# Adapting an Existing Example-Based Machine Translation (EBMT) System for New Language Pairs based on an Optimized Bilingual Knowledge Bank (BKB)

**Lim Huan Ngee** **Ye Hong Hoe** **Lim Chai Kim** **Tang Enya Kong**

Computer Aided Translation Unit
School of Computer Sciences
Universiti Sains Malaysia
11800 PENANG, MALAYSIA

huanngee@hotmail.com hhye@cs.usm.my chaikimlim@gmail.com enyakong@cs.usm.my

**Abstract**

Sourcing for large amount of text and translating them are some of the challenges in building an Example-Based Machine Translation (EBMT) system. These big amounts of translated texts are annotated into the S-SSTC format to cover an extensive vocabulary and sentence structures. However, the Bilingual Knowledge Bank (BKB), which is a collection of the S-SSTCs, will normally contain redundancy. Hence, the idea of an optimized BKB is born. An optimized BKB (redundancy reduced) is smaller in size but is as equally extensive in term of its sentence structure coverage compared to an un-optimized BKB. Therefore, an optimized BKB enhances the performance of the EBMT. In this paper, we introduce the idea of an optimized BKB and propose it to be re-used to effectively construct new BKBs in order to adapt an existing EBMT for new language pairs.

## 1. Introduction

The basic idea of EBMT is to translate a sentence by using similar translation examples. The original idea of EBMT is attributed to Nagao (1984). Translation examples can be collected from parallel corpus and then stored in a database. A parallel corpus is aligned when "the two texts have been analysed into corresponding segments" (Somers, 1999:150).

A BKB can be considered as a database of translation examples where examples are normally annotated with syntactic tree structures and translation units are encoded between source and target parts of the examples (Sadler and Vendelmans, 1990; Al-Adhaileh and Tang, 2001).

Correspondences between language string and its representation tree are not always straightforward. For this reason, Boitet and Zaharin (1988) argued for the need to separate language string from its representation tree, and thus proposed Structured String-Tree Correspondence (SSTC). Furthermore, Al-Adhaileh et al. (2002) proposed a flexible annotation schema (i.e. S-SSTC) that makes use of synchronous property and flexibility of SSTC to describe translation examples.

As of end of year 2006, Computer Aided Translation Unit, USM, in collaboration with various parties, managed to complete an English-Malay EBMT system that uses a large BKB constructed from text from a few different domains. It is realised that the large BKB contains unnecessary redundancy which can be eliminated to speed up the EBMT performance. Hence, initiative to identify and eliminate the redundancies

in the BKB, called optimization, is needed. We propose the optimized BKB to be used to construct new BKBs in new language pairs. We also look into the changes needed to adapt the EBMT to support new languages.

# 2. Optimization of the Existing BKB

Our BKB contains translation examples annotated with S-SSTCs. To cover an extensive vocabulary and sentence structures, a BKB needs to contain a huge amount of translation examples. One of our BKBs contains approximately 25,000 examples extracted from *Kamus Inggeris-Melayu Dewan* [4]. The most frequent problem faced in a large collection of data in the existing BKB is data redundancy where part of examples containing the same sentence structures even though it still contributes to the vocabulary coverage. Other than that, some of the sentence structures in the existing BKB do not fulfill the static grammar structure [8] which is important for a machine translation system. Static grammar structures are a collection of arbitrary rules that dictate how words may be assembled into clauses and sentences. Therefore, the idea of an optimized BKB is born. Optimized BKB is created based on the existing BKB and Static Grammar for English [8]. Static Grammar for English is a guide which contains all the static grammar structures. Though optimized BKB is smaller in size which contains only approximately 1,000 rows of data compared to the existing BKB but it is equally extensive in term of its sentence structure coverage compared to an un-optimized BKB and most importantly reduces data redundancy. The process of optimizing BKB contains two steps which will be explained in the next sub-section.

### 2.1 Process of Optimizing the Existing BKB

There are two steps involved in building an optimized BKB. Firstly, sentences which have the same structures or duplicate structures are removed to retrieve minimum set of examples and reduce redundancy. As an example consider the following sentences in figure 1:

1. He was already 28 ***when he entered his second year at university.***
2. ***When he read out my letter in front of the class***, I nearly died.
3. You put me in an invidious position ***when you made promises on my behalf***

**Figure 1**: Sentences 1, 2 and 3 have the same sentence structure (ADV [1] + PRON [2] + v [3] + N [4] + PREP [5]) in highlighted portion.

These three sentences represent the sub-structure of ADV + PRON + v + N + PREP in the existing BKB as identified by the structure index generated automatically [9]. Structure index contains a list of tree structures of source language. The tree structures can be generalized by replacing the lexical words in the nodes with parts of

[1] ADV = Adverb
[2] PRON = Pronoun
[3] v = Verb
[4] N = Noun
[5] PREP = Preposition

speech of the words. Out of the three sentences, only sentence (1) is taken to represent the sub-structure of ADV + PRON + v + N + PREP.

After the retrieval of the minimum set of examples for the new optimized BKB, it is checked against Static Grammar for English. Every example needs to be checked for its structure against static grammar structure in order to remove examples with inappropriate structures, incomplete structures or unknown structures (all this will be reconsidered later to identify new valid structures to be added again to the BKB by a linguist).

The structures can be covered by static grammar structure either partially, as a whole or covered by more than 1 static grammar structures. Besides that, static grammar structures which could not be found in the minimum set of examples are added as new examples. Only examples with proper structures and fulfill the static grammar structures are kept in the new optimized BKB.

| He | wanted | to | know | if | *there* | *were* | *any* | *white* | *ants.* |
|---|---|---|---|---|---|---|---|---|---|
| PRON | V | AU_INF[6] | V | ( | | V | ) | ( N | ) |

**Figure 2**: An example of a sentence which is partially covered by Static English Grammar (Existential Clause: there + be + NP) in highlighted portion

| *May* | *I* | *have* | *the* | *next* | *dance* | *?* |
|---|---|---|---|---|---|---|
| AU_V[7] | PRON | ( | | V subtree | | ) |

**Figure 3**: An example of a sentence which is wholly covered by Static English Grammar (Repositioning of fronted auxiliary ('not' do) to the subject) in highlighted portion

| *Do* | *not* | *wander* | *too* | *far* | *afield* | . |
|---|---|---|---|---|---|---|
| ( AU_V ) | | V | ADV | (ADV subtree) | | |

**Figure 4**: An example of a sentence which is covered by two static grammar structures which is the negation (do + not + v) and adverb phrase (ADVP[8]).

[6] AU_INF = Auxiliary infinitive
[7] AU_V = Auxiliary Verb
[8] ADVP = Adverbial phrase

Below is the workflow for the overall process of optimizing the existing BKB.

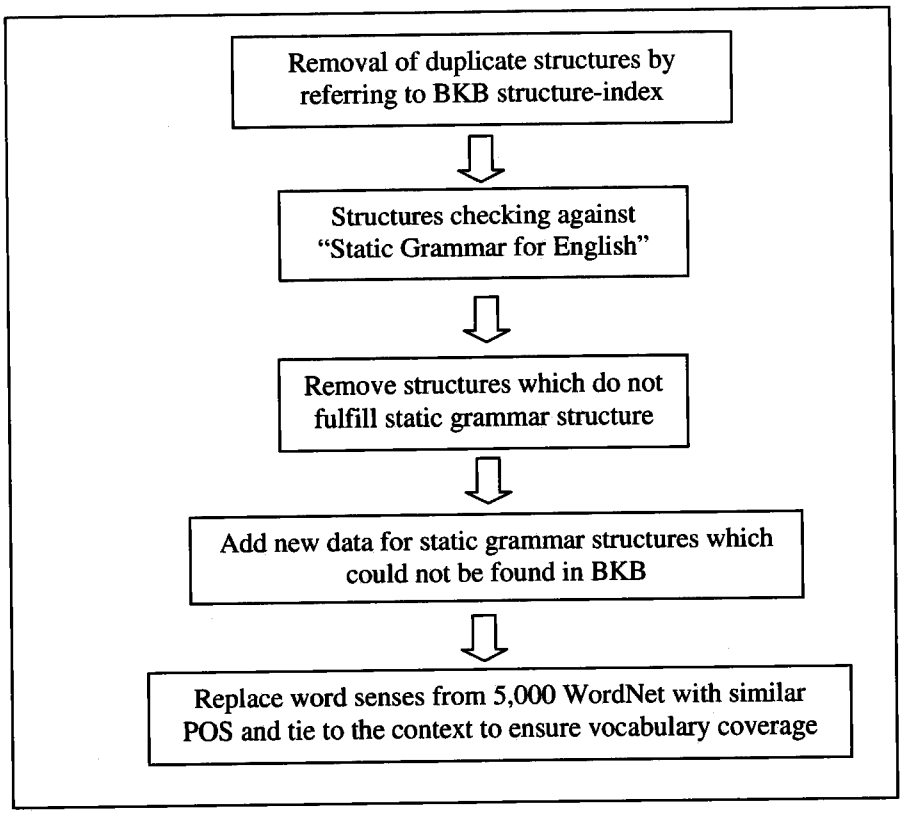


**Figure 5**: Workflow for the process of optimizing the existing BKB

## 3. Construction of a new BKB

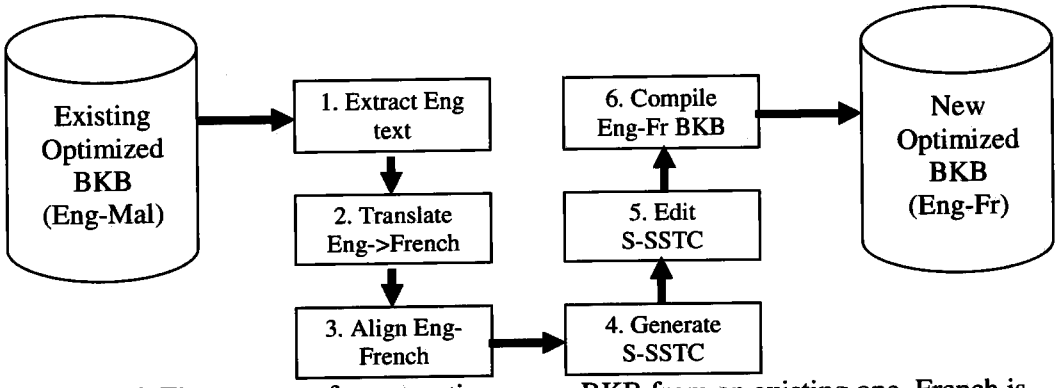


**Figure 6**: The process of constructing a new BKB from an existing one. French is taken as a new language example.

## 3.1 Extracting Source Text

As shown in Figure 6 above, the construction of a new BKB begins with the extraction of the source text from an existing optimized BKB. For example, if we have an English-Malay BKB and want to construct a new English-French BKB, we extract the English text from the existing English-Malay BKB. Constructing a new language pair with its source language that belongs to one of the existing language of the optimized BKB will speed up the whole BKB construction process for a new language pair as we can make use of all the annotation done for the source language.

## 3.2 Translating Source Text

The extracted source text is then translated into the new language we are interested in. The translation process will need human translators' expertise to ensure the quality of the translation. To speed up translation, translators may use any Machine Translation system supporting the desired language pair to provide the first draft of translation (e.g. by using Google translation tool).

## 3.3 Aligning Parallel Text

The output from the translation process is a Parallel Text. Parallel texts will need to be sentence and word aligned by someone who knows the new language pair. Sentence and Word Alignment can be done with the help of a Text Aligner tool. During the construction of the English-Malay BKB, an English-Malay Automatic Text Aligner was developed. This Text Aligner refers to a bi-text file to automatically align words in the sentence pairs. To support the automatic alignment of a new language, a new bi-text file of the new desired language pair will be needed. Besides that, the sentence and word tokenizer will need to be changed if the new language has a different way of segmenting words.

## 3.4 Generating S-SSTC

Once the Parallel Texts are properly aligned, the S-SSTC file can then be automatically generated. Automatic generation of the S-SSTCs need to refer to a Functional Dependency Grammar (FDG) server to get the tree structure of the source sentence. If the source language is a new language, the FDG server will need to be replaced with another FDG server supporting the new language. However, in the case of constructing a new BKB where the source language is one of the existing languages in the BKB, for example, constructing an English-French BKB from an English-Malay BKB, the English S-SSTCs can be reused.

## 3.5 Editing S-SSTC

Automatically generated S-SSTCs are usually not perfect. Hence, manual editing of the S-SSTCs by a human expert is needed. The editing can be done using an S-SSTC Editor. Minor changes to the S-SSTC Editor are needed to support a new character set used by the new language. Also the sentence and word tokenizer will need to be changed if the new language has a different way of segmenting words.

### 3.6 Compiling the BKB

After all the S-SSTCs are edited properly, they are imported into a database and indexed for fast reference. This process is language independent. Thus no changes to this process are needed to support a new language. With this, the construction of the new BKB is considered complete.

## 4. Adaptation of the Existing EBMT System

In this section, we will look into how the EBMT system itself can be changed to support the new language pair with its optimized BKB. The core engine of our existing English-Malay EBMT system is to a large extent language-independent. This makes the EBMT system suitable to be adapted to a new language pair without much redevelopment work.

### 4.1 Existing EBMT System

Figure 7 shows an overview of our existing EBMT system in relation to the required language tools and resources. Firstly, an input text is segmented into sentences and then words. Each word is then tagged with part of speech (POS). The system then refers to the indexed BKB and the bilingual lexicon to produce the translation as its output.

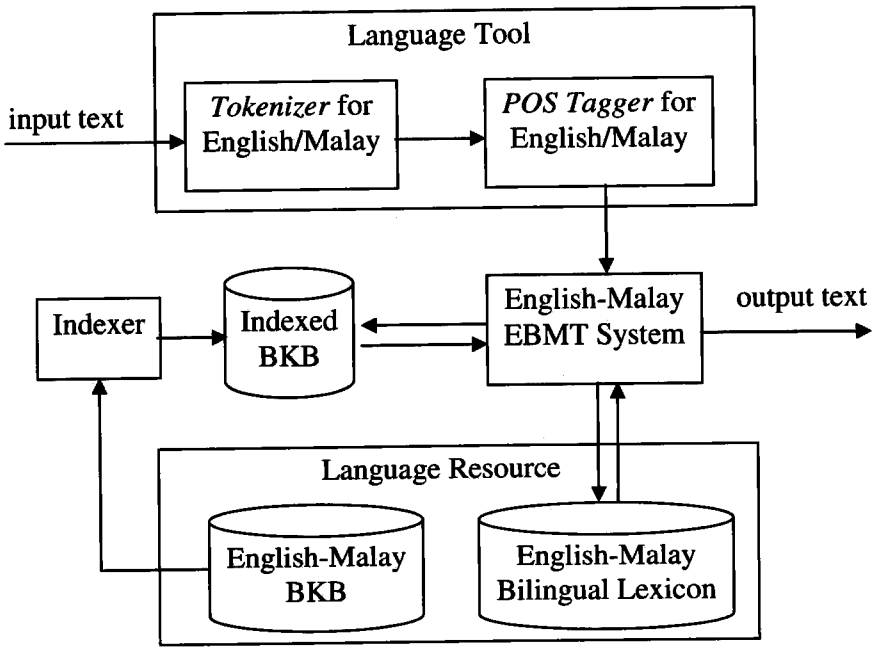


**Figure 7:** An overview of the existing English-Malay EBMT system.

### 4.2 New EBMT System

To adapt the existing EBMT system for a new language pair, the language tools and resources need to be enhanced. Figure 8 shows an overview of a new EBMT system with English-French as the example new language pair. For English→French translation, we can use the existing English tools. For French→English translation, the tokenizer may remain usable but the POS tagger needs to be replaced. Besides the tools, we need to prepare the language resources, namely the BKB and the bilingual lexicon. The BKB will contain English-French translation examples which are annotated with S-SSTCs whereas the bilingual lexicon will contain a list of source words (tagged with POS) with their translation equivalents (this data may be derived from the WordNet which consists of entries for the new language pair [10]).

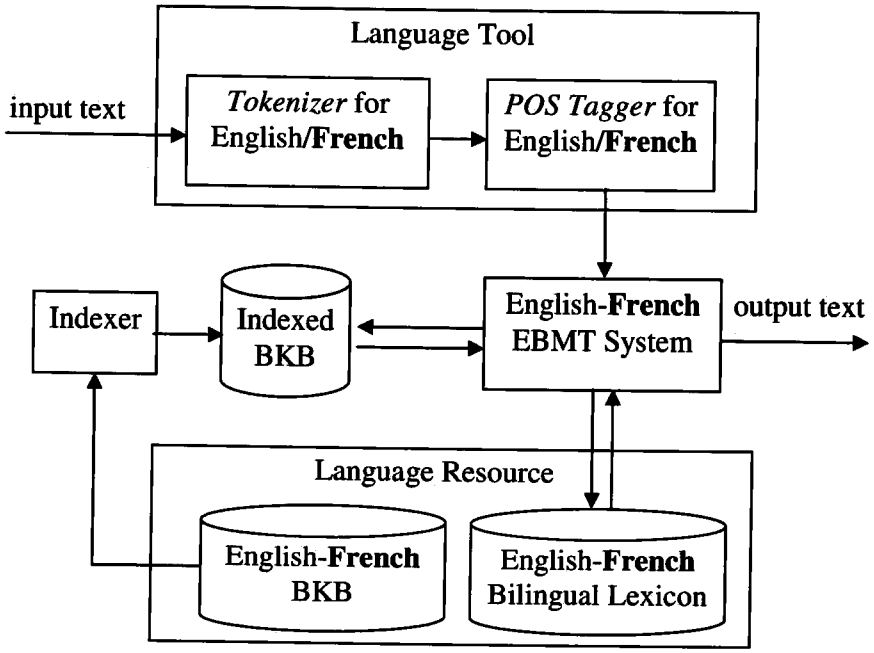


**Figure 8:** An overview of the English-French (new) EBMT system.

### 4.3 Further Improvement

Starting with the optimized BKB in the new language pair, we can improve the quality of the new EBMT system by further fine-tuning the BKB. The BKB can be fine-tuned by learning from the users' input through our Computer-Aided Translation (CAT) tool. The CAT tool is able to get translation from the EBMT system together with the S-SSTC structure (of the translation). The CAT tool also can get multiple translation equivalents from the EBMT system for each text segment (e.g. phrase, word) identified by the S-SSTC sub-structure. The CAT tool will then capture the translation selections from the users and send them to the EBMT system for learning. In addition, users can use the CAT tool to add new S-SSTCs to the BKB.

## 5. Conclusion

In this paper we introduced the idea of an optimized BKB, how the optimization is done, how the new BKB can be constructed from an existing one and how the EBMT system can be adapted to support a new language pair. With an optimized BKB, which is much smaller in size compared to the unoptimized BKB, it is possible to adapt the EBMT system to support a new language pair faster. On top of the new optimized BKB, the new EBMT system may progressively improve itself by learning from users' input with the existence of a CAT Tool that is able to capture the translation selection or new translation from users.

## References


[1] Al-Adhaileh, M. H. and Tang, E. K. (2001). Converting a Bilingual Dictionary into a Bilingual Knowledge Bank Based on the Synchronous SSTC. In *Proceedings of Machine Translation Summit VIII*, Spain, pp. 351-356.

[2] Al-Adhaileh, M. H., Tang, E. K. and Zaharin, Y. (2002). A Synchronization Structure of SSTC and its Applications in Machine Translation. In *COLING 2002 Post-Conference Workshop on Machine Translation in Asia*, Taipei, Taiwan.

[3] Boitet, C. and Zaharin, Y. (1988). Representation Trees and String-Tree Correspondences. In *COLING 1988*, Budapest, pp. 59-64.

[4] *Kamus Inggeris-Melayu Dewan*. Dewan Bahasa dan Pustaka, Kuala Lumpur, Malaysia, 2002.

[5] Nagao, M. (1984). A Framework of a Mechanical Translation between Japanese and English by Analogy Principle. In Elithorn A. and Banerji R. (eds.), *Artificial and Human Intelligence*, Amsterdam: North-Holland, pp. 173-180.

[6] Sadler, V. and Vendelmans, R. (1990). Pilot Implementation of a Bilingual Knowledge Bank. In *COLING 1990*, Helsinki, Finland, pp. 449-451.

[7] Somers, H. (1999). Review Article: Example-based Machine Translation. *Journal of Machine Translation*, pp. 113-157.

[8] *Static Grammar for English.* Computer Aided Translation Project, Universiti Sains Malaysia, Penang, Malaysia, Jan 1984.

[9] Ye, H. H. (2006). *Indexing of Bilingual Knowledge Bank Based on the Synchronous SSTC Structure*. Master's thesis, School of Computer Sciences, Universiti Sains Malaysia, Penang, Malaysia.

[10] Lim L.T., Tan E. H., Tang E. K. (2007). Digitising Dictionaries for Advanced Look-up and Lexical Knowledge Research in Malay. In 11th International Translation Conference in Kuala Lumpur, Malaysia.