Constraining the Phrase-Based, Joint Probability Statistical Translation Model

Alexandra Birch   Chris Callison-Burch   Miles Osborne
School of Informatics
University of Edinburgh
2 Buccleuch Place
Edinburgh, EH8 9LW, UK
a.c.birch-mayne@sms.ed.ac.uk

Abstract

The Joint Probability Model proposed by Marcu and Wong (2002) provides a probabilistic framework for modeling phrase-based statistical machine translation (SMT). The model's usefulness is, however, limited by the computational complexity of estimating parameters at the phrase level. We present a method of constraining the search space of the Joint Probability Model based on statistically and linguistically motivated word alignments. This method reduces the complexity and size of the Joint Model and allows it to display performance superior to the standard phrase-based models for small amounts of training material.

1 Introduction

Machine translation is a hard problem because of the highly complex, irregular and diverse nature of natural languages. It is impossible to accurately model all the linguistic rules that shape the translation process, and therefore a principled approach uses statistical methods to make optimal decisions given incomplete data.

The original IBM Models (Brown et al., 1993) learned only word-to-word alignment probabilities which made it computationally feasible to estimate model parameters from large amounts of training data. Phrase-based SMT models, such as the Alignment Template Model (Och, 2003b), improve on word-based models because phrases provide local context which leads to better lexical choice and more reliable local reordering. However, most phrase-based models extract their phrase pairs from previously word-aligned corpora using ad-hoc heuristics. These models perform no search for optimal phrasal alignments. Even though this is an efficient strategy, it is a departure from the rigourous statistical framework of the IBM Models.

Marcu and Wong (2002) proposed a Joint Probability Model which directly estimates phrase translation probabilities from the corpus. This model neither relies on potentially sub-optimal word alignments nor on heuristics for phrase extraction. Instead, it searches the phrasal alignment space, simultaneously learning translation lexicons for both words and phrases. The Joint Model has been shown to outperform standard models on restricted data sets such as the small data track for Chinese-English in the 2004 NIST MT Evaluation (Przybocki, 2004).

However, considering all possible phrases and all their possible alignments vastly increases the computational complexity of the Joint Model when compared to its word-based counterpart. This results in prohibitively slow training and heavy use of memory resources. The large size of the model means that only a very small proportion of the alignment space can be searched, and this reduces the chances of finding optimum parameters. Furthermore, the complexity of the Joint Model makes it impossible to scale up to the larger training corpora available today, preventing the model from being more widely adopted.

We propose a method of constraining the search

Proceedings of the 7th Conference of the Association for Machine Translation in the Americas, pages 10-18,
Cambridge, August 2006.

space of the Joint Model to areas where most of the unpromising phrasal alignments are eliminated and yet as many potentially useful alignments as possible are still explored. The Joint Model is constrained to phrasal alignments which do not contradict a set high confidence word alignments for each sentence. These high confidence alignments can incorporate information from both statistical and linguistic sources. We show that by using the points of high confidence from the intersection of the bi-directional Viterbi alignments to reduce complexity, translation quality also improves. We also show that the addition of linguistic information from a machine readable dictionary and aligning identical words further improves the model.

In addition to showing how the translation quality can be improved through linguistic constraints, we show how to more quickly estimate the parameters. We describe a modification to the Expectation Maximisation (EM) algorithm which greatly increases the speed of the training without compromising the quality of the resulting translations.

2 Models

2.1 Standard Phrase-based Model

Most phrase-based models (Och, 2003b; Koehn et al., 2003; Vogel et al., 2003) rely on a pre-existing set of word-based alignments from which they induce their parameters. In this project we use the model described by Koehn et al. (2003) which extracts its phrase alignments from a corpus that has been word aligned. From now on we refer to this phrase-based model as the *Standard Model*.

The Standard Model decomposes the foreign input sentence F into a sequence of I phrases $\overline{f}_1, \ldots, \overline{f}_I$. Each foreign phrase $\overline{f_i}$ is translated to an English phrase $\overline{e_i}$ using the probability distribution $\theta(\overline{f}_i|\overline{e}_i)$. English phrases may be reordered using a relative distortion probability $d(\cdot)$. The model is defined as follows:

$$p(F|E) = \prod_{i=1}^{I} \theta(\overline{f}_i|\overline{e}_i) d(\cdot) \tag{1}$$

As alignments between phrases are constructed from word alignments, there is no summing over possible alignments. This model performs no search for optimal phrase pairs. Instead, it extracts phrase pairs $(\overline{f}_i, \overline{e}_i)$ in the following manner. First, it uses the IBM Models to learn the Viterbi alignments for English to Foreign and Foreign to English. It then uses a heuristic to reconcile the two alignments, starting from the points of high confidence in the intersection of the two Viterbi alignments and growing towards the points in the union. Points from the union are selected if they are adjacent to points from the intersection and their words are previously unaligned. Och and Ney (2004) discusses and compares variations on this strategy.

Phrases are then extracted by selecting phrase pairs which are 'consistent' with the symmetrised alignment. Here 'consistent' means that all words within the source language phrase are only aligned to the words of the target language phrase and vice versa. Finally the phrase translation probability distribution is estimated using the relative frequencies of the extracted phrase pairs.

This approach to phrase extraction means that phrasal alignments are locked into the symmetrised alignment. This is problematic because the symmetrisation process will grow an alignment based on arbitrary decisions about adjacent words, and, because word alignments inadequately represent the real dependencies between translations. Also, by heuristically creating phrasal alignments from the Viterbi word-level alignments, we throw away the probabilities that were estimated when learning word alignment parameters and we can introduce errors. In contrast, the Joint Model can search areas of the alignment space in order to learn a distribution of possible phrasal alignments that better handles the uncertainty inherent in the translation process.

2.2 Joint Probability Model

The Joint Probability Model (Marcu and Wong, 2002), does not rely on a pre-existing set of word-level alignments. Like the IBM Models, it uses Expectation Maximisation to align and estimate the probabilities for sub-sentential units in a parallel corpus. Unlike the IBM Models, it does not constrain the alignments to being single words.

The basic model is defined as follows. Phrases are created from words and commonly occurring sequences of words. *Concepts*, c_j, are defined as a pair of aligned phrases $< \overline{e}_i, \overline{f}_i >$. A set of concepts which completely covers the sentence pair is denoted by C. Phrases are restricted to being sequences of words which occur above a certain frequency in the corpus. We use a threshold of 5 occurrences. Commonly occurring phrases are more likely to lead to the creation of useful phrase pairs, because they are more likely to occur in the test data. Without restricting ourselves to frequent phrases, the search space would be much larger.

The probability of a sentence and its translation is the sum of all possible alignments, $\mathcal{C}$ each of which is defined as the product of the probability of all individual concepts:

$$p(F, E) = \sum_{C \in \mathcal{C}} \prod_{<\overline{e}_i, \overline{f}_i> \in C} p(<\overline{e}_i, \overline{f}_i>) \quad (2)$$

The model is trained by initialising the translation table and then performing EM as described below.

2.2.1 Initialising Translation Table

Before starting EM all phrasal alignments are assumed to be equally probable. Under these circumstances, the probability of a concept c_j in sentences (E, F) is equal to the number of phrasal alignments which contain this concept divided by the total number of phrasal alignments that can be built between the two sentences. This probability can be approximated by using the lengths of the two phrases and the lengths of the two sentences with Stirling numbers of the second kind as described by Marcu and Wong (2002). We are thus able to initialise all possible alignments.

The size of the translation table is largely determined by the initialisation phase, and so it greatly impacts on the scalability of the model.

2.2.2 Expectation Maximisation

After initialising the translation parameters, alignments will have different probabilities. It is no longer possible to collect fractional counts over all possible alignments in polynomial time. EM is therefore performed approximately to improve parameters and increase the probability of the corpus.

An iteration of EM starts by creating an initial phrasal alignment of high probability. This is done by selecting the highest probability concepts that cover the sentence pair. Then the model hill-climbs towards the optimal Viterbi alignment by using a set of modifying operations. These operations break and merge concepts, swap words between concepts and move words across concepts. The model calculates the probabilities associated with all alignments generated in this process and collects fractional counts for the concepts based on these probabilities.

2.2.3 Complexity

Training the Joint Model is even more computationally challenging than training the already demanding IBM models. Considering all possible segmentations of phrases and all their possible alignments vastly increases the number of possible alignments that can be formed between two sentences. This number is exponential with relation to the length of the shorter sentence.

E Length	F Length	No. Alignments
5	5	6721
10	10	818288740923
20	20	4.4145633531e+32
40	40	2.7340255177e+83

Table 1. The number of possible phrasal alignments for sentence pairs calculated using Stirling numbers of the second kind.

Table 1 shows just how many phrasal alignments are possible between sentences of different length. Even for medium length sentences that are 20 words in lengths, the total number of alignments is huge. Apart from being intractable, when one has a very large parameter estimation space the EM algorithm struggles to discover good parameters. One approach to dealing with this problem is to constrain the search space. For example, Pereira and Schabes (1992) proposed a method for dealing with this problem for PCFG estimation from treebanks. They encouraged the probabilities into good regions of the parameter space by constraining the search to only consider parses that did not cross Penn-Treebank nodes. We adopt a similar approach for constraining the joint model, by only considering alignments that do not contradict high probability word alignments.

During EM a very small proportion of the possible alignments are searched and many good alignments are likely to be missed. Normally alignments that are not visited in an iteration of EM, would be dropped from the phrase table, which can result in a sparseness of data. In order to avoid this, we take an approach similar to 'sparse EM' described by Neal and Hinton (1998). Counts from previous iterations are retained after being weighted lower, so as to allow the current iteration to have a significant impact on the probabilities.

3 Constraining the Joint Model

The Joint Model requires a strategy for restricting the search for phrasal alignments to areas of the alignment space which contain most of the probability mass. We propose a method which examines phrase pairs that are consistent with the set of high confidence word alignments defined for the sentence. By 'consistent' we mean that for a concept $< \overline{e}_i, \overline{f}_i >$ to be valid, we make sure that if any word in $\overline{e}_i$ is part of a high confidence alignment, then the word to which it is aligned must be included in $\overline{f}_i$ and vice versa. Phrases must still occur above a certain frequency in the corpus to be considered.

The constraints on the model are applied during both the initialisation phase and the EM phase of the training. To constrain the phrase pairs during EM, we assign a small, non-zero probability to all phrase pairs that are not consistent with the word alignments which will then only be considered when unaligned words remain after linking together high probability phrase pairs. Since all words must be aligned, there is no analogue of the NULL word present within the IBM models.

3.1 IBM Constraints

The Standard Model is based on a complex series of models, parameters and heuristics which allow it to be efficient. The Joint Probability Model is a more principled and conceptually simpler model but it is very inefficient. By using the IBM Models to constrain the Joint Model, we are searching areas in the phrasal alignment space where both models overlap. We combine the advantage of prior knowledge about likely word alignments with the ability to perform a probabilistic search around them.

The IBM constraints are the high confidence word alignments that result from taking the intersection of the bi-directional Viterbi alignments. This strategy

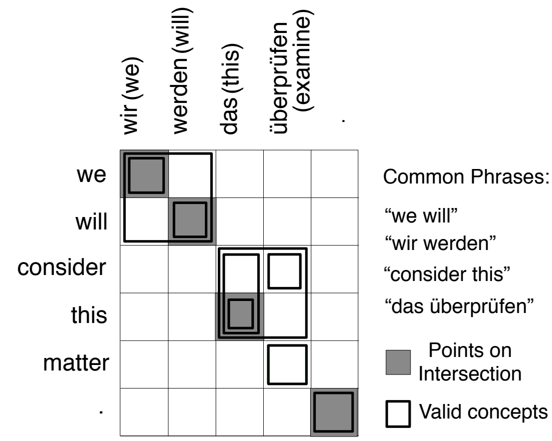

Figure 1. The area of alignment space searched using IBM constraints and applying the restriction to common phrases.

for extracting phrase pairs that are coherent with the Viterbi alignments is similar to that of the phrase-based Standard Model. However, the constrained Joint Model does not lock the search into a heuristically derived symmetrized alignment.

Figure 1 shows the space searched by the model for an example sentence. All valid concepts are consistent with all high confidence word alignments and either comprise of words or commonly occurring phrases. The concept $<$'wir','consider'$>$ would break the high confidence alignment between 'wir' and 'we' and would therefore be invalid. We can also see that the model searches more intensively areas of the sentence about which there is little certainty. Searching over an area of lower probability is preferable to using a heuristic to arbitrarily align all unaligned words. Searching allows good phrasal alignments to be discovered, for instance $<$'das überprüfen', 'consider this' $>$.

By using the IBM Models to constrain the Joint Model, we are searching areas in the phrasal alignment space where both models overlap. We combine the advantage of prior knowledge about likely word alignments with the ability to perform a probabilistic search around them.

3.2 Linguistic Constraints

By constraining the Joint Model using high confidence word alignments, any external knowledge sources can be included into the probabilistic framework. Linguistic constraints can be combined to guide the training of the Joint Model. In this paper we use a bilingual dictionary and identical words to contribute further alignment points. These constraints are combined a straightforward manner. First the IBM constraints are collected, then words that are identical in the source and target phrase and that do not contradict the IBM alignments are aligned. Finally, single word entries from the dictionary that connect a word in the source sentence to a word in the target sentence are used to align as yet unaligned words. Our dictionary includes morphological variations like plural and tense.

These linguistic constraints are useful with small sets of training data, but for larger corpora, dictionaries and identical words would contribute less to the quality of the final translations. However, the advantage of being able to include any knowledge about word alignments within a statistical model is compelling.

4 Improving the Joint Model

4.1 Lexical Weighting

The Joint Probability Model can only be trained with small amounts of parallel data and consequently the resulting parameters suffer from sparse counts. In order to make fractional counts more reliable, we can include information which encodes our prior belief about word-to-word alignments. This is desirable as word alignments are less prone to sparse statistics than phrasal alignments.

When training the Joint Model, we have initially assumed a uniform probability across all possible alignments. In a sentence, concepts of the same size will be assigned the same fractional counts. If one concept occurs more often over the entire corpus, its final parameter value will be higher. However, when the training corpus is very small, it is unlikely for the model to have seen representative occurrences of the concepts.

In order to overcome this problem, the Joint Model can use information about word-alignments generated by the IBM models. A simple way to include this knowledge is to use the high confidence points from the intersection of the bi-directional Viterbi alignments. Concepts which contain many points of high confidence will be more probable than concepts of the same size which contain none.

We define a prior count which reflects the probability of the phrasal alignment given the high confidence word alignments:

$$pc(\overline{e},\overline{f}) = \frac{|align|}{min(|\overline{e}|,|\overline{f}|)}$$

We divide the number of word alignments contained within the concept by the total number of possible word alignments for the concept, which is equal to the length of the shorter of the two phrases. We add a small fraction (0.1) to both the numerator and the denominator to smooth and avoid zero probabilities.

One way to include this prior count in the model would be to calculate it separately and then use it in the decoding process as one of the features of the log linear model. This would be similar to the lexical weighting employed by Koehn et al. (2003). In the Joint Model, however, we must perform EM and including these probabilities in the training of the model will improve the overall quality of alignments searched. These counts are thus included in the initialisation phase of the Joint Model training with the calculation of the fractional counts:

$$fc(\overline{e},\overline{f}) = (1-\lambda)p(\overline{e},\overline{f}|E,F) + \lambda pc(\overline{e},\overline{f})$$

The fractional count for each concept in each sentence is calculated by interpolating the joint probability of the concept, based on the Stirling numbers, and the prior count, which reflects the probability of the phrasal alignment given the high confidence word alignments. The use of the weight to balance the two contributions allows us to adjust for differences in scale and our confidence in each of the two measures. After testing various settings for λ the value 0.5 gave the best Bleu scores. Callison-Burch et al. (2004) used a similar technique for combining word and sentence aligned data. However, they inserted data from labelled word alignments which meant that they did not need to sum over all possible alignments for a sentence pair.

4.2 Fast Hill-climbing

The constraints on the Joint Model reduce its size by restricting the initialisation phase of the training. This is one of the two major drawbacks of the model discussed by Marcu and Wong (2002). The other major drawback is the computational cost of the training procedure. Fast hill-climbing is necessary to make EM training more tractable.

The Joint Model examines all possible swaps, splits, merges and moves for the set of concepts that have been selected as part of the initial alignment. Normal hill-climbing repeatedly performs a very expensive search over all possible steps, selecting the best step each time and applying it until no further improvement is found. In fast hill-climbing, instead of selecting only the best step, we collect all the steps that improve the probability of the initial phrasal alignment, and only search once. We then apply them one by one to the initial phrasal alignment.

This approach has the disadvantage of heavily weighting the initial alignment. All alignments generated during one iteration of EM are only one step away from the pervious alignment, so the counts for the concepts in this alignment will be high. This is a drastic change to Viterbi training, but such measures are needed to reduce the training time from nearly 5 hours to complete one iteration of EM for just 5000 sentences.

5 Experiments

The experiments were run using the German-English Europarl corpus (Koehn, 2005). Europarl contains proceedings from the European Parliament covering the years 1996-2003. The test set consisted of the standard Europarl test set of 1755 sentences which ranged from 5 to 15 words in length. This makes results directly comparable to Koehn et al. (2003). For the language model we used the SRI Language Modelling Toolkit (Stolcke, 2002) to train a trigram model on the English section of the Europarl corpus.

To perform the translations we used the Pharaoh (Koehn, 2004) beam search decoder version 1.2.8, with all the standard settings. Our evaluation metric was Bleu (Papineni et al., 2002) which compares the output sentences with human translated sentences using 4-gram precision.

The translation models are included within a log-linear model (Och and Ney, 2002) which allows a weighted combination of features functions. Only three features were used for both the Joint and the Standard Model: $p(e|f)$, $p(f|e)$ and the language model, and they were given equal weights.

5.1 Fast Hill-Climbing

EM training of the Joint Model was prohibitively slow even for the smallest data sets, so the first experiment explores the gains to be made by using fast hill-climbing on a training corpus of 5000 sentences.

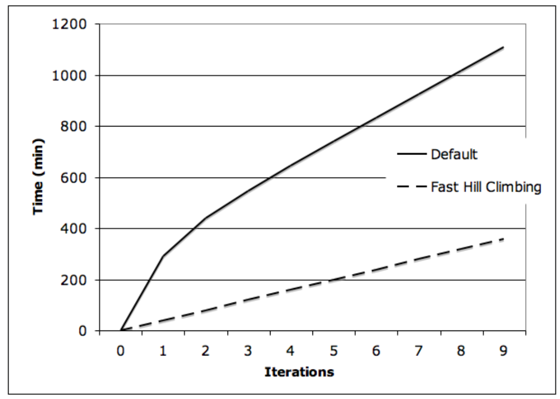

Figure 2. Time taken for EM training in minutes per iteration for 5,000 sentences on a machine with 2Gb RAM and a 2.4GHz CPU

In Figure 2 we can see that fast hill-climbing is much faster than the normal hill-climbing. We have reduced the time taken to perform the first iteration from nearly 5 hours to about 40 minutes, which is about a factor of eight.

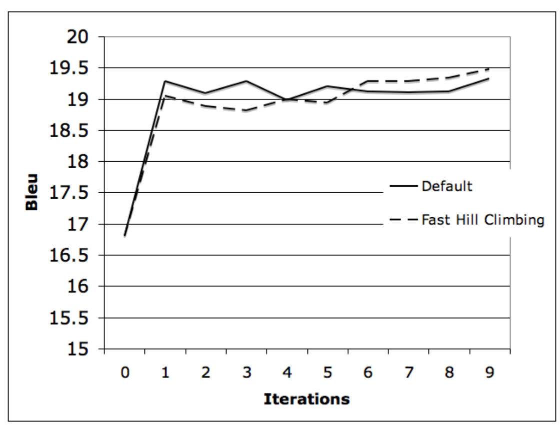

Figure 3. Bleu scores using 5,000 sentences training data

The effect of fast hill-climbing on the quality of translations can be seen in Figure 3. The default method slightly outperforms fast hill-climbing for the first few iterations, but then fast hill-climbing overtakes it. The difference in performance between the two methods is small and we apply fast hill-climbing in the remaining experiments.

5.2 IBM Constraints

The effect of applying IBM word constraints is explored by comparing it to the unconstrained Joint Model and to the Standard Model. The unconstrained Joint Model becomes intractable with very small amounts of training data. On a machine with 2 Gb of memory, we were only able to train on 10,000 parallel sentences (429234 words) of the German-English Europarl corpora. Beyond this, pruning is required to keep the model in memory during EM.

In Tables 2 and 3 we can see the differences in size and performance between the baseline model and the Joint Model for different sizes of training corpora. The unconstrained Joint Model produces a very large translation table, containing more than 6 million phrase pairs. The size of the model hampers its performance, resulting in a poor Bleu score.

By using IBM constraints, the performance of the Joint Model improves by guiding it to explore the more promising areas of the search space. It even out-performs phrase-based Standard Model. The resulting translation table is, however, still quite large and only about four times smaller than the unconstrained Joint Model. On examining the phrase pairs produced for each sentence, we discovered that the reason for the large size of the model was due to longer sentences for which there were few points of high confidence.

The lexical weighting also improve the performance of the Joint Model. The addition of information about word alignments allows for better decisions about phrasal alignments because of the sparsity of data in the phrasal alignment space.

Corpus Size	10,000	20,000	40,000
Standard Model	21.69	23.61	25.52
Joint Model	19.93	-	-
+ IBM	22.13	23.08	24.16
+ IBM + Lex	22.79	24.33	25.99

Table 2. Bleu scores for the Joint Model with IBM constraints and prior counts, corpus size indicates number of sentence pairs

Corpus Size	10,000	20,000	40,000
Standard Model	0.09	0.20	0.41
Joint Model	6.17	-	-
+ IBM	1.45	2.73	4.99
+ IBM + Lex	1.45	2.72	4.96

Table 3. Translation table size in millions of phrase pairs

5.3 Linguistic Constraints

The effect of adding linguistic constraints to the IBM word constraints is shown in tables 4 and 5.

The bilingual dictionary which comes with Ding, an open source translation program[1] was used.

Corpus Size	10,000	20,000	40,000
Standard Model	0.09	0.20	0.41
Joint + IBM + Lex	1.45	2.72	4.96
+ Ident.	1.36	2.55	4.64
+ Ident. + Dict.	1.09	2.07	3.83

Table 4. Translation table size in millions of phrase pairs when linguistic constraints are added to the Joint Model

Corpus Size	10,000	20,000	40,000
Standard Model	21.69	23.61	25.52
Joint + IBM + Lex	22.79	24.33	25.99
+ Ident.	23.30	24.90	26.12
+ Ident. + Dict.	23.20	24.96	26.13

Table 5. Bleu scores for different training corpus sizes

Table 4 shows that by adding high confidence alignments for identical words and forcing phrase pairs to be consistent with these as well as the IBM constraints, we reduce the size of the model but only slightly. Including points from the bilingual dictionary results in a sizeable reduction of about 20%.

Table 5 shows that the inclusion of lexical information into the model improves performance. The improvement in Bleu score seems to reduce with the increase in training data. As the model is trained on more data, external knowledge sources provide less advantage.

5.4 Scalability

Even though the constrained Joint Model reduces complexity, pruning is still needed in order to scale

[1] See http://www-user.tu-chemnitz.de/~fri/ding/

up to larger corpora. After the initialization phase of the training, all phrase pairs with fractional counts less than 10 million times that of the phrase pair with the highest count, are pruned from the phrase table. This reduced the phrase table from 5.38 to 2.89 million phrase pairs. The model is also parallelized in order to speed up training.

To test the scalability of the Joint Model with pruning, we performed experiments on a larger parallel corpus. We used the Spanish-English data from the 2006 HLT-NAACL Workshop on Statistical Machine Translation, which consists of 730,740 sentences of training data and 3064 test sentences which are on average much longer than the test sentences used in previous experiments. Here sparse EM was not used to avoid running out of memory. The weights of the feature functions were optimized using minimum error rate training (Och, 2003a).

	BLEU	Size
Joint + IBM	26.17	2.28
Standard Model	28.35	19.04

Table 6. Bleu scores and model size in millions of phrase pairs for Spanish-English

The results in Table 6 show that the Joint Model is capable of training on larger data sets, with a reasonable performance. On smaller data sets, as shown in sections 5.2 and 5.3 the Joint Model shows performance superior or comparable to the Standard Model. However, here it seems that the Standard Model has an advantage which is statistically significant according to the sign method (Koehn and Monz, 2006). This is almost certainly related to the fact that the Joint Model results in a much smaller phrase table. The size of the resulting Joint Model is in fact comparable to the size of the model in previous experiments when training with just 20,000 sentences. This is because the model must be kept in memory for collecting fractional counts in EM and even though the corpus is bigger, the memory available remains the same (the Standard Model phrase table is created on disk). To keep the Joint Model within memory, pruning is necessary after initialization because this is where most phrase pairs are visited. During EM, only a very small proportion of phrase pairs are visited and the model shrinks slightly with each iteration.

Pruning eliminates many phrase pairs, but further investigation indicates that this has little impact on BLEU scores. The fact that only a small proportion of the alignment space is searched is very likely to be hampering the Joint Model's performance. The small number of alignments visited leads to data sparseness and over-fitting. Another factor could be efficiency trade-offs like the fast but not optimal competitive linking search for phrasal alignments.

6 Related Work

DeNero et al. (2006) argue that training a translation model at the phrase level results in inferior parameters to the standard, heuristic phrase-based models. They suggest that the reason for this is that EM optimizes by selecting different segmentations and loses important phrase translation ambiguity. They say that the model results in a very peaked distribution and entropy drops too low. However, their argument only holds for conditional models. In a conditional model, there is competition for the probability mass of the conditioned word, and instead of spreading that mass between different translations, different segmentations will tend to be selected for.

DeNero et al.'s argmuents do not apply to a joint model. There is no such competition and the resulting phrase table's entropy is in fact higher than that of the Standard Model.

7 Conclusion

The Joint Model is a more principled way to estimate phrase translation probabilities than the ad hoc heuristics that are standardly used in SMT. The Joint Model presents challenges including a larger search space, local minima and over-fitting. But these can all be overcome straightforwardly through constraints on the search space, good initialization, bias towards retain shorter phrase pairs, and limiting the number of iterations of EM training.

In this paper we have shown that using the Joint Probability Model to estimate phrase translation probabilities results in a better performance than the heuristic approach for smaller data sets. This suggests that there are gains to be had by using a more principled statistical framework. For larger data sets performance is slightly behind the phrase-based models. This seems likely to be due to the smaller number of alignments extracted in the final iteration of EM and future work will concentrate on ways to alleviate this.

By introducing constraints to the alignment space we can greatly reduce the complexity of the model and increase its performance. The strategy of using IBM constraints with the Joint Model allows it to search areas of the alignment space with a higher probability mass, resulting in better parameters. A constrained Joint Probability Model can train on larger corpora making the model more widely applicable. Also, our particular method of constraining the Joint Model makes it easy to include linguistic information within SMT's probabilistic framework to improve alignment quality.

The Joint Model would benefit from lexical weighting like that used in many phrase-based models (Koehn et al., 2003). Using IBM Model 1 to extract a lexical alignment weight for each phrase pair would decrease the impact of data sparseness, and other kinds of smoothing techniques will be investigated. Better search algorithms for Viterbi phrasal alignments during EM would increase the number and quality of model parameters.

8 Acknowledgments

We would like to thank Philipp Koehn for access to the Pharaoh decoder and to significance testing scripts. The minimum error rate script was kindly provided by David Chiang from ISI. Thanks to Kevin O'Connor for allowing us to extend his initial implementation of the Joint Model code.

References

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Chris Callison-Burch, David Talbot, and Miles Osborne. 2004. Statistical machine translation with word- and sentence-aligned parallel corpora. In *Proceedings of ACL*.

John DeNero, Dan Gillick, James Zhang, and Dan Klein. 2006. Why generative phrase models underperform surface heuristics. In *Proceedings of NAACL Workshop on Statistical Machine Translation*.

Philipp Koehn and Christof Monz. 2006. Manual and automatic evaluation of machine translation between european languages. In *Proceedings of NAACL Workshop on Statistical Machine Translation*.

Philipp Koehn, Franz Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of HLT/NAACL*, pages 127–133.

Philipp Koehn. 2004. Pharaoh: a beam search decoder for phrase-based statistical machine translation models. In *Proceedings of AMTA*, pages 115–124.

Philipp Koehn. 2005. A parallel corpus for statistical machine translation. In *Proceedings of MT-Summit*.

Daniel Marcu and William 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*.

Radford Neal and Geoffrey Hinton, 1998. *Learning in Graphical Models*, chapter A view of the EM algorithm that justifies incremental, sparse and other variants, pages 355–368. Kluwer Academic Publishers.

Franz Josef Och and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of ACL*.

Franz Josef Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*.

Franz Josef Och. 2003a. Minimum error rate training in statistical machine translation. In *Proceedings of ACL*.

Franz Josef Och. 2003b. *Statistical Machine Translation: From Single-Word Models to Alignment Templates*. Ph.D. thesis, RWTH Aachen Department of Computer Science, Aachen, Germany.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL*, pages 311–318.

Fernando Pereira and Yves Schabes. 1992. Inside-outside reestimation from partially bracketed corpora. In *Proceedings of ACL*, pages 128–135.

Mark Przybocki. 2004. NIST 2004 machine translation evaluation results. Confidential e-mail to workshop participants, May.

Andreas Stolcke. 2002. SRILM - an extensible language modeling toolkit. In *Proceedings of Spoken Language Processing*, pages 901–904.

Stephan Vogel, Ying Zhang, Fei Huang, Alicia Tribble, Ashish Venugopal, Bing Zhao, and Alex Waibel. 2003. The CMU statistical machine translation system. In *Machine Translation Summit*.