Designing a Tool for Exploiting Bilingual Comparable Corpora

Peter Bennison and Lynne Bowker

Dublin City University
Dublin 9, Ireland
pbenn@hotmail.com, lynne.bowker@dcu.ie

Abstract

Translators have a real need for a tool that will allow them to exploit information contained in bilingual comparable corpora. ExTrECC is designed to be a semi-automatic tool that processes bilingual comparable corpora and presents a translator with a list of potential equivalents (in context) of the search term. The task of identifying translation equivalents in a non-aligned, non-translated corpus is a difficult one, and ExTrECC makes use of a number of techniques, some of which are simple and others more sophisticated. The basic design of ExTrECC (graphical user interface, architecture, algorithms) is outlined in this paper.

1. Introduction

Translators have long been aware that dictionaries and other lexicographic resources have a number of shortcomings when it comes to identifying appropriate translation equivalents. Recent research (e.g. Schäffner, 1998; Williams, 1996) has shown that, as a translation resource, text-based data offers many advantages over lexicographic resources, which are often incomplete and decontextualized and hence users may not comprehend or use the terms correctly. Therefore, practising translators often consult **comparable** texts in the target language (i.e., the language into which they are translating) in order to find appropriate terms and usage information. A comparable text is one that has the same communicative function as the source text (i.e., the text to be translated). In other words, it deals with the same subject matter and is of the same text type, but it has been originally written in the target language. For example, if a translator was translating a computer manual from English into French, then he or she might find it beneficial to consult an existing French computer manual to determine appropriate vocabulary, syntax, register, style, etc.

Although the value of text-based resources is clear (i.e., they are more up-to-date and provide more contextual information), the drawback associated with paper-based comparable texts is that they are difficult and time-consuming for translators to consult. Unlike in a dictionary, the terms are not presented in alphabetical order but are found embedded in running text. The challenge lies in finding a way to access the required information in a systematic and semi-automatic way.

One active research area at the intersection of translation and computing is the investigation of the potential of electronic corpora to act as translation or terminology resources (e.g. Bowker, 1996; Meyer and Mackintosh, 1996; Pearson, 1998). A corpus is a large collection of machine-readable texts that can be manipulated and interrogated with the help of a computer. To date, however, most of the work has been carried out on monolingual corpora. This has provided an interesting start, but translators actually have a greater need for bilingual information.

Another area of active research is in the alignment and extraction of translation equivalents from **parallel** corpora (i.e., corpora containing source texts and their translations). While this research is interesting for studying the translation process, and is undoubtedly useful in the development of machine translation systems, it is of limited use as a resource for human translators because translated texts do not share the full range of linguistic features of authentic texts produced in the target language. Translators are very hesitant to rely on translated material as an authentic resource – what translators actually need are bilingual **comparable** corpora and tools for accessing the information contained within them. A bilingual comparable corpus essentially consists of two monolingual corpora that are similar with regard to subject matter, text type and publication date. A tool for processing bilingual comparable corpora must be able to identify translation equivalents in two corpora that are non-aligned and non-translated.

The aim of this paper is to outline the design of such a tool, referred to as ExTrECC (Extraction of Translation Equivalents from Comparable Corpora), which is currently under development at Dublin City University. The paper will be divided into five main sections. Section 2 discusses the design and compilation of a bilingual comparable corpus. Section 3 provides a design overview. Section 4 outlines the user interface considerations. Section 5 focuses on system architecture. Finally, section 6 discusses algorithms under consideration for system implementation.

2. Bilingual Comparable Corpus Design and Compilation

In order to extract useful information from a corpus, considerable thought must be given to the design of the corpus. It is essential that the two monolingual elements of a bilingual comparable corpus (BCC) be comparable in terms of size, text type and publication date.

Most translation deals with specialized rather than general language, and for the development and initial testing of ExTrECC, we have constructed a BCC in the specialized field of computer viruses. The languages being used for prototype development of the system are French and English, but in principle, this tool can be modified to work with other language pairs.

Both the French subcorpus and the English subcorpus contain fifty texts, and each subcorpus has a total word count of between thirty and thirty-five thousand words. The texts are all articles about computer viruses that have been taken from semi-specialized computer magazines (e.g. *Informatiques magazine*, *PC Direct*, *Information Week*). The corpus covers a five-year period and all the texts were published between January 1995 and December 1999.

3. ExTrECC: Design Overview

ExTrECC is intended to be a semi-automatic tool that processes bilingual comparable corpora and presents a translator with a list of potential equivalents (in context) of the search term. The task of identifying translation equivalents in a non-aligned, non-translated corpus is non-trivial, and ExTrECC makes use of a number of techniques, some of which are simple and others more sophisticated. The following sections describe the design of the tool, including the graphical user interface (GUI), the architecture and the core algorithms used. It is important to note that this tool is a prototype that is currently under active development. Consequently, some aspects of the application may evolve over time. For example, the algorithms used to calculate and analyze word co-occurrence measures will initially be simple, but later on, more sophisticated statistics will be introduced, along with other approaches that do not rely on co-occurrence measures. The design presented here does not explicitly distinguish features existing in the prototype implementation from those features that will be added to the tool at a later stage.

4. Graphical User Interface (GUI)

The primary purpose of the user interface is to allow the user to enter a search term in the source language and to present this user with a list of potential equivalents in the target language. ExTrECC is designed to assist translators, not to replace them. It is important to stress that this tool provides a selection of potential equivalents rather than one absolute equivalent because the algorithms discussed below do not provide definitive mappings between source and target language terms. Rather, it is up to the translator to assess which target language term or expression is suitable for his or her purposes.

ExTrECC is a bi-directional tool, meaning that either language can be used as source or target. For any given search, the user must first specify which language is to be used as the source language. Next, the user enters a search term, which is then displayed in a KWIC concordance.

ExTrECC will then list (in real time) the highest-ranking candidates in the target language. Double clicking on a specific term in the candidate list will generate a second KWIC concordance – this time for the specified target language term. Double clicking on a given line in either KWIC concordance will expand the amount of text surrounding the term.

The GUI will also allow the user to adjust display properties (e.g. context window size) and adjust parameters used by the algorithms, where appropriate. It will also allow the translator to load and preprocess new corpora and lexicons. This means that ExTrECC will be independent of any particular corpus or subject field.

5. Architecture

The architecture of a software application describes the structure of its components and how they interact with one another. ExTrECC has the following parts and capabilities: GUI, preprocessor, search engine, persistence module, Internet capability.

5.1. GUI

The GUI controls all aspects of the application, from the management of corpora and lexicons to the entry of source language terms and display of target language expressions. The GUI can be used to restrict access to some administrative functions (i.e., to prevent novice users from entering incorrect or damaging information).

5.2. Preprocessor

The preprocessor will load and analyze the two monolingual subcorpora of the BCC and an optional lexicon, producing information that is independent of any query. For example, the preprocessor will construct word co-occurrence matrices and word frequency vectors from each subcorpus. The search engine will use these data structures when specific source language terms are entered into the tool. This preprocessing step is vital in order for the application to respond to user requests in a timely fashion.

The lexicon will be an optional component of the tool. If it is absent, the preprocessing algorithms will still work, but the quality of the proposals for target language equivalents may suffer. If the lexicon exists, and is similar in theme to the BCC loaded by the preprocessor, the algorithms described below will produce better results.

5.3. Search Engine

The search engine scans the preprocessor data structures to construct a list of target language terms or expressions that may be a match for the source language term. As an optional feature, the successful candidates ('successful' means they are selected by the end user as acceptable translations of the source term) may be added to lexicon associated with the BCC in question. This updated lexicon can then be used in future preprocessing or searches.

5.4. Persistence Module

The persistence module will store the results of the preprocessor and search engine. This will allow ExTrECC to avoid unnecessary processing on start-up and will enhance the quality of search results by using the selections from previous queries.

5.5. Internet capability

Internet capability has important architectural considerations. The Internet version of ExTrECC consists of a web server that will store multiple corpora and lexicons. Internet users will be able to submit queries using a downloaded applet. The attraction of this architecture is that the results from potentially many translators will be stored by the persistence module on the web server, enhancing the quality of search results for all users. In addition, this distributed approach minimizes the amount of information that needs to be stored locally on a user's computer. It also makes it easier for an ExTrECC administrator to manage the corpora and lexicons: the administrator only needs to update and preprocess data on the web server in order for the information to be available to all users.

6. Algorithms

The procedures used by ExTrECC during preprocessing and searching are described here. The main premise of the algorithms described below is that there is a correlation between patterns of word co-occurrences in texts of different languages (Rapp, 1995). This means that regardless of the language combination and text, terms and expressions are used in largely the same way in comparable subcorpora.

6.1. Preprocessing

A certain amount of preprocessing of the BCC is necessary before running ExTrECC. First, the two subcorpora must be part-of-speech (POS) tagged. This is done outside of ExTrECC by a tagger.

The next step is referred to as matching. The aim here is to identify those words in the two subcorpora that have the same spelling or similar stems in both languages. These will later be presented as possible candidates of each other.

During the following stage, word frequency and co-occurrence measures for all the words in each subcorpus will be computed. Frequency is the number of times a word appears in a corpus. An example of a co-occurrence measure is one that considers two words as ‘co-occurring’ if they exist within the same sentence. More sophisticated measures are also used.

A co-occurrence matrix for all the words in each subcorpus is then constructed. This process uses the POS tagging information, matching information and lexicon (if present).

Once the co-occurrence matrix has been created, it can be used in two different ways:

As described by Rapp (1995), it is possible to randomly permute one matrix and compare the distance to other matrix. The ‘distance’ between two matrices is a measure of how different they are. The matrix should be permuted until the distance measure is below a critical value. Then, the order of the words in the two matrices will indicate a word translation list from source to target language. The strength of this approach is that it is simple and easy to program. The algorithm’s main weaknesses are that it assumes a one-to-one correspondence between source and target terms. The distance calculation may indicate similar matrices when many correspondences are incorrect. Finally, random permutations may take a long time to produce a distance measure that is acceptable.

Alternatively, Kumiko (1996) describes a process where both co-occurrence matrices are used to construct a translation matrix *T* that contains conditional probabilities that a target language word is a translation of a given source language word. The construction of the translation matrix is iterative, which means that the earlier versions of *T* will guide the creation of later versions. The final version of *T* minimizes the distance between the target language co-occurrence matrix and the matrix resulting from *T* times the source language co-occurrence matrix. This final version is used to match translation candidates between source and target language corpora. This approach has several strengths. Firstly, several words from the source language can map to a single word in the target language. Secondly, construction of *T* is iterative, meaning that the algorithm should converge. Finally, known translations (or stop lists) can be inserted as definitive (i.e., unchanging) probabilities in translation matrix.

The weaknesses of this approach are that it is more complex to program since it requires a steepest descent (SDM) or conjugate gradient algorithm to find *T* efficiently, and it may not be able to resolve ambiguities if there are several terms in target language that map to a single source language word.

There is, however, an alternative approach to co-occurrence matrices. As described in Fung and Yee (1998), it is possible to use a seed lexicon to compare occurrences of words in each corpus with the words in the lexicon. A vector for a word is constructed by recording the number of times an item in the lexicon appears in the same sentence as the word. The dimension of the vector is same as number of items in the lexicon. Word vectors are constructed for all terms in the corpus that are not in the lexicon. Similarity statistics can then be calculated using the vectors to match unknown word vectors to their counterparts in the other language. A ranking algorithm selects the best target language candidate for a source language word according to direct comparison of similarity measures (Fung and Yee, 1998). The advantage of this approach is that terms can be added to the lexicon in order to find further candidates, which will increase the number of matches of words between the source and target languages. These new matches can be added to the lexicon, allowing further matches to be found. The main disadvantage is that in order to bootstrap the process, the algorithm needs a “quality” lexicon that contains terms belonging to the same subject field as corpus. A further disadvantage is that the approach cannot deal with multi-word terms, it can only match single-word source language terms to single-word target language terms.

6.2. Search Engine

Given a source language term, the search engine need only search through the data structures produced in the preprocessing stage for potential target terms and then output these terms in an order corresponding to the most probable match. The search engine also needs to extract the surrounding context of all terms as requested by the user.

6.3. Persistence

When a user determines that a target term is an appropriate translation for a source term, this information can be stored in a lexicon for future use in analysis of co-occurrence matrices. Also, the co-occurrence matrices and translation matrix *T* can be stored so that there is no need to create these matrices the next time the tool is started.

7. Concluding Remarks

Translators are increasingly turning towards corpus-based resources to help them with their task, but to date, there are relatively few tools in existence which have been designed specifically with translators in mind. Given that translators prefer to use comparable corpora (rather than parallel corpora), there is a real need for a tool that can help them to exploit such resources. ExTrECC is a tool that aims to meet this need, and this paper has described the prototype version that is currently under active development. It is hoped that this tool will be further expanded and refined with the help of feedback from professional translators.

8. Acknowledgements

The research described here has been partially funded by an Albert College Research Fellowship from Dublin City University, Ireland.

9. References

Bowker, L. (1996). Towards a Corpus-Based Approach to Terminography. *Terminology,* 3(1), 27--52.

Fung, P. and Yee, L.Y. (1998). An IR Approach for Translating New Words from Nonparallel, Comparable Texts. In *Proceedings of the 36th Annual Meeting of the Association for Computational Lingusitics and the 17th International Conference on Computational Linguistics (COLING-ACL'98)* , Vol.1.

Tanaka, K. and Iwasaki, H. (1996). Extraction of Lexical Translations from Non-Aligned Corpora. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING-96).* Denmark: Center for Sprogteknologi.

Meyer, I. and Mackintosh, K. (1996). The Corpus from a Terminographer's Viewpoint. *International Journal of Corpus Linguistics*, 1(2), 257--285.

Pearson, J. (1998). *Terms in Context*. Amsterdam: John Benjamins Publishing Company.

Rapp, R. (1995). Identifying Word Translations in Non-Parallel Texts. In *Proceedings of 33rd Annual Meeting of the Association for Computational* Linguistics. MIT: Association for Computational Linguistics.

Schäffner, C. (1998). Parallel Texts in Translation. In L. Bowker, M. Cronin, D. Kenny and J. Pearson (Eds.), *Unity in Diversity? Current Trends in Translation Studies* (83--90). Manchester: St. Jerome Publishing.

Williams, I.A. (1996). A Translator's Reference Needs: Dictionaries or Parallel Texts? *Target* 8(2), 275-299.

9.1. Partial Bibliography of Typical Texts in the Corpus

Les nouveaux virus font muter les antivirus Computer Reseller News France , December 03, 1998

http://www.techweb.com/se/directlink.cgi?CRNF19981203S0043

Menaces : virus et utilisateurs en tete Informatiques Magazine , June 25, 1999

http://www.techweb.com/se/directlink.cgi?INF19990625S0015

Improved Virus Cures -- Vendors grapple with instant obsolescence of antivirus software

Information Week, October 27, 1997, Issue: 654

http://www.techweb.com/se/directlink.cgi?IWK19971027S0057

New Type of PC Virus Continually Updates Itself Via The Internet, Information Week, December 13, 1999, Issue: 765

http://www.techweb.com/se/directlink.cgi?IWK19991213S0032