An Iteratively-Trained Segmentation-Free Phrase Translation Model for Statistical Machine Translation

Robert C. Moore **Chris Quirk**
Microsoft Research
Redmond, WA 98052, USA
{bobmoore,chrisq}@microsoft.com

Abstract

Attempts to estimate phrase translation probablities for statistical machine translation using iteratively-trained models have repeatedly failed to produce translations as good as those obtained by estimating phrase translation probablities from surface statistics of bilingual word alignments as described by Koehn, et al. (2003). We propose a new iteratively-trained phrase translation model that produces translations of quality equal to or better than those produced by Koehn, et al.'s model. Moreover, with the new model, translation quality degrades much more slowly as pruning is tightend to reduce translation time.

1 Introduction

Estimates of conditional phrase translation probabilities provide a major source of translation knowledge in phrase-based statistical machine translation (SMT) systems. The most widely used method for estimating these probabilities is that of Koehn, et al. (2003), in which phrase pairs are extracted from word-aligned bilingual sentence pairs, and their translation probabilities estimated heuristically from surface statistics of the extracted phrase pairs. We will refer to this approach as "the standard model".

There have been several attempts to estimate phrase translation probabilities directly, using generative models trained iteratively on a parallel corpus using the Expectation Maximization (EM) algorithm. The first of these models, that of Marcu and Wong (2002), was found by Koehn, et al. (2003), to produce translations not quite as good as their method. Recently, Birch et al. (2006) tried the Marcu and Wong model constrained by a word alignment and also found that Koehn, et al.'s model worked better, with the advantage of the standard model increasing as more features were added to the overall translation model. DeNero et al. (2006) tried a different generative phrase translation model analogous to IBM word-translation Model 3 (Brown et al., 1993), and again found that the standard model outperformed their generative model.

DeNero et al. (2006) attribute the inferiority of their model and the Marcu and Wong model to a hidden segmentation variable, which enables the EM algorithm to maximize the probability of the training data without really improving the quality of the model. We propose an iteratively-trained phrase translation model that does not require different segmentations to compete against one another, and we show that this produces translations of quality equal to or better than those produced by the standard model. We find, moreover, that with the new model, translation quality degrades much more slowly as pruning is tightend to reduce translation time.

Decoding efficiency is usually considered only in the design and implementation of decoding algorithms, or the choice of model structures to support faster decoding algorithms. We are not aware of any attention previously having been paid to the effect of different methods of parameter estimation on translation efficiency for a given model structure.

The time required for decoding is of great importance in the practical application of SMT tech-

Proceedings of the Second Workshop on Statistical Machine Translation, pages 112–119,
Prague, June 2007.

nology. One of the criticisms of SMT often made by adherents of rule-based machine translation is that SMT is too slow for practical application. The rapidly falling price of computer hardware has ameliorated this problem to a great extent, but the fact remains that every factor of 2 improvement in translation efficiency means a factor of 2 decrease in hardware cost for intensive applications of SMT, such as a web-based translation service ("Translate this page"). SMT surely needs all the help in can get in this regard.

2 Previous Approaches

Koehn, et al.'s (2003) method of estimating phrase-translation probabilities is very simple. They start with an automatically word-aligned corpus of bilingual sentence pairs, in which certain words are linked, indicating that they are translations of each other, or that they are parts of phrases that are translations of each other. They extract every possible phrase pair (up to a given length limit) that (a) contains at least one pair of linked words, and (b) does not contain any words that have links to other words not included in the phrase pair.[1] In other words, word alignment links cannot cross phrase pair boundaries. Phrase translation probabilities are estimated simply by marginalizing the counts of phrase instances:

$$p(x|y) = \frac{C(x,y)}{\sum_{x'} C(x',y)}$$

This method is used to estimate the conditional probabilities of both target phrases give source phrases and source phrases given target phrases.

In contrast to the standard model, DeNero, et al. (2006) estimate phrase translation probabilities according to the following generative model:

1. Begin with a source sentence a.

2. Stochastically segment a into some number of phrases.

3. For each selected phrase in a, stochastically choose a phrase position in the target sentence b that is being generated.

4. For each selected phrase in a and the corresponding phrase position in b, stochastically choose a target phrase.

5. Read off the target sentence b from the sequence of target phrases.

DeNero et al.'s analysis of why their model performs relatively poorly hinges on the fact that the segmentation probabilities used in step 2 are, in fact, not trained, but simply assumed to be uniform. Given complete freedom to select whatever segmentation maximizes the likelihood of any given sentence pair, EM tends to favor segmentations that yield source phrases with as few occurrences as possible, since more of the associated conditional probability mass can be concentrated on the target phrase alignments that are possible in the sentence at hand. Thus EM tends to maximize the probability of the training data by concentrating probability mass on the rarest source phrases it can construct to cover the training data. The resulting probability estimates thus have less generalizability to unseen data than if probability mass were concentrated on more frequently occurring source phrases.

3 A Segmentation-Free Model

To avoid the problem identified by DeNero et al., we propose an iteratively-trained model that does not assume a segmentation of the training data into non-overlapping phrase pairs. We refer to our model as "iteratively-trained" rather than "generative" because we have not proved any of the mathematical properties usually associated with generative models; e.g., that the training procedure maximizes the likelihood of the training data. We will motivate the model, however, with a generative story as to how phrase alignments are produced, given a pair of source and target sentences. Our model extends to phrase alignment the concept of a sentence pair generating a word alignment developed by Cherry and Lin (2003).

Our model is defined in terms of two stochastic processes, *selection* and *alignment*, as follows:

1. For each word-aligned sentence pair, we identify all the possible phrase pair instances according to the criteria used by Koehn et al.

[1]This method of phrase pair extraction was originally described by Och et al. (1999).

2. Each source phrase instance that is included in any of the possible phrase pair instances independently selects one of the target phrase instances that it forms a possible phrase pair instance with.

3. Each target phrase instance that is included in any of the possible phrase pair instances independently selects one of the source phrase instances that it forms a possible phrase pair instance with.

4. A source phrase instance is aligned to a target phrase instance, if and only if each selects the other.

Given a set of selection probability distributions and a word-aligned parallel corpus, we can easily compute the expected number of alignment instances for a given phrase pair type. The probability of a pair of phrase instances x and y being aligned is simply $p_s(x|y) \times p_s(y|x)$, where p_s is the applicable selection probability distribution. The expected number of instances of alignment, $E(x, y)$, for the pair of phrases x and y, is just the sum of the alignment probabilities of all the possible instances of that phrase pair type.

From the expected number of alignments and the total number of occurrences of each source and target phrase type in the corpus (whether or not they particpate in possible phrase pairs), we estimate the conditional phrase translation probabilities as

$$p_t(y|x) = \frac{E(x,y)}{C(x)}, \quad p_t(x|y) = \frac{E(x,y)}{C(y)},$$

where E denotes expected counts, and C denotes observed counts.

The use of the total observed counts of particular source and target phrases (instead of marginalized expected joint counts) in estimating the conditional phrase translation probabilities, together with the multiplication of selection probabilities in computing the alignment probability of particular phrase pair instances, causes the conditional phrase translation probability distributions generally to sum to less than 1.0. We interpret the missing probability mass as the probability that a given word sequence does not translate as any contiguous word sequence in the other language.

We have seen how to derive phrase translation probabilities from the selection probabilities, but where do the latter come from? We answer this question by adding the following constraint to the model:

> The probabilty of a phrase y selecting a phrase x is proportional to the probability of x translating as y, normalized over the possible non-null choices for x presented by the word-aligned sentence pair.

Symbolically, we can express this as

$$p_s(x|y) = \frac{p_t(y|x)}{\sum_{x'} p_t(y|x')}$$

where p_s denotes selection probability, p_t denotes translation probability, and x' ranges over the phrase instances that could possibly align to y. We are, in effect, inverting and renormalizing translation probabilities to get selection probabilities. The reason for the inversion may not be immediately apparent, but it in fact simply generalizes the e-step formula in the EM training for IBM Model 1 from words to phrases.

This model immediately suggests (and, in fact, was designed to suggest) the following EM-like training procedure:

1. Initialize the translation probability distributions to be uniform. (It doesn't matter at this point whether the possibility of no translation is included or not.)

2. E step: Compute the expected phrase alignment counts according to the model, deriving the selection probabilities from the current estimates of the translation probabilities as described.

3. M step: Re-estimate the phrase translation probabilities according to the expected phrase alignment counts as described.

4. Repeat the E and M steps, until the desired degree of convergence is obtained.

We view this training procedure as iteratively trying to find a set of phrase translation probabilities that satisfies all the constraints of the model, although we have not proved that this training procedure always converges. We also have not proved that the procedure maximizes the likelihood of anything, although we find empirically that each iteration decreases the conditional entropy of the phrase translation model. In any case, the training procedure seems to work well in practice. It is also very similar to the joint training procedure for HMM word-alignment models in both directions described by Liang et al. (2006), which was the original inspiration for our training procedure.

4 Experimental Set-Up and Data

We evaluated our phrase translation model compared to the standard model of Koehn et al. in the context of a fairly typical end-to-end phrase-based SMT system. The overall translation model score consists of a weighted sum of the following eight aggregated feature values for each translation hypothesis:

- the sum of the log probabilities of each source phrase in the hypothesis given the corresponding target phrase, computed either by our model or the standard model,
- the sum of the log probabilities of each target phrase in the hypothesis given the corresponding source phrase, computed either by our model or the standard model,
- the sum of lexical scores for each source phrase given the corresponding target phrase,
- the sum of lexical scores for each target phrase given the corresponding source phrase,
- the log of the target language model probability for the sequence of target phrases in the hypothesis,
- the total number of words in the target phrases in the hypothesis,
- the total number of source/target phrase pairs composing the hypothesis,
- the distortion penalty as implemented in the Pharaoh decoder (Koehn, 2003).

The lexical scores are computed as the (unnormalized) log probability of the Viterbi alignment for a phrase pair under IBM word-translation Model 1 (Brown et al., 1993). The feature weights for the overall translation models were trained using Och's (2003) minimum-error-rate training procedure. The weights were optimized separately for our model and for the standard phrase translation model. Our decoder is a reimplementation in Perl of the algorithm used by the Pharaoh decoder as described by Koehn (2003).[2]

The data we used comes from an English-French bilingual corpus of Canadian Hansards parliamentary proceedings supplied for the bilingual word alignment workshop held at HLT-NAACL 2003 (Mihalcea and Pedersen, 2003). Automatic sentence alignment of this data was provided by Ulrich Germann. We used 500,000 sentences pairs from this corpus for training both the phrase translation models and IBM Model 1 lexical scores. These 500,000 sentence pairs were word-aligned using a state-of-the-art word-alignment method (Moore et al., 2006). A separate set of 500 sentence pairs was used to train the translation model weights, and two additional held-out sets of 2000 sentence pairs each were used as test data.

The two phrase translation models were trained using the same set of possible phrase pairs extracted from the word-aligned 500,000 sentence pair corpus, finding all possible phrase pairs permitted by the criteria followed by Koehn et al., up to a phrase length of seven words. This produced approximately 69 million distinct phrase pair types. No pruning of the set of possible phrase pairs was done during or before training the phrase translation models. Our phrase translation model and IBM Model 1 were both trained for five iterations. The training procedure for our phrase translation model trains models in both directions simultaneously, but for IBM Model 1, models were trained separately in each direction. The models were then pruned to include only phrase pairs that matched the source sides of the small training and test sets.

5 Entropy Measurements

To verify that our iterative training procedure was behaving as expected, after each training iteration

[2]Since Perl is a byte-code interpreted language, absolute decoding times will be slower than with the standard machine-language-compiled implementation of Pharaoh, but relative times between models should be comparable.

we measured the conditional entropy of the model in predicting English phrases given French phrases, according to the formula

$$H(E|F) = \sum_f p(f) \sum_e p_t(e|f) \log_2 p_t(e|f),$$

where e and f range over the English and French phrases that occur in the extracted phrase pairs, and $p(f)$ was estimated according to the relative frequency of these French phrases in a 2000 sentence sample of the French sentences from the 500,000 word-aligned sentence pairs. Over the five training iterations, we obtained a monotonically decreasing sequence of entropy measurements in bits per phrase: 1.329, 1.177, 1.146, 1.140, 1.136.

We also compared the conditional entropy of the standard model to the final iteration of our model, estimating $p(f)$ using the first of our 2000 sentence pair test sets. For this data, our model measured 1.38 bits per phrase, and the standard model measured 4.30 bits per phrase. DeNero et al. obtained corresponding measurements of 1.55 bits per phrase and 3.76 bits per phrase, for their model and the standard model, using a different data set and a slightly different estimation method.

6 Translation Experiments

We wanted to look at the trade-off between decoding time and translation quality for our new phrase translation model compared to the standard model. Since this trade-off is also affected by the settings of various pruning parameters, we compared decoding time and translation quality, as measured by BLEU score (Papineni et al, 2002), for the two models on our first test set over a broad range of settings for the decoder pruning parameters.

The Pharaoh decoding algorithm, has five pruning parameters that affect decoding time:

- Distortion limit
- Translation table limit
- Translation table threshold
- Beam limit
- Beam threshold

The distortion limit is the maximum distance allowed between two source phrases that produce adjacent target phrases in the decoder output. The distortion limit can be viewed as a model parameter, as well as a pruning paramter, because setting it to an optimum value usually improves translation quality over leaving it unrestricted. We carried out experiments with the distortion limit set to 1, which seemed to produce the highest BLEU scores on our data set with the standard model, and also set to 5, which is perhaps a more typical value for phrase-based SMT systems. Translation model weights were trained separately for these two settings, because the greater the distortion limit, the higher the distortion penalty weight needed for optimal translation quality.

The translation table limit and translation table threshold are applied statically to the phrase translation table, which combines all components of the overall translation model score that can be computed for each phrase pair in isolation. This includes all information except the distortion penalty score and the part of the language model score that looks at n-grams that cross target phrase boundaries. The translation table limit is the maximum number of translations allowed in the table for any given source phrase. The translation table threshold is the maximum difference in combined translation table score allowed between the highest scoring translation and lowest scoring translation for any given source phrase. The beam limit and beam threshold are defined similarly, but they apply dynamically to the sets of competing partial hypotheses that cover the same number of source words in the beam search for the highest scoring translation.

For each of the two distortion limits we tried, we carried out a systematic search for combinations of settings of the other four pruning parameters that gave the best trade-offs between decoding time and BLEU score. Starting at a setting of 0.5 for the threshold parameters[3] and 5 for the limit parameters we performed a hill-climbing search over step-wise relaxations of all combinations of the four parame-

[3] We use difference in weighted linear scores directly for our pruning thresholds, whereas the standard implementation of Pharaoh expresses these as probability ratios. Hence the specific values for these parameters are not comparable to published descriptions of experiments using Pharaoh, although the effects of pruning are exactly the same.

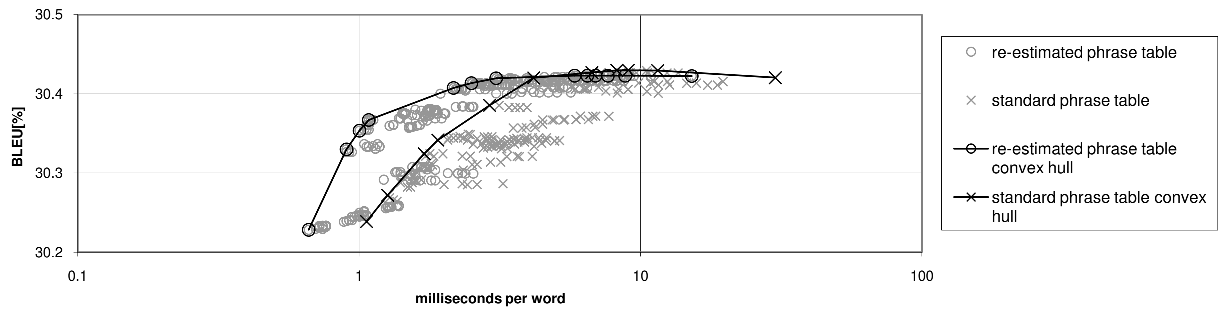

Figure 1: BLEU vs Decoding Time (DL = 1)

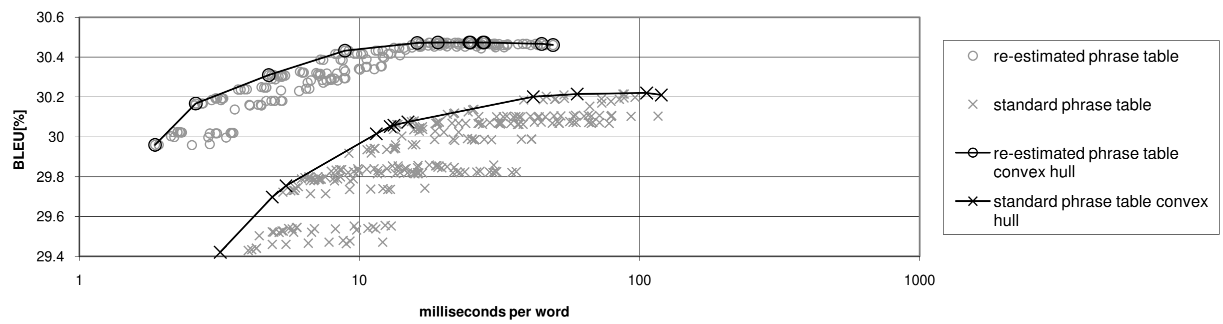

Figure 2: BLEU vs Decoding Time (DL = 5)

ters, incrementing the threshold parameters by 0.5 and the limit parameters by 5 at each step. For each resulting point that provided the best BLEU score yet seen for the amount of decoding time used, we iterated the search.

The resulting possible combinations of BLEU score and decoding time for the two phrase translation models are displayed in Figure 1, for a distortion limit of 1, and Figure 2, for a distortion limit of 5. BLEU score is reported on a scale of 1–100 (BLEU[%]), and decoding time is measured in milliseconds per word. Note that the decoding time axis is presented on a log scale.

The points that represent pruning parameter settings one might consider using in a practical system are those on or near the upper convex hull of the set of points for each model. These upper-convex-hull points are highlighted in the figures. Points far from these boundaries represent settings of one or more of the parameters that are too restrictive to obtain good translation quality, together with settings of other parameters that are too permissive to obtain good translation time.

Examining the results for a distortion limit of 1, we found that the BLEU score obtained with the loosest pruning parameter settings (2.5 for both threshold paramters, and 25 for both limit parameters) were essentially identical for the two models: 30.42 BLEU[%]. As the pruning parameters are tightened to reduce decoding time, however, the new model performs much better. At a decoding time almost 6 times faster than for the settings that produced the highest BLEU score, the change in score was only -0.07 BLEU[%] with the new model. To obtain a slightly worse[4] BLEU score (-0.08 BLEU[%]) using the standard model took 90% more decoding time.

It does appear, however, that the best BLEU score for the standard model is slightly better than the best BLEU score for the new model: 30.43 vs. 30.42. It is in fact currious that there seem to be numerous points where the standard model gets a slightly

[4]Points on the convex hulls with exactly comparable BLEU scores do not often occur.

better BLEU score than it does with with the loosest pruning settings, which should have the lowest search error.

We conjectured that this might be an artifact of our test procedure. If a model is at all reasonable, most search errors will reduce the ultimate objective function, in our case the BLEU score, but occasionally a search error will increase the objective function just by chance. The smaller the number of search errors in a particular test, the greater the likelihood that, by chance, more search errors will increase the objective function than decrease it. Since we are sampling a fairly large number of combinations of pruning parameter settings (179 for the standard model with a distortion limit of 1), it is possible that a small number of these have more "good" search errors than "bad" search errors simply by chance, and that this accounts for the small number of points (13) at which the BLEU score exceeds that of the point which should have the fewest search errors. This effect may be more pronounced with the standard model than with the new model, simply because there is more noise in the standard model.

To test the hypothesis that the BLEU scores greater than the score for the loosest pruning settings simply represent noise in the data, we collected all the pruning settings that produced BLEU scores greater than or equal to the the one for the loosest pruning settings, and evaluated the standard model at those settings on our second held-out test set. We then looked at the correlation between the BLEU scores for these settings on the two test sets, and found that it was very small and negative, with $r = -0.099$. The standard F-test for the significance of a correlation yielded $p = 0.74$; in other words, completely insignificant. This strongly suggests that the apparent improvement in BLEU score for certain tighter pruning settings is illusory.

As a sanity check, we tested the BLEU score correlation between the two test sets for the points on the upper convex hull of the plot for the standard model, between the point with the fastest decoding time and the point with the highest BLEU score. That correlation was very high, with $r = 0.94$, which was significant at the level $p = 0.0004$ according to the F-test. Thus the BLEU score differences along most of the upper convex hull seem to reflect reality, but not in the region where they equal or exceed the score for the loosest pruning settings.

At a distortion limit of 5, there seems no question that the new model performs better than the standard model. The difference BLEU scores for the upper-convex-hull points ranges from about 0.8 to 0.2 BLEU[%] for comparable decoding times. Again, the advantage of the new model is greater at shorter decoding times. Compared to the results with a distortion limit of 1, the standard model loses translation quality, with a change of about -0.2 BLEU[%] for the loosest pruning settings, while the new model gains very slightly ($+0.04$ BLEU[%]).

7 Conclusions

This study seems to confirm DeNero et al.'s diagnosis that the main reason for poor performance of previous iteratively-trained phrase translation models, compared to Koehn et al.'s model, is the effect of the hidden segmentation variable in these models. We have developed an iteratively-trained phrase translation model that is segmentation free, and shown that, at a minimum, it eliminates the shortfall in BLEU score compared to the standard model. With a larger distortion limit, the new model produced translations with a noticably better BLEU score.

From a practical point of view, the main result is probably that BLEU score degrades much more slowly with our model than with the standard model, when the decoding search is tuned for speed. For some settings that appear reasonable, this difference is close to a factor of 2, even if there is no difference in the translation quality obtainable when pruning is loosened. For high-demand applications like web page translation, roughly half of the investment in translation servers could be saved while providing this level of translation quality with the same response time.

Acknowledgement

The authors would like to thank Mark Johnson for many valuable discussions of how to analyze and present the results obtained in this study.

References

Alexandra Birch, Chris Callison-Burch, Miles Osborne, and Philipp Koehn. 2006. Constraining the Phrase-Based, Joint Probability Statistical Translation Model. In *Proceedings of the HLT-NAACL 06 Workshop, Statistical Machine Translation*, pp. 154–157, New York City, New York, USA.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263–311.

Colin Cherry and Dekang Lin. 2003. A Probability Model to Improve Word Alignment. In *Proceedings of the 41st Annual Meeting of the ACL*, pp. 88–95, Sapporo, Japan.

John DeNero, Dan Gillick, James Zhang, and Dan Klein. 2006. Why Generative Phrase Models Underperform Surface Heuristics. In *Proceedings of the HLT-NAACL 06 Workshop, Statistical Machine Translation*, pp. 31–38, New York City, New York, USA.

Philipp Koehn. 2003. Noun Phrase Translation. PhD Dissertation, Computer Science, University of Southern California, Los Angeles, California, USA.

Philipp Koehn, Franz Joseph Och, and Daniel Marcu. 2003. Statistical Phrase-Based Translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 127–133, Edmonton, Alberta, Canada.

Percy Liang, Ben Taskar, and Dan Klein. 2006. Alignment by Agreement. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 104–111, New York City, New York, USA.

Daniel Marcu and William Wong. 2002. A Phrase-Based, Joint Probability Model for Statistical Machine Translation. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing*, pp. 133–139, Philadelphia, Pennsylvania, USA.

Rada Mihalcea and Ted Pedersen. 2003. An Evaluation Exercise for Word Alignment. In *Proceedings of the HLT-NAACL 2003 Workshop, Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pp. 1–6, Edmonton, Alberta, Canada.

Robert C. Moore, Wen-tau Yih, and Andreas Bode. 2006. Improved Discriminative Bilingual Word Alignment. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp. 513-520, Sydney, Australia.

Franz Joseph Och, Christoff Tillmann, and Hermann Ney. 1999. Improved Alignment Models for Statistical Machine Translation. In *Proceedings of the 1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pp. 20–28, College Park, Maryland, USA.

Franz Joseph Och. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *Proceedings of the 41st Annual Meeting of the ACL*, pp. 160–167, Sapporo, Japan.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 311–318, Philadelphia, Pennsylvania, USA.