A phrase-based hidden Markov model approach to machine translation

Jesús Andrés-Ferrer
Universidad Politécnica de Valencia
Dept. Sist. Informáticos y Computación
jandres@dsic.upv.es

Alfons Juan-Císcar
Universidad Politécnica de Valencia
Dept. Sist. Informáticos y Computación
ajuan@dsic.upv.es

Abstract

Current statistical machine translation systems are based on phrases heuristically extracted. In this work, a new approach for phrase-based statistical machine translation is proposed which can properly described as a hidden Markov model. The proposed model, its associated forward and backward recurrences, and its EM-based maximum likelihood estimation is detailed. Empirical results are reported on a spanish-english translation task.

1 Introduction

The machine translation problem is stated as the problem of translating a *source (input)* sentence, $\mathbf{x}$, into a *target (output)* sentence, $\mathbf{y}$. In accordance with the statistical approach to machine translation, the optimal translation $\hat{\mathbf{y}}$ of a source sentence $\mathbf{x}$ is given by (Brown et al., 1993):

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}\in\mathcal{Y}^\star} p(\mathbf{y}\,|\,\mathbf{x})p(\mathbf{y}) \qquad (1)$$

where $p(\mathbf{y}\,|\,\mathbf{x})$ is approximated by an *inverse translation model* and $p(\mathbf{y})$ is modelled with a *language model*; which is usually instanced by a *n-gram language model* (Stolcke, 1997).

The first proposed models, the so-called *IBM translation models* (Brown et al., 1993), tackled the problem with word-level dictionaries plus alignments between words. However, current systems model the inverse conditional probability in Equation (1) using *phrase dictionaries*. This phrase-based methodology stores specific sequences of target words (*target phrase*) into which a sequence of source words (*source phrase*) is translated. The key concept of this approach is the procedure through which these phrase pairs are inferred.

A popular, phrase-based technique consists in using the IBM alignment models (Brown et al., 1993) to obtain a symmetrised alignment matrix from which *coherent* phrases are extracted (Och and Ney, 2004). Then, a simple count normalisation is carried out in order to obtain a conditional phrase dictionary.

Alternatively, some approaches have been described in the last few years in which phrase dictionaries are statistically inferred. In particular, a joint probability model for phrase-based estimation is proposed in Marcu and Wong (2002). In this work, all possible segmentations are extracted using the EM algorithm (Dempster et al., 1977), without any matrix alignment constrain, in contrast to the approach followed in Och and Ney (2004). Based on this work, A. Birch et al. (2006), constrained the EM to only consider phrases which agree with the alignment matrix, thus reducing the size of the phrase dictionaries (or tables).

A possible drawback of the above phrase-based models is that they are not conditional, but joint models that need to be renormalised in order to make them conditional. In this work, we propose a direct, conditional phrase-based approach for monotone translation. Monotonicity allows us to derive a relatively simple statistical model which can be properly described as a *phrase-based hidden Markov model*. In what follows, we first introduce our model in Section 2, and then their associated forward and backward recurrences in Section 3. EM-based maximum likelihood estimation of the model parameters is described in Section 4. Empirical results are reported in Section 5 and then some concluding remarks are given.

2 Phrase-based hidden Markov model

Let $\mathbf{x}$ and $\mathbf{y}$ be a pair of source and target sentences of known length, J and I. In order to define our phrase-based hidden Markov model for $p(\mathbf{x} \,|\, \mathbf{y})$, it is first convenient to introduce our definition of monotone segmentation, both for the monolingual and bilingual cases.

A monotone, monolingual segmentation of $\mathbf{x}$ into a given number of segments, T, is any sequence of indexes $\mathbf{j} = (j_0, j_1, \ldots, j_T)$ such that $1 = j_0 < j_1 < \cdots < j_T = J$. Similarly, a monotone, segmentation of $\mathbf{y}$ into T segments is any sequence of indexes $\mathbf{i} = (i_0, i_1, \ldots, i_T)$ such that $1 = i_0 < i_1 < \cdots < i_T = I$. Given two monotone, monolingual segmentations of $\mathbf{x}$ and $\mathbf{y}$ into T segments, $\mathbf{j}$ and $\mathbf{i}$, their associated *bilingual* segmentation of $\mathbf{x}$ and $\mathbf{y}$ is defined as $\mathbf{s} = s_1 s_2 \cdots s_T$ with $s_t = (j_{t-1}+1, j_t, i_{t-1}+1, i_t)$, $t = 1, \ldots, T$. Reciprocally, given a monotone, *bilingual* segmentation of $\mathbf{x}$ and $\mathbf{y}$, we can easily extract their associated monolingual counterparts.

Figure 1 shows an example in which all possible bilingual segmentations for $J = 4$ and $I = 5$ are represented as paths in a directed, multi-stage graph. The initial stage of the graph has a single, artificial node labelled as "init", which is only included to point to the initial segments of all the possible segmentations. There are 12 of such initial segments, vertically aligned on the first stage. Similarly, there are 15, 3 and 13 segments aligned on the second, third and final stages, respectively. The total number of segments is then 43. There is a unique segmentation of unit length, $\mathbf{s} = s_1 = (1415)$, which is represented by the rightmost path, but there are 12, 18 and 4 segmentations of length 2, 3 and 4, respectively; comprising 35 segmentations in total. As empty segments are not allowed, segmentation lengths range from one to the length of the shortest sentence. Note that segments on the first stage can only appear in the first position of a segmentation. Also, segments on the second and final stages can only appear on analogous positions in a segmentation. However, those three on the third stage (i.e. (3334), (3333) and (3344)) may appear in the second or third positions, although they cannot end any segmentation. For instance, (3334) appears in the second position of $((1212), (3334), (4455))$ and also in the third position of $((1111), (2222), (3334), (4455))$.

Note that we are using the terms segment and segmentation only for positions (indexes) in the input and output sentences. We reserve the term *phrase* for actual portions of the given sentences. Thus, for instance, the bilingual segmentation $((1212), (3334), (4455))$ of $\mathbf{x} = x_1 x_2 x_3 x_4$ and $\mathbf{y} = y_1 y_2 y_3 y_4 y_5$ results in the bilingual phrases $(x_1 x_2, y_1 y_2)$, $(x_3, y_3 y_4)$ and (x_4, y_5). In what follows, we will write $\mathbf{x}(s_t)$ to denote the portion of $\mathbf{x}$ delimited by (the input part of) segment s_t; more generally, $\mathbf{x}(s_{t'}^{t})$ will denote the concatenation $\mathbf{x}(s_{t'})\mathbf{x}(s_{t'+1}) \cdots \mathbf{x}(s_t)$. Analogous notation will be used for $\mathbf{y}$: $\mathbf{y}(s_t)$ and $\mathbf{y}(s_{t'}^{t})$.

Now, we can define our model for $p(\mathbf{x} \,|\, \mathbf{y})$ as a full exploration of all bilingual segmentations of $\mathbf{x}$ and $\mathbf{y}$,

$$p(\mathbf{x} \,|\, \mathbf{y}) = \sum_{T=1}^{\min(J,I)} \sum_{\mathbf{s}} p(\mathbf{x}, \mathbf{s}, T \,|\, \mathbf{y}) \qquad (2)$$

where the second sum can be defined over all possible bilingual segmentations, or only over those of length T. The joint probability of occurrence of $\mathbf{s}$ and T is null if the actual length of $\mathbf{s}$ is not T.

To compute $p(\mathbf{x}, \mathbf{s}, T \,|\, \mathbf{y})$ in (2), we use the following decomposition:

$$p(\mathbf{x}, \mathbf{s}, T \,|\, \mathbf{y}) = p(\mathbf{s} \,|\, \mathbf{y}) p(T \,|\, \mathbf{y}, \mathbf{s}) p(\mathbf{x} \,|\, \mathbf{y}, \mathbf{s}, T) \quad (3)$$

where $p(\mathbf{s} \,|\, \mathbf{y})$ is modelled as a first-order Markovian process,

$$p(\mathbf{s} \,|\, \mathbf{y}) := \prod_{t \geq 1} p(s_t \,|\, s_{t-1}) \qquad (4)$$

with $s_0 :=$"init", and $p(\mathbf{x} \,|\, \mathbf{y}, \mathbf{s}, T)$ is modelled as composed of independent bilingual phrases,

$$p(\mathbf{x} \,|\, \mathbf{y}, \mathbf{s}, T) := \prod_{t \geq 1} p(\mathbf{x}(s_t) \,|\, \mathbf{y}(s_t), s_t) \qquad (5)$$

Clearly, the above modelling assumptions lead to a phrase-based HMM-like model. Its set of states is that of all possible bilingual segments, while its set of transitions includes all pairs $\langle q', q \rangle$ in which the state (segment) q is a successor of q', $q \in \mathrm{Succ}(q')$; e.g. $\langle (1212), (2234) \rangle$.

In this work, we will further assume that both initial and transition state probabilities are uniformly distributed; hence, for each q and q', including "init" for q',

$$p(q \,|\, q') := \begin{cases} \frac{1}{|\mathrm{Succ}(q')|} & \text{if } q \in \mathrm{Succ}(q') \\ 0 & \text{otherwise} \end{cases} \qquad (6)$$

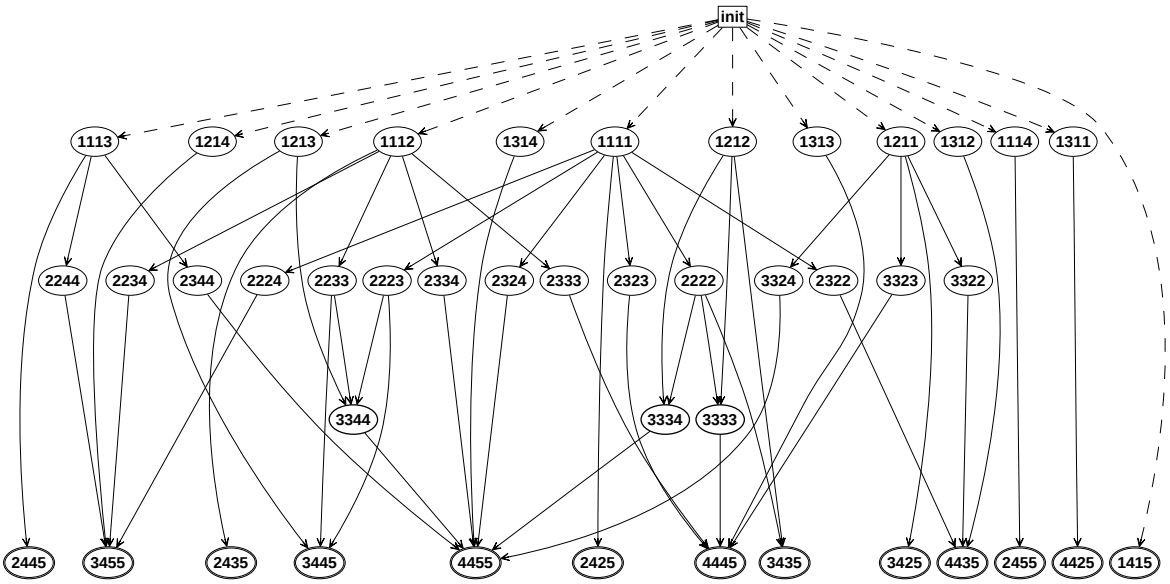

Figure 1: Directed, multi-stage graph representing all possible bilingual segmentations for an input sentence of length 4 and an output sentence of length 5. Each node defines a different segment; the first two digits of the node label are the segment limits in the input sentence, while the other two digits correspond to the output sentence.

Also, T is assumed to be uniformly distributed,

$$p(T \,|\, \mathbf{y}, \mathbf{s}) := \begin{cases} \frac{1}{\min(I,J)} & \text{if } |\mathbf{s}| = T \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

and the phrase translation probabilities are assumed to be stored in a single, state-independent table $\mathrm{p}(\cdot|\cdot)$:

$$p(\mathbf{x}(s_t) \,|\, \mathbf{y}(s_t), s_t) := \mathrm{p}(\mathbf{x}(s_t) \,|\, \mathbf{y}(s_t)) \quad (8)$$

Using the above assumptions, our model (2) can be rewritten as follows:

$$p(\mathbf{x} \,|\, \mathbf{y}) = \frac{1}{Z} \sum_{\substack{\mathbf{s} \\ |s| \leq Z}} \prod_{t=1}^{|\mathbf{s}|} \frac{\mathrm{p}(\mathbf{x}(s_t) \,|\, \mathbf{y}(s_t))}{|\mathrm{Succ}(s_{t-1})|} \quad (9)$$

with $Z = \min(J, I)$. The vector of parameters governing this model only includes a table of phrase translation probabilities,

$$\Theta = \{\mathrm{p}(\mathbf{u} \,|\, \mathbf{v}) : (\mathbf{u}, \mathbf{v}) \text{ bilingual phrase}\} \quad (10)$$

3 Forward and backward probabilities

As usual with HMMs, we will discuss here the so-called *forward* and *backward* probabilities for efficient computation of the model itself, as given in (9), and its EM-based parameter re-estimation (discussed in Section 4). To fix ideas, consider $\mathbf{x}$ and $\mathbf{y}$ to be two arbitrary sentences for which we have to compute (9).

Given a segmentation length and position, T and t, and a state q, the forward probability is defined as the joint probability

$$\alpha_{tq}^T := p(\mathbf{x}(s_1^t), s_t = q \,|\, \mathbf{y}, T) \quad (11)$$

where s_1^t is any partial segmentation, from positions 1 to t, such that $s_t = q$. This probability can be efficiently computed by dynamic programming, using the so-called *forward recurrence*,

$$\alpha_{tq}^T = \sum_{q' \,:\, q \in \mathrm{Succ}(q')} \alpha_{t-1q'}^T \frac{\mathrm{p}(\mathbf{x}(q) \,|\, \mathbf{y}(q))}{|\mathrm{Succ}(q')|} \quad (12)$$

with $\alpha_{tq}^T = 1$ for $t = 0$ and $q =$"init"; 0 otherwise.

The backward probability also depends on a given segmentation length and position, T and t, and a state q. It is defined as the conditional probability

$$\beta_{tq}^T := p(\mathbf{x}(s_{t+1}^T) \,|\, \mathbf{y}, T, s_t = q) \quad (13)$$

where s_{t+1}^T is any partial segmentation, from positions $t+1$ to T, that might follow segment q

in position t. As before, it can be efficiently computed by dynamic programming, using a "reverse" version of the forward recurrence called *backward recurrence,*

$$\beta_{tq}^{T} = \sum_{q' \in \mathrm{Succ}(q)} \beta_{t+1q'}^{T} \frac{\mathrm{p}(\mathbf{x}(q') \,|\, \mathbf{y}(q'))}{|\mathrm{Succ}(q)|} \quad (14)$$

with $\beta_{Tq}^{T} = 1$ for any *terminal* $q = (\cdot, I, \cdot, J)$ and any t; 0 otherwise.

Equation (9) can be computed using (12) as

$$p(\mathbf{x} \,|\, \mathbf{y}) = \frac{1}{Z} \sum_{T} \sum_{q=(\cdot,I,\cdot,J)} \alpha_{Tq}^{T} \quad (15)$$

or using (14) as

$$p(\mathbf{x} \,|\, \mathbf{y}) = \frac{1}{Z} \sum_{T} \beta_{0\text{“init”}}^{T} \quad (16)$$

4 Maximum likelihood estimation

As discussed in Section 2, the unknown vector of parameters of our phrase-based HMM model only includes a table of phrase translation probabilities (10). We will describe here its EM-based maximum likelihood estimation with respect to a collection of training translation pairs $(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_N, \mathbf{y}_N)$.

The log-likelihood function of $\mathbf{\Theta}$ is:

$$\begin{aligned} L(\mathbf{\Theta}) &= \sum_{n} \log p(\mathbf{x}_n \,|\, \mathbf{y}_n) \\ &= \sum_{n} \log \frac{1}{Z_n} \sum_{\substack{\mathbf{s} \\ |s| \leq Z_n}} \prod_{t=1}^{|\mathbf{s}|} \frac{\mathrm{p}(\mathbf{x}_n(s_t) \,|\, \mathbf{y}_n(s_t))}{|\mathrm{Succ}(s_{t-1})|} \end{aligned} \quad (17)$$

The EM algorithm maximises Eq. (17) iteratively, through the application of two basic steps in each iteration: the E(xpectation) step and the M(aximisation) step. The E step computes the expected value of the "hidden" variables given an estimation of the parameters. The M step maximises (17) using the expected values computed in the E-step. Given an initial value of the parameters, $\mathbf{\Theta}^{(0)}$, these two steps are repeated until convergence to a local maximum of the likelihood function.

In our case, the E step requires the computation, for each pair $(\mathbf{x}_n, \mathbf{y}_n)$, of sample versions of (12) and (14), as well as the joint probability

$$\xi_{ntq'q}^{T} := p(\mathbf{x}_n, s_{t-1} = q', s_t = q \,|\, \mathbf{y}_n, T) \quad (18)$$

which can be efficiently computed as

$$\xi_{ntq'q}^{T} = \frac{\alpha_{nt-1q'}^{T} \; \mathrm{p}(\mathbf{x}(q) \,|\, \mathbf{y}(q)) \, \beta_{ntq}^{T}}{p(\mathbf{x}_n \,|\, \mathbf{y}_n) \, |\mathrm{Succ}(q')|} \quad (19)$$

On the other hand, the M step re-estimates the table of phrase translation probabilities,

$$\mathrm{p}(\mathbf{u} \,|\, \mathbf{v}) = \frac{N(\mathbf{u}, \mathbf{v})}{\sum_{\mathbf{u}'} N(\mathbf{u}', \mathbf{v})} \quad (20)$$

where $N(\mathbf{u}, \mathbf{v})$ is the expected number of occurrences of the the pair $(\mathbf{u}, \mathbf{v})$; i.e.

$$N(\mathbf{u}, \mathbf{v}) = \sum_{n \, q' q \, T} \frac{1}{T} \sum_{t} \xi_{ntq'q}^{T} \, \delta_{nq}(\mathbf{u}, \mathbf{v}) \quad (21)$$

with $\delta_{nq}(\mathbf{u}, \mathbf{v})$ defined as 1 if $\mathbf{u} = \mathbf{x}_n(q)$ and $\mathbf{v} = \mathbf{y}_n(q)$; 0 otherwise.

5 Experiments

Our phrase-based hidden Markov model was assessed on the EUTRANS-I dataset (Amengual et al., 2000). This dataset comprises 12 000 bilingual sentence pairs from a limited-domain spanish-english machine translation application for human-to-human communication situations in the front-desk of a hotel. It was semi-automatically built from a small seed corpus of sentence pairs collected from traveller-oriented booklets. Some basic statistics are shown in Table 1.

	test set		training set	
	spa	eng	spa	eng
sentences	2K		10K	
avg. length	12.7	12.6	12.9	13.0
vocabulary	611	468	686	513
singletons	63	49	8	10
running words	35K	36K	97K	99K
perplexity	3.7	2.97	3.63	2.96

Table 1: Statistics of the EUTRANS-I corpus.

Two basic experiments were carried out. In the first experiment, we did not use our phrase-based HMM since the experiment was designed only to obtain a baseline result for comparison purposes. Instead, we used GIZA++ to obtain a table of phrase translation probabilities, by simple count normalisation of phrases in the training set that were coherent with the symmetric alignment matrix (Och and Ney, 2004). Then, we used

Pharaoh (Koehn, 2004) to search for the most probable translation of each source sentence in the test set, and the quality of the translated set was evaluated using the *word error rate* (WER) and *bilingual evaluation understudy* (BLEU) measures (Papineni et al., 2001). We obtained a WER of 7.7% and BLEU of 89.1%. These are relatively good results since, in general, low values of WER and high values of BLEU are a clear indication of high quality translations.

In the second experiment, we used our phrase-based hidden Markov model to better train the phrase translation table. We proceeded as in the first experiment although, now, the phrase table obtained before was used to initialise the EM algorithm proposed in Section 4 for parameter training in accordance with criterion (17). In this case, we obtained a WER of 7.8% and BLEU of 88.5%.

Obviously, the result obtained with our model was not better than that obtained with the baseline approach. In analysing the phrase table provided by our model, we found that the EM algorithm prefers long to short phrases; that is, given a target phrase, long source phrases are favoured with higher probabilities. To empirically check this hypothesis, we repeated the two basic experiments described above by first discarding training phrases longer that a given maximum threshold. For the most restrictive thresholds, however, phrases longer than the threshold were not discarded so as to ensure full coverage of the training data. The results are shown in Figure 2 in terms of BLEU.

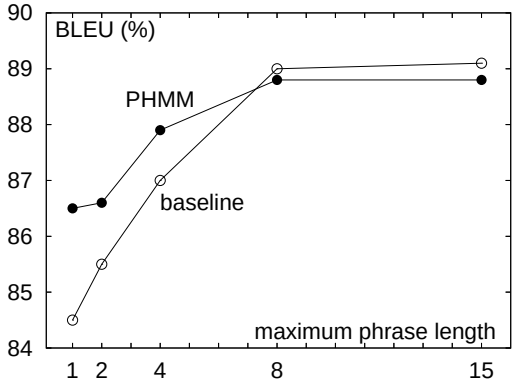

Figure 2: BLEU (%) as a function of the maximum phrase length threshold, for the baseline approach and our phrase-based HMM (PHMM).

The results in Figure 2 confirm our hypothesis on the bias to long phrases in our model. A possible remedy to this problem is to refine our phrase-based HMM with inclusion of length models to penalise long phrases.

6 Conclusions

A phrase-based hidden Markov model has been proposed for statistical machine translation. We have described the forward and backward recurrences for efficient computation of the model and its EM-based parameter re-estimation, which has been also described. Empirically results have been reported comparing the proposed model with a baseline system. It has been found that our model is biased to long phrases and does not provide better results than the baseline system unless phrases are restricted to a maximum length. For future work, we plan to include a length model for phrases and to carry out more experiments with larger datasets.

We also intend to take into account example-based hybrid data-driven systems (Groves and Way, 2006).

7 Acknowledgements

This work has been partially supported by the *Conselleria d'Empresa Universitat i Ciència - Generalitat Valenciana* (Valencia, Spain) under the grant CTBPRA/2005/004 and by the CICYT *Centro de Investigación Científica y Tecnológica* under the Spanish project iDoc (TIC2003-08681-C02) and by *Consellería d'Empresa, Universitat i Ciència - Generalitat Valenciana* under contract GV06/252..

References

Miles Osborne Alexandra Birch, Chris Callison-Burch and Philipp Koehn. 2006. Constraining the phrase-based, joint probability statistical translation model. In *Proceedings of the NAACL 2006 Workshop on Statistical Machine TranslationConference*.

Juan C. Amengual, José M. Benedí, Asunción Castano, Antonio Castellanos, Víctor M. Jiménez, David Llorens, Andrés Marzal, Moisés Pastor, Federico Prat, Enrique Vidal, and Juan M. Vilar. 2000. The EuTrans-I speech translation system. *Machine Translation*, 15:75–103.

Peter F. Brown, Stephen Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–312.

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Statist. Soc. Ser. B*, 39(1):1–22.

D. Groves and A. Way. 2006. Hybrid data-driven models of mt. *Machine Translation, Special Issue on EBMT*.

Philipp Koehn. 2004. Pharaoh: a beam search decoder for phrase-based statistical machine translation models. In *Proceedings of the AMTA 2004*.

Daniel Marcu and Qilliam Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)*, Philadelphia, PA, July.

F.J. Och and H. Ney. 2004. The Alignment Template Approach to Statistical Machine Translation . *Computational Linguistics*, 30(4):417–449, December.

Kishore A. Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2001. Bleu: a method for automatic evaluation of machine translation. Technical Report RC22176 (W0109-022), IBM Research Division, Thomas J. Watson Research Center, Yorktown Heights, NY, September.

Andreas Stolcke. 1997. SRILM - an extensible language modelling toolkit. Technical report, Speech Technology and Research Laboratory, SRI International, Menlo Park, CA, USA.