Phrase Extraction for Machine Translation

Michael Rosner and Jo-Ann Bajada

Department of Artificial Intelligence, University of Malta
{mike.rosner|jbaj002}@um.edu.mt

Abstract. Statistical Machine Translation (SMT) developed in the late 1980s, based initially upon a word-to-word translation process. However, such processes have difficulties when good quality translation is not strictly word-to-word. Easy cases can be handled by allowing insertion and deletion of single words, but for more general word reordering phenomena, a more general translation process is required. There is currently much interest in phrase-to-phrase models, which can overcome this problem, but require that candidate phrases, together with their translations, be identified in the training corpora. Since phrase delimiters are not explicit, this gives rise to a new problem; that of phrase pair extraction. The current project proposes a phrase extraction algorithm which uses a window of n words around source and target words to extract equivalent phrases. The extracted phrases together with their probabilities, are used as input to an existing Machine Translation system for the purpose of evaluating the phrase extraction algorithm

1 Introduction

Machine Translation (MT) was one of the first applications envisioned for computers. As early as 1949, Warren Weaver described translation as a decoding problem. In 1954 IBM demonstrated the concept in a basic word-for-word translation system.

Interest in MT spans from the academic to the commercial domain. Over the years it has gained popularity on the web, resulting in its being one of Google's most used features. The European Union spends more than 15% of its annual budget on translation, so even partial automation of the translation process could lead to huge savings. MT uses a number of other Natural Language Processing (NLP) technologies including parsing, generation and word sense disambiguation amongst others.

Amongst the problems with MT are word order, word sense, pronoun references, tense and idioms. A number of different approaches to MT have developed over the years. The choice of approach depends on the domain for which MT is being considered. Is the system going to be used for a single or multiple languages? Is there a constrained vocabulary or will it be required to translate any text? Are there existing resources which can be used? How much time do we have to develop the system? What quality of translation is sufficient for the application at hand?

Word-for-word translation systems make use of a bilingual dictionary to translate every word in the text. This is a simplistic approach which is easy to implement and provides a rough idea of the nature of the source text. However, there are problems with word order which result in low quality translations. Syntactic transfer systems such as ELU (e.g. Estival et. al [EBRS90]) have been used to solve the word order problem, since the source sentence is parsed, its constituents rearranged and then translated. However, such systems require transfer rules for each of the languages under consideration. Other approaches exist, including Example-based Machine Translation (EBMT) which uses the concept of analogy to perform translation. The source sentence is decomposed into a sequence of fragments. Each fragment is then translated individually and then composed properly to form the target sentence. Statistical Machine Translation (SMT) (Brown-et.al. [BCDP$^+$88]) with which this paper is concerned, uses probability to find the most probable target translation given a source text.

The remainder of this paper is structured as follows. Section 2 and 3 give an overview of SMT and Phrase Alignment respectively. Section 4 proposes an algorithm for phrase identification, extraction and alignment. Section 5 discusses how the proposed method will be integrated in an existing MT system to test and evaluate the output of the phrase alignment module. We conclude in section 6 with an overview of some outstanding issues and suggestions for future work.

2 Statistical Machine Translation

Statistical Machine Translation (SMT) finds the most probable translation of a sentence on the basis of a model which is inferred from training data consisting of large quantities of translated text. It has a number of advantages over transfer-based and example-based MT. It is data-driven, language-independent and does not require any linguist or language experts. In addition, a new system can be prototyped quickly and at low cost provided that parallel training corpora are available for the language pair in question.

SMT rests upon a remarkably simple insight: given an occurrence of a sentence m in, say, Maltese, and any other sentence e in, say, English, there is a non-zero probability that given m, e expresses what the speaker of m had in mind when m was said. We write this probability $P(e|m)$, which intuitively represents the probability that e is *a translation* of m.

Now suppose that for the same m, I am offered two candidates: e_1 and e_2. How will I choose which is the better translation? Obviously I will work out $P(e1|m)$ and $P(e2|m)$ and then choose e_1 if $P(e1_m) > P(e_2|m)$, else I will choose e_2. This gives us the basis for a theory of translation. In order to find the translation of m, find $\hat{e} = argmaxP(e|m)$, i.e. the e which maximises $P(e|m)$. Bayes' theorem allows us to decompose the latter probability into $P(e)P(m|e)/P(m)$, so that $\hat{e} = argmaxP(e)P(m|e)/P(m)$, which simplifies to

$$\hat{e} = argmaxP(e)P(m|e)$$

because the denominator is independent of e.

In a classic paper Brown et al. [BCDPDP93] refer to this equation as the "Fundamental Equation of Machine Translation", since it summarises three computational issues that need to be addressed in the design of SMT systems. These are:

- Estimating the *language model* probability $P(e)$;
- Estimating the *translation model* probability $P(m|e)$;
- Designing a *search method* to identify the English string which maximises the product.

The last issue, which is usually referred to as "decoding", brings the translation process firmly back to the familiar territory of search optimisation, which we will not discuss the process further here, except to say that (1) it normally source-sentence driven, proceeding by translating successive segments of the source sentence, and (2) at any given stage, extension of a partial translation to the next segment involves a probabilistic calculation that eliminates the least probable of the possible extensions. Such calculations make use of the (already existing) translation model.

In an ideal world, a translation model $P(m|e)$ would take the form of a gigantic lookup table that associated a probability, i.e. a real number between 0 and 1, to every possible pairing of a Maltese string and an English one. The question is therefore, how we go about estimating those probabilities. Clearly, there is neither enough data nor computing power in the world to estimate the probabilities by counting the frequencies of every string pair individually.

The general solution is therefore (1) to divide the translation model of the whole sentence into smaller parts for which translation probabilities are more readily available and (2) to combine the translation probabilities of the parts. The simplest way of doing this, and the basis of word-based SMT systems, is to consider the translation probabilities between individual words in m and e. In such cases, for a sentence of length k:

$$P(m|e) = P(m_1|e_1)\times, \ldots \times P(m_i|e_i), \ldots \times P(m_k|e_k)$$

where m_i, e_i represent the ith word of m and e and $1 = i = k$.

Clearly, such a model will work well when there is a 1:1 correspondence between the words of source and target sentences. However, this is a simplistic assumption, and when it fails, this is the system may assign a high probability to a low quality sentence.

By considering the empty string as a word, systems based on word translation models can handle insertions and deletions. However, they cannot capture local word reordering. Even simple cases like noun/adjective ordering differences between source and target cannot be dealt with, and the net result is low translation quality.

3 Phrase Alignment

As SMT evolved, concerns about translation quality as well as better availability of data have led to development of SMT based on other kinds of statistical model.

Och and Ney [ON04]) present an "alignment template" approach to translation which allows for many-to-many relations between source and target words, whilst Huang et al. [HKJ06] adopt a syntax-directed approach based on the relationship between the nodes in a syntax-tree on the source side and a derived target-language string.

This paper is concerned with the problem of extracting phrase-translation models from bilingual data. Such models, once available, can then be used by a statistical translation system to actually carry out translations.

The basic problem is that, given a source and target sentence, the number of possible phrase alignments is much too large for the probability of each to be individually calculated. Therefore, we need some principled way to identify the most promising candidates. The essence of the principle is the following hypothesis: words that are translation equivalents are more likely to be embedded in phrases that are translation equivalents than in phrases which are not.

If this hypothesis is true, we can use words that are known *a priori* to be translation equivalents to generate related phrases in on both source and target sides, and hence, to generate phrase alignments.

4 Proposed Methodology

We already have in place a system for the paragraph, sentence and word alignment of Maltese and English bilingual corpora (Bajada [Baj04]). The method being proposed uses previously aligned sentences and words as the starting point for phrase identification and alignment.

The method has two key elements; phrase identification and phrase alignment. Essentially, *phrase identification* takes a word and delivers a set of phrases that involve that word; *phrase alignment* takes a phrase within a sentence and delivers the most probable translation equivalent of that phrase. The next two sections describe these two processes in more detail.

4.1 Phrase Identification

The proposed algorithm assumes the existence of a word translation model in the sense defined above, and starts off considering a source word and its target translation. At this stage each word in the source vocabulary is considered individually. Suppose we are considering a particular source word, w.

Definition 1. *An n-m-phrase is considered to be a contiguous sequence of words* $l_1, \ldots, l_n, w, r1, \ldots, r_m$ m,$n = 0$ *focused on the word* w.

The function get-phrase(w, s, n, m) extracts an $n-m$-phrase containing w from sentence s so that by setting particular values for n and m we can, for example, focus on left or right context, or a combination of both. If s has insufficient words, the result is padded out with null strings working outward from w, thus maintaining the relative position of w within the phrase whose total length is $m + n + 1$.

An $n - m$-phrase p of length k focused on word w and given n and m is deemed to be interesting for sentence s if

- $w \in s$
- $p = \mathit{get\_phrase}(w, s, n, m)$ for $n + m + 1 = k$
- The probability of p exceeds a certain threshold which is defined with respect to all other $n - m$-phrases in the source corpus.

For a given corpus, the best values of n and m are established empirically, and ultimately judged by the quality of the phrase pairs extracted as described in section 5. Our initial hypothesis is that $n = k - 1$ and $m = 0$ i.e the given word w is at the extreme right of the phrase whose total length is k.

There are many different ways in which the probability of p might be established with reference to a training corpus e.g. with respect to an n–gram language model of the source corpus or possibly a subcorpus of sentences containing at least one occurrence of w. We are currently experimenting with some simple models.

4.2 Phrase Alignment

The process of phrase identification presented in the previous section is central to the phrase alignment process. However, a number of steps are required to put the whole process of phrase extraction and alignment together.

As mentioned previously we have available a set of word alignments and their respective translation probabilities. We need to identify those phrases which are of interest to us and which, if extracted and aligned, will be of benefit to the translation process. We assume that those source words having a combination of a high word count in the training text and a high translation probability are those words of interest to the alignment process. The word translation equivalents are sorted and filtered according to these criteria. The following steps are then carried out on the sorted list to complete the alignment process.

For each word pair (m, e) in the list

- $A = \{(ss, st) \mid m \in ss \;\&\; e \in st\}$
- $B = \{(ps, pt, P) \mid (ss, st) \in A \;\&\; \begin{array}{l} ps = \mathit{phrase\_identify}(ss, m, k, 0) \;\& \\ pt = \mathit{phrase\_identify}(st, e, k, 0) \;\& \\ P = P(ss|st)\} \end{array}$
- Return B

The function *phrase_identify* simply invokes the phrase identification process outlined in the previous section, yielding a phrase whose probability is above threshold.

The final step in the alignment process is the calculation of the translation probabilities for the resulting phrase pairs. There are various ways of doing this, the simplest being to take the product of the translation probabilities of the words which are part of the phrases. This works because the words of the two phrases starting with the m and e are mostly aligned according to our principle.

Alternatively bigram probabilities might be used: as with any statistical model, other factors can be introduced, such as the relative frequency of the collected phrases, as discussed in Koehn's lectures.

Outputting a phrase table which contains the translation probabilities of the extracted phrases is an important step which allows the results to be used as the Translation Model component of a SMT system.

The above steps extract a phrase table which is suitable for evaluation. The system needs to be as flexible as possible to cater for different language pairs. The phrase extraction algorithm has to be tuned to determine the optimal extraction method — words before, words after or a combination of words before and after the initial word.

In addition a cutoff point for phrase extraction needs to be determined; that is, when to stop processing the phrase. This may be done by requiring the phrase to have a probability greater than a threshold value of by setting the maximum length of the phrase.

The aim of the current project is to develop a highly configurable system, the parameters of which will be used to determine the extraction of phrases from a training corpus.

Developing the system this way will allow us to experiment with and modify these parameters until an optimum set is found. This will be determined by applying the extracted phrase table to and SMT system and using evaluation metrics to measure the quality of the output translations.

5 Discussion

5.1 How the System is Used

The system described above yields a series of phrase translation models, i.e. tables whose entries are triples of the form {source phrase, target phrase, probability). By varying the parameters of get-phrase, we obtain different translation models, and our aim is to investigate the settings which produce the best translations.

The proof of the pudding is in the eating — and in our case the eating corresponds to how well the model actually translates a set of test sentences. To put this into practice we require a translation system into which we can plug the translation table, and a way to assess the quality of individual translations. Fortunately, both of these are available off the shelf.

Moses (Koehn et. al [KHB+07]) is a state-of-the-art factored phrase-based beam-search decoder for SMT which is freely available. It is a traditional SMT system in that it relies on a language model and a translation model to perform translation. In addition, Moses offers a number of advanced features and additional tools for machine translation, such as scoring and analysis tools.

5.2 Translation Model

The Translation Model used by Moses is a phrase translation table, with entries such as those shown in Figure 1.

```
das ist ||| this is ||| 0.3
```

Fig. 1:

The entry indicates that the probability of translating the German phrase *das ist* into the English phrase *this is* is 0.3. The integration and evaluation of our results requires that they be written to a phrase translation table formatted as shown in Figure 1.

5.3 Evaluation

The Moses toolkit also provides a number of support tools enabling further analysis of output translations. Among these is a simple BLEU scoring tool. BLEU (Papineni et al [PSWZ88]) is an automatic evaluation metric used for MT proposed by the IBM MT research group. The central idea of automatic evaluation is to use a weighted average of variable length phrase matches against the reference translations.

BLEU averages the precision for unigram, bigram and up to 4-grams and applies a length penalty if the generated sentence is shorter than the best matching (in length) reference translation.

Application of the BLEU metric to the translations output by Moses using our phrase translation table will indicate the quality of the translations. This will in turn provide indication of the validity of the aligned phrases.

6 Conclusion

There are a number of unknown variables which factor in the identification and extraction of phrases. For two different languages there is no guarantee that equivalent phrases will have the same attributes, such as length and word order. Developing an algorithm to automate such a process requires that the algorithm be flexible enough to cater for such variation.

The automatic extraction of word-based translation models from corpora is performed by assuming that every pair of source/target words chosen from two aligned sentences have an equal probability of being equivalent. More refined models allow for the positions of words within the aligned sentences contribute to the probability of their alignment. Such principles can be extrapolated to phrase alignment; that is, we assume that only phrases from equivalent sentences are a correct match. In addition, having aligned the texts at word level, we have a dictionary of words and their possible alignments which may be extended to phrases.

The default assumption is that phrase extraction from Maltese and English texts may be carried out in the same way because of the similar structure between the two languages. A different language pair might require a different methodology. We are also able to adopt this approach due to the level of confidence in the sentence and word translation equivalents obtained by the current system. The developed system needs to be as flexible as possible to cater for different language pairs. The aim of the current project is to develop a highly configurable system which depends on a number of parameters to determine the identification and alignment of phrases from a training corpus. Developing such a system will allow us to experiment with and modify these parameters until an optimum set is found.

Manual evaluation of results obtained using an initial implementation of the proposed algorithm on a set of bilingual texts gave positive results. The results indicated that parameterisation of the algorithm would improve the results.

Having extracted a set of equivalent phrases offers a number of processing options. It would be interesting to attempt template extraction, whereby the resulting phrases are analysed for similarities and used to build templates to offer more generic translation options.

References

[Baj04] J-A Bajada. Investigation of translation equivalences from parallel texts. Technical Report 2004–1, Dept CSAI, University of Malta, Msida MSD2080, Malta, 2004.

[BCDP+88] P.F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Frederick Jelinek, Robert L. Mercer, and Paul S. Roossin. A statistical approach to machine translation. In *Proceedings of the 12th International Conference on Computational Linguistics (COLING-88)*, pages 71–76, Budapest, August 1988.

[BCDPDP93] P.F. Brown, John Cocke, Stephen A. Della Pietra, and Robert L. Della Pietra, Mercer. The mathematics of statistical machine translation. *Computational Linguistics*, 19.3:263–311, 1993.

[EBRS90] D. Estival, A. Ballim, G. Russell, and Warwick S. A syntax and semantics for feature-structure transfer. In *Proceedings of the 3rd International Conference on Theoretical and Methodological Issues in Machine Translation of Natural Language*, pages 131–143, University of Texas, 1990.

[HKJ06] L. Huang, K. Knight, and A. Joshi. Statistical syntax-directed translation with extended domain of locality. In *Proceedings of the AMTA*, pages 131–143, Cambridge, Ma., 2006.

[KHB+07] P. Koehn, Hieu Hoang, Alexandra Birch, Chris Callison Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In *Demonstration Session, ACL07*, pages 131–143, Prague, 2007.

[ON04] Franz Josef Och and Hermann Ney. The mathematics of statistical machine translation. *Computational Linguistics*, 30:417–449, 2004.

[PSWZ88] K. Papineni, R. Salim, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the ACL*, pages 311–319, Philadelphia, August 1988.