrieval*. American Association for Artificial Intelligence, 1997.

[7] David Hull and Gregory Grefenstette. A dictionary-based approach to multilingual information retrieval. In *Proceedings of the 19th ACM SIGIR Conference on Research and Development in Information Retrieval*, 1996.

[8] W. Kraaij and D. Hiemstra. Cross-language retrieval with the Twenty-One system. In E. Voorhees and D. Harman, editors, *Proceedings of the 6th Text Retrieval Conference TREC-6*, pages 753–761. NIST Special Publication 500-240, 1998.

[9] S.E. Robertson and K. Sparck Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27:129–146, 1976.

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

[11] E.M. Voorhees and D.K. Harman. Overview of the 6th text retrieval conference. In *Proceedings of the 6th Text Retrieval Conference TREC-6*, pages 1–24. NIST Special Publication 500-240, 1998.