

Using comparable corpora to solve problems difficult for human translators

Serge Sharoff, Bogdan Babych, Anthony Hartley

Centre for Translation Studies

University of Leeds, LS2 9JT UK

{s.sharoff,b.babych,a.hartley}@leeds.ac.uk

Abstract

In this paper we present a tool that uses comparable corpora to find appropriate translation equivalents for expressions that are considered by translators as difficult. For a phrase in the source language the tool identifies a range of possible expressions used in similar contexts in target language corpora and presents them to the translator as a list of suggestions. In the paper we discuss the method and present results of human evaluation of the performance of the tool, which highlight its usefulness when dictionary solutions are lacking.

1 Introduction

There is no doubt that both professional and trainee translators need access to authentic data provided by corpora. With respect to polysemous lexical items, bilingual dictionaries list several translation equivalents for a headword, but words taken in their contexts can be translated in many more ways than indicated in dictionaries. For instance, the Oxford Russian Dictionary (ORD) lacks a translation for the Russian expression *исчерпывающий ответ* ('comprehensive answer'), while the Multitran Russian-English dictionary suggests that it can be translated as *irrefragable answer*. Yet this expression is extremely rare in English; on the Internet it occurs mostly in pages produced by Russian speakers.

On the other hand, translations for polysemous words are too numerous to be listed for all possible contexts. For example, the entry for *strong* in ORD already has 57 subentries and yet it fails to mention many word combinations frequent in

the British National Corpus (BNC), such as *strong {feeling, field, opposition, sense, voice}*. *Strong voice* is also not listed in the Oxford French, German or Spanish Dictionaries.

There has been surprisingly little research on computational methods for finding translation equivalents of words from the general lexicon. Practically all previous studies have concerned detection of terminological equivalence. For instance, project Termight at AT&T aimed to develop a tool for semi-automatic acquisition of termbanks in the computer science domain (Dagan and Church, 1997). There was also a study concerning the use of multilingual webpages to develop bilingual lexicons and termbanks (Grefenstette, 2002). However, neither of them concerned translations of words from the general lexicon. At the same time, translators often experience more difficulty in dealing with such general expressions because of their polysemy, which is reflected differently in the target language, thus causing the dependency of their translation on the corresponding context. Such variation is often not captured by dictionaries.

Because of their importance, words from the general lexicon are studied by translation researchers, and comparable corpora are increasingly used in translation practice and training (Varantola, 2003). However, such studies are mostly confined to lexicographic exercises, which compare the contexts and functions of potential translation equivalents once they are known, for instance, *absolutely* vs. *assolutamente* in Italian (Partington, 1998). Such studies do not provide a computational model for *finding* appropriate translation equivalents for expressions that are not listed or are inadequate in dictionaries.

Parallel corpora, consisting of original texts and

their exact translations, provide a useful supplement to decontextualised translation equivalents listed in dictionaries. However, parallel corpora are not representative. Many of them are in the range of a few million words, which is simply too small to account for variations in translation of moderately frequent words. Those that are a bit larger, such as the Europarl corpus, are restricted in their domain. For instance, all of the 14 instances of *strong voice* in the English section of Europarl are used in the sense of ‘the opinion of a political institution’. At the same time the BNC contains 46 instances of *strong voice* covering several different meanings.

In this paper we propose a computational method for using comparable corpora to find translation equivalents for source language expressions that are considered as difficult by trainee or professional translators. The model is based on detecting frequent multi-word expressions (MWEs) in the source and target languages and finding a mapping between them in comparable monolingual corpora, which are designed in a similar way in the two languages.

The described methodology is implemented in ASSIST, a tool that helps translators to find solutions for difficult translation problems. The tool presents the results as lists of translation suggestions (usually 50 to 100 items) ordered alphabetically or by their frequency in target language corpora. Translators can skim through these lists and identify an example which is most appropriate in a given context.

In the following sections we outline our approach, evaluate the output of the prototype of ASSIST and discuss future work.

2 Finding translations in comparable corpora

The proposed model finds potential translation equivalents in four steps, which include

1. expansion of words in the original expression using related words;
2. translation of the resultant set using existing bilingual dictionaries;
3. further expansion of the set using related words in the target language;
4. filtering of the set according to expressions frequent in the target language corpus.

In this study we use several comparable corpora for English and Russian, including large reference corpora (the BNC and the Russian Reference Corpus) and corpora of major British and Russian newspapers. All corpora used in the study are quite large, i.e. the size of each corpus is in the range of 100-200 million words (MW), so that they provide enough evidence to detect such collocations as *strong voice* and *clear defiance*.

Although the current study is restricted to the English-Russian pair, the methodology does not rely on any particular language. It can be extended to other languages for which large comparable corpora, POS-tagging and lemmatisation tools, and bilingual dictionaries are available. For example, we conducted a small study for translation between English and German using the Oxford German Dictionary and a 200 MW German corpus derived from the Internet (Sharoff, 2006).

2.1 Query expansion

The problem with using comparable corpora to find translation equivalents is that there is no obvious bridge between the two languages. Unlike aligned parallel corpora, comparable corpora provide a model for each individual language, while dictionaries, which can serve as a bridge, are inadequate for the task in question, because the problem we want to address involves precisely translation equivalents that are not listed there.

Therefore, a specific query needs first to be generalised in order to then retrieve a suitable candidate from a set of candidates. One way to generalise the query is by using *similarity classes*, i.e. groups of words with lexically similar behaviour. In his work on distributional similarity (Lin, 1998) designed a parser to identify grammatical relationships between words. However, broad-coverage parsers suitable for processing BNC-like corpora are not available for many languages. Another, resource-light approach treats the context as a bag of words (BoW) and detects the similarity of contexts on the basis of collocations in a window of a certain size, typically 3-4 words, e.g. (Rapp, 2004). Even if using a parser can increase precision in identification of contexts in the case of long-distance dependencies (e.g. *to cook Alice a whole meal*), we can find a reasonable set of relevant terms returned using the BoW approach, cf. the results of human evaluation for English and German by (Rapp, 2004).

For each source word s_0 we produce a list of similar words: $\Theta(s_0) = s_1, \dots, s_N$ (in our tool we use $N = 20$ as the cutoff). Since lists of distributionally words can contain words irrelevant to the source word, we filter them to produce a more reliable similarity class $S(s_0)$ using the assumption that the similarity classes of similar words have common members:

$$\forall w \in S(s_0), w \in \Theta(s_0) \& w \in \bigcup \Theta(s_i)$$

This yields for *experience* the following similarity class: *knowledge, opportunity, life, encounter, skill, feeling, reality, sensation, dream, vision, learning, perception, learn*.¹ Even if there is no requirement in the BoW approach that words in the similarity class are of the same part of speech, it happens quite frequently that most words have the same part of speech because of the similarity of contexts.

2.2 Query translation and further expansion

In the next step we produce a translation class by translating all words from the similarity class into the target language using a bilingual dictionary ($T(w)$ for the translation of w). Then for Step 3 we have two options: a full translation class (TF) and a reduced one (TR).

TF consists of similarity classes produced for all translations: $S(T(S(s_0)))$. However, this causes a combinatorial explosion. If a similarity class contains N words (the average figure is 16) and a dictionary lists on average M equivalents for a source word (the average figure is 11), this procedure outputs on average $M \times N^2$ words in the full translation class. For instance, the complete translation class for *experience* contains 998 words. What is worse, some words from the full translation class do not refer to the domain implied in the original expression because of the ambiguity of the translation operation. For instance, the word *dream* belongs to the similarity class of *experience*. Since it can be translated into Russian as *сказка* ('fairy-tale'), the latter Russian word will be expanded in the full translation class with words referring to legends and stories. In the later stages of the project, word sense disambiguation in corpora could improve precision of translation classes. However at the present stage we attempt to trade the recall of the tool for greater precision by translating words in the source similarity class,

¹Ordered according to the score produced by the Singular Value Decomposition method as implemented by Rapp.

and generating the similarity classes of translations only for the source word:

$$TR(s_0) = S(T(s_0)) \cup T(S(s_0)).$$

This reduces the class of *experience* to 128 words.

This step crucially relies on a wide-coverage machine readable dictionary. The bilingual dictionary resources we use are derived from the source file for the Oxford Russian Dictionary, provided by OUP.

2.3 Filtering equivalence classes

In the final step we check all possible combinations of words from the translation classes for their frequency in target language corpora.

The number of elements in the set of theoretically possible combinations is usually very large: $\prod T_i$, where T_i is the number of words in the translation class of each word of the original MWE. This number is much larger than the set of word combinations which is found in the target language corpora. For instance, *daunting experience* has 202,594 combinations for the full translation class of *daunting experience* and 6,144 for the reduced one. However, in the target language corpora we can find only 2,256 collocations with frequency > 2 for the full translation class and 92 for the reduced one.

Each theoretically possible combination is generated and looked up in a database of MWEs (which is much faster than querying corpora for frequencies of potential collocations). The MWE database was pre-compiled from corpora using a method of filtering, similar to part-of-speech filtering suggested in (Justeson and Katz, 1995): in corpora each N-gram of length 2, 3 and 4 tokens was checked against a set of filters.

However, instead of pre-defined patterns for entire expressions our filtering method uses sets of *negative* constraints, which are usually applied to the edges of expressions. This change boosts recall of retrieved MWEs and allows us to use the same set of patterns for MWEs of different length. The filter uses constraints for both lexical and part-of-speech features, which makes configuration specifications more flexible.

The idea of applying a negative feature filter rather than a set of positive patterns is based on the observation that it is easier to describe undesirable features than to enumerate complete lists of patterns. For example, MWEs of any length ending with a preposition are undesirable (particles in

	British news	Russian news
no of words	217,394,039	77,625,002
REs in filter	25	18
2-grams	6,361,596	5,457,848
3-grams	14,306,653	11,092,908
4-grams	19,668,956	11,514,626

Table 1: MWEs in News Corpora

phrasal verbs, which are desirable, are tagged differently by the Tree Tagger, so there is no problem with ambiguity here). Our filter captures this fact by having a negative condition for the right edge of the pattern (regular expression `/_IN$/`), rather than enumerating all possible configurations which do not contain a preposition at the end. In this sense the filter is permissive: everything that is not explicitly forbidden is allowed, which makes the description more economical.

The same MWE database is used for checking frequencies of multiword collocates for corpus queries. For this task, candidate N-grams in the vicinity of searched patterns are filtered using the same regular expression grammar of MWE constraints, and then their corpus frequency is checked in the database. Thus scores for multiword collocates can be computed from contingency tables similarly to single-word collocates.

In addition, only MWEs with a frequency higher than 1 are stored in the database. This filters out most expressions that co-occur by chance. Table 1 gives an overview of the number of MWEs from the news corpus which pass the filter. Other corpora used in ASSIST (BNC and RRC) yield similar results. MWE frequencies for each corpus can be checked individually or joined together.

3 Evaluation

There are several attributes of our system which can be evaluated, and many of them are crucial for its efficient use in the workflow of professional translators, including: usability, quality of final solutions, trade-off between adequacy and fluency across usable examples, precision and recall of potentially relevant suggestions, as well as real-text evaluation, i.e. “What is the coverage of difficult translation problems typically found in a text that can be successfully tackled?”

In this paper we focus on evaluating the quality of potentially relevant translation solutions, which is the central point for developing and calibrat-

ing our methodology. The evaluation experiment discussed below was specifically designed to assess the usefulness of translation suggestions generated by our tool – in cases where translators have doubts about the usefulness of dictionary solutions. In this paper we do not evaluate other equally important aspects of the system’s functionality, which will be the matter of future research.

3.1 Set-up of the experiment

For each translation direction we collected ten examples of possibly recalcitrant translation problems – words or phrases whose translation is not straightforward in a given context. Some of these examples were sent to us by translators in response to our request for difficult cases. For each example, which we included in the evaluation kit, the word or phrase either does not have a translation in ORD (which is a kind of a baseline standard reference for Russian translators), or its translation has significantly lower frequency in a target language corpus in comparison to the frequency of the source expression. If an MWE is not listed in available dictionaries, we produced compositional (word-for-word) translations using ORD. In order to remove a possible anti-dictionary bias from our experiment, we also checked translations in Multitran, an on-line translation dictionary, which was often quoted as one of the best resources for translation from and into Russian.

For each translation problem five solutions were presented to translators for evaluation. One or two of these solutions were taken from a dictionary (usually from Multitran, and if available and different, from ORD). The other suggestions were manually selected from lists of possible solutions returned by ASSIST. Again, the criteria for selection were intuitive: we included those suggestions which made best sense in the given context. Dictionary suggestions and the output of ASSIST were indistinguishable in the questionnaires to the evaluators. The segments were presented in sentence context and translators had an option of providing their own solutions and comments. Table 2 shows one of the questions sent to evaluators. The problem example is *четкая программа* (‘precise programme’), which is presented in the context of a Russian sentence with the following (non-literal) translation *This team should be put together by responsible politicians, who have a*

Problem example	
четкая программа, as in	
Собрать эту команду должны ответственные люди, имеющие четкую программу выхода из кризиса.	
Translation suggestions	Score
clear plan	
clear policy	
clear programme	
clear strategy	
concrete plan	
Your suggestion ? (optional)	

Table 2: Example of an entry in questionnaire

clear strategy for resolving the current crisis. The third translation equivalent (*clear programme*) in the table is found in the Multitran dictionary (ORD offers no translation for *четкая программа*). The example was included because *clear programme* is much less frequent in English (2 examples in the BNC) in comparison to *четкая программа* in Russian (70). Other translation equivalents in Table 2 are generated by ASSIST.

We then asked professional translators affiliated to a translator’s association (identity withheld at this stage) to rate these five potential equivalents using a five-point scale:

- 5 = The suggestion is an appropriate translation as it is.
- 4 = The suggestion can be used with some minor amendment (e.g. by turning a verb into a participle).
- 3 = The suggestion is useful as a hint for another, appropriate translation (e.g. suggestion *elated* cannot be used, but its close synonym *exhilarated* can).
- 2 = The suggestion is not useful, even though it is still in the same domain (e.g. *fear* is proposed for a problem referring to *hatred*).
- 1 = The suggestion is totally irrelevant.

We received responses from eight translators. Some translators did not score all solutions, but there were at least four independent judgements for each of the 100 translation variants. An example of the combined answer sheet for all responses to the question from Table 2 is given in Table 3 (t1,

Translation	t1	t2	t3	t4	t5	σ
clear plan	5	5	3	4	4	0.84
clear policy	5	5	3	4	4	0.84
clear programme	5	5	3	4	4	0.84
clear strategy	5	5	5	5	5	0.00
concrete plan	1	5	3	3	5	1.67
Best Dict	5	5	3	4	4	0.84
Best Syst	5	5	5	5	5	0.00

Table 3: Scores to translation equivalents

t2,... denote translators; the dictionary translation is *clear programme*).

3.2 Interpretation of the results

The results were surprising in so far as for the majority of problems translators preferred very different translation solutions and did not agree in their scores for the same solutions. For instance, *concrete plan* in Table 3 received the score 1 from translator t1 and 5 from t2.

In general, the translators very often picked up on different opportunities presented by the suggestions from the lists, and most suggestions were equally legitimate ways of conveying the intended content, cf. the study of legitimate translation variation with respect to the BLEU score in (Babych and Hartley, 2004). In this respect it may be unfair to compute average scores for each potential solution, since for most interesting cases the scores do not fit into the normal distribution model. So averaging scores would mask the potential usability of really inventive solutions.

In this case it is more reasonable to evaluate two *sets* of solutions – the one generated by ASSIST and the other found in dictionaries – but not each solution individually. In order to do that for each translation problem the best scores given by each translator in each of these two sets were selected. This way of generalising data characterises the general quality of suggestion sets, and exactly meets the needs of translators, who collectively get ideas from the presented sets rather than from individual examples. This also allows us to measure inter-evaluator agreement on the *dictionary* set and the *ASSIST* set, for instance, via computing the standard deviation σ of absolute scores across evaluators (Table 3). This appeared to be a very informative measure for dictionary solutions.

In particular, standard deviation scores for the *dictionary* set (threshold $\sigma = 0.5$) clearly split

Agreement: σ for dictionary ≤ 0.5				
Example	Dict		ASSIST	
	Ave	σ	Ave	σ
political upheaval	4.83	0.41	4.67	0.82
Disagreement: σ for dictionary >0.5				
Example	Dict		ASSIST	
	Ave	σ	Ave	σ
clear defiance	4.14	0.90	4.60	0.55

Table 4: Examples for the two groups

Agreement: σ for dictionary ≤ 0.5				
Sub-group	Dict		ASSIST	
	Ave	σ	Ave	σ
Agreement E→R	4.73	0.46	4.47	0.80
Agreement R→E	4.90	0.23	4.52	0.60
Agreement–All	4.81	0.34	4.49	0.70
Disagreement: σ for dictionary >0.5				
Sub-group	Dict		ASSIST	
	Ave	σ	Ave	σ
Disagreement E→R	3.63	1.08	3.98	0.85
Disagreement R→E	3.90	1.02	3.96	0.73
Disagreement–All	3.77	1.05	3.97	0.79

Table 5: Averages for the two groups

our 20 problems into two distinct groups: the first group below the threshold contains 8 examples, for which translators typically agree on the quality of dictionary solutions; and the second group above the threshold contains 12 examples, for which there is less agreement. Table 4 shows some examples from both groups and Table 5 presents average evaluation scores and standard deviation figures for both groups.

Overall performance on all 20 examples is the same for the dictionary responses and for the system’s responses: average of the mean top scores is about 4.2 and average standard deviation of the scores is 0.8 in both cases (for set-best responses). This shows that ASSIST can reach the level of performance of a combination of two authoritative dictionaries for MWEs, while for its own translation step it uses just a subset of one-word translation equivalents from ORD. However, there is another side to the evaluation experiment. In fact, we are less interested in the system’s performance on all of these examples than on those examples for which there is greater disagreement among translators, i.e. where there is some degree of dissatisfaction with dictionary suggestions.

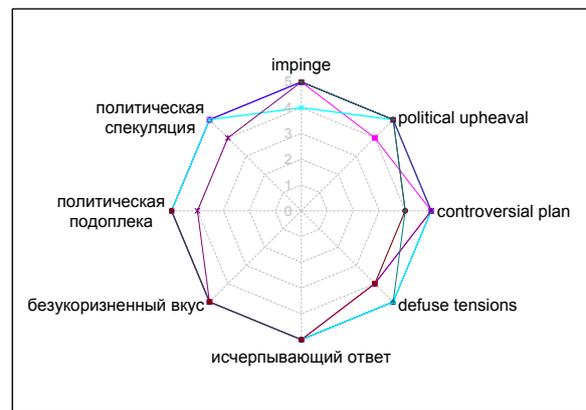


Figure 1: Agreement scores: dictionary

Interestingly, dictionary scores for the agreement group are always higher than 4, which means that whenever translators agreed on the dictionary scores they were usually satisfied with the dictionary solution. But they never agreed on the inappropriateness of the dictionary: inappropriateness revealed itself in the form of low scores from *some* translators.

This agreement/disagreement threshold can be said to characterise two types of translation problems: those for which there exist generally accepted dictionary solutions, and those for which translators doubt whether the solution is appropriate. Best-set scores for these two groups of dictionary solutions – the agreement and disagreement group – are plotted on the radar charts in Figures 1 and 2 respectively. The identifiers on the charts are problematic source language expressions as used in the questionnaire (not translation solutions to these problems, because a problem may have several solutions preferred by different judges). Scores for both translation directions are presented on the same chart, since both follow the same pattern and receive the same interpretation.

Figure 1 shows that whenever there is little doubt about the quality of dictionary solutions, the radar chart approaches a circle shape near the edge of the chart. In Figure 2 the picture is different: the circle is disturbed, and some scores frequently approach the centre. Therefore the disagreement group contains those translation problems where dictionaries provide little help.

The central problem in our evaluation experiment is whether ASSIST is helpful for problems in the second group, where translators doubt the quality of dictionary solutions.

Firstly, it can be seen from the charts that judge-

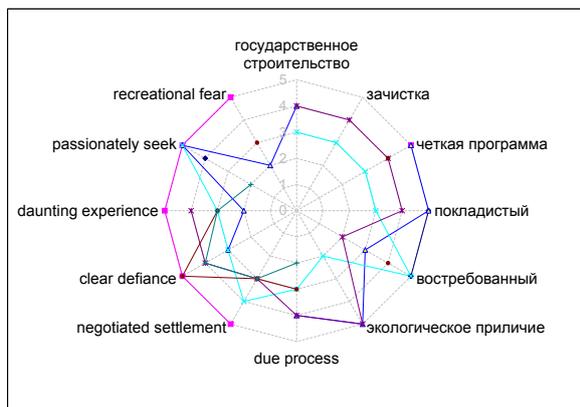


Figure 2: Disagreement scores: dictionary

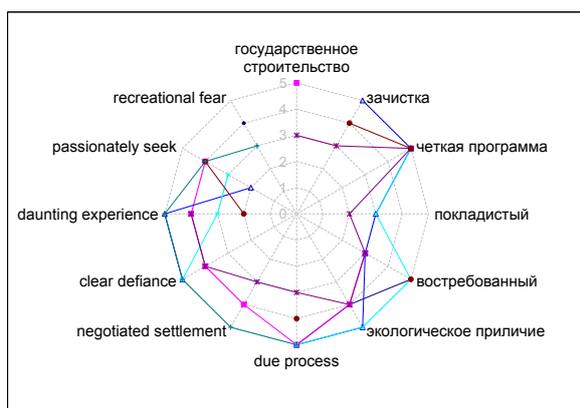


Figure 3: Disagreement scores: ASSIST

ments on the quality of the system output are more consistent: score lines for system output are closer to the circle shape in Figure 1 than those for dictionary solutions in Figure 2 (formally: the standard deviation of evaluation scores, presented in Table 4, is lower).

Secondly, as shown in Table 4, in this group average evaluation scores are slightly higher for ASSIST output than for dictionary solutions (3.97 vs 3.77) – in the eyes of human evaluators ASSIST outperforms good dictionaries. For good dictionary solutions ASSIST performance is slightly lower: (4.49 vs 4.81), but the standard deviation is about the same.

Having said this, solutions from our system are really not in competition with dictionary solutions: they provide less literal translations, which often emerge in later stages of the translation task, when translators correct and improve an initial draft, where they have usually put more literal equivalents (Shveitser, 1988). It is a known fact in translation studies that non-literal solutions are harder

to see and translators often find them only upon longer reflection. Yet another fact is that non-literal translations often require re-writing other segments of the sentence, which may not be obvious at first glance.

4 Conclusions and future work

The results of evaluation show that the tool is successful in finding translation equivalents for a range of examples. What is more, in cases where the problem is genuinely difficult, ASSIST consistently provides scores around 4 – “minor adaptations needed”. The precision of the tool is low, it suggests 50-100 examples with only 2-4 useful for the current context. However, recall of the output is more relevant than precision, because translators typically need just one solution for their problem, and often have to look through reasonably large lists of dictionary translations and examples to find something suitable for a problematic expression. Even if no immediately suitable translation can be found in the list of suggestions, it frequently contains a hint for solving the problem in the absence of adequate dictionary information.

The current implementation of the model is restricted in several respects. First, the majority of target language constructions mirror the syntactic structure of the source language example. Even if the procedure for producing similarity classes does not impose restrictions on POS properties, nevertheless words in the similarity class tend to follow the POS of the original word, because of the similarity of their contexts of use. Furthermore, dictionaries also tend to translate words using the same POS. This means that the existing method finds mostly NPs for NPs, verb-object pairs for verb-object pairs, etc, even if the most natural translation uses a different syntactic structure, e.g. *I like doing X* instead of *I do X gladly* (when translating from German *ich mache X gerne*).

Second, suggestions are generated for the query expression independently from the context it is used in. For instance, the words *judicial*, *military* and *religious* are in the similarity class of *political*, just as *reform* is in the simclass of *upheaval*. So the following example

The plan will protect EC-based investors in Russia from political upheavals damaging their business. creates a list of “possible translations” evoking various reforms and transformations.

These issues can be addressed by introducing a model of the semantic context of situation, e.g. ‘changes in business practice’ as in the example above, or ‘unpleasant situation’ as in the case of *daunting experience*. This will allow less restrictive identification of possible translation equivalents, as well as reduction of suggestions irrelevant for the context of the current example.

Currently we are working on an option to identify semantic contexts by means of ‘semantic signatures’ obtained from a broad-coverage semantic parser, such as USAS (Rayson et al., 2004). The semantic tagset used by USAS is a language-independent multi-tier structure with 21 major discourse fields, subdivided into 232 sub-categories (such as I1.1- = Money: lack; A5.1- = Evaluation: bad), which can be used to detect the semantic context. Identification of semantically similar situations can be also improved by the use of segment-matching algorithms as employed in Example-Based MT (EBMT) and translation memories (Planas and Furuse, 2000; Carl and Way, 2003).

The proposed model looks similar to some implementations of statistical machine translation (SMT), which typically uses a parallel corpus for its translation model, and then finds the best possible recombination that fits into the target language model (Och and Ney, 2003). Just like an MT system, our tool can find translation equivalents for queries which are not explicitly coded as entries in system dictionaries. However, from the user perspective it resembles a dynamic dictionary or thesaurus: it translates difficult words and phrases, not entire sentences. The main thrust of our system is its ability to find translation equivalents for difficult contexts where dictionary solutions do not exist, are questionable or inappropriate.

Acknowledgements

This research is supported by EPSRC grant EP/C005902.

References

Bogdan Babych and Anthony Hartley. 2004. Extending the BLEU MT evaluation method with frequency weightings. In *Proceedings of the 42^d Annual Meeting of the Association for Computational Linguistics*, Barcelona.

Michael Carl and Andy Way, editors. 2003. *Recent advances in example-based machine translation*. Kluwer, Dordrecht.

Ido Dagan and Kenneth Church. 1997. Termit: Coordinating humans and machines in bilingual terminology acquisition. *Machine Translation*, 12(1/2):89–107.

Gregory Grefenstette. 2002. Multilingual corpus-based extraction and the very large lexicon. In Lars Borin, editor, *Language and Computers, Parallel corpora, parallel worlds*, pages 137–149. Rodopi.

John S. Justeson and Slava M. Katz. 1995. Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, 1(1):9–27.

Dekang Lin. 1998. Automatic retrieval and clustering of similar words. In *Joint COLING-ACL-98*, pages 768–774, Montreal.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Alan Partington. 1998. *Patterns and meanings: using corpora for English language research and teaching*. John Benjamins, Amsterdam.

Emmanuel Planas and Osamu Furuse. 2000. Multi-level similar segment matching algorithm for translation memories and example-based machine translation. In *COLING, 18th International Conference on Computational Linguistics*, pages 621–627.

Reinhard Rapp. 2004. A freely available automatically generated thesaurus of related words. In *Proceedings of the Forth Language Resources and Evaluation Conference, LREC 2004*, pages 395–398, Lisbon.

Paul Rayson, Dawn Archer, Scott Piao, and Tony McEnery. 2004. The UCREL semantic analysis system. In *Proc. Beyond Named Entity Recognition Workshop in association with LREC 2004*, pages 7–12, Lisbon.

Serge Sharoff. 2006. Creating general-purpose corpora using automated search engine queries. In Marco Baroni and Silvia Bernardini, editors, *WaCky! Working papers on the Web as Corpus*. Gedit, Bologna.

A.D. Shveitser. 1988. Теория перевода: Статус, проблемы, аспекты. Nauka, Moscow. (In Russian: Theory of Translation: Status, Problems, Aspects).

Krista Varantola. 2003. Translators and disposable corpora. In Federico Zanettin, Silvia Bernardini, and Dominic Stewart, editors, *Corpora in Translator Education*, pages 55–70. St Jerome, Manchester.