Statistical Machine Translation through Global Lexical Selection and Sentence Reconstruction

Srinivas Bangalore, Patrick Haffner, Stephan Kanthak
AT&T Labs - Research
180 Park Ave, Florham Park, NJ 07932
{srini,haffner,skanthak}@research.att.com

Abstract

Machine translation of a source language sentence involves selecting appropriate target language words and ordering the selected words to form a well-formed target language sentence. Most of the previous work on statistical machine translation relies on (*local*) associations of target words/phrases with source words/phrases for lexical selection. In contrast, in this paper, we present a novel approach to lexical selection where the target words are associated with the entire source sentence (*global*) without the need to compute local associations. Further, we present a technique for reconstructing the target language sentence from the selected words. We compare the results of this approach against those obtained from a finite-state based statistical machine translation system which relies on local lexical associations.

1 Introduction

Machine translation can be viewed as consisting of two subproblems: (a) lexical selection, where appropriate target language lexical items are chosen for each source language lexical item and (b) lexical reordering, where the chosen target language lexical items are rearranged to produce a meaningful target language string. Most of the previous work on statistical machine translation, as exemplified in (Brown et al., 1993), employs word-alignment algorithm (such as GIZA++ (Och and Ney, 2003)) that provides local associations between source and target words. The source-to-target word alignments are sometimes augmented with target-to-source word alignments in order to improve precision. Further, the word-level alignments are extended to phrase-level alignments in order to increase the extent of local associations. The phrasal associations compile some amount of (*local*) lexical reordering of the target words – those permitted by the size of the phrase. Most of the state-of-the-art machine translation systems use phrase-level associations in conjunction with a target language model to produce sentences. There is relatively little emphasis on (*global*) lexical reordering other than the local reorderings permitted within the phrasal alignments. A few exceptions are the hierarchical (possibly syntax-based) transduction models (Wu, 1997; Alshawi et al., 1998; Yamada and Knight, 2001; Chiang, 2005) and the string transduction models (Kanthak et al., 2005).

In this paper, we present an alternate approach to lexical selection and lexical reordering. For lexical selection, in contrast to the local approaches of associating target to source words, we associate target words to the entire source sentence. The intuition is that there may be lexico-syntactic features of the source sentence (not necessarily a single source word) that might trigger the presence of a target word in the target sentence. Furthermore, it might be difficult to exactly associate a target word to a source word in many situations – (a) when the translations are not exact but paraphrases (b) when the target language does not have one lexical item to express the same concept that is expressed by a source word. Extending word to phrase alignments attempts to address some of these situations while alleviating the noise in word-level alignments.

As a consequence of this global lexical selection approach, we no longer have a tight association between source and target language words. The result of lexical selection is simply a bag of words in the target language and the sentence has to be reconstructed using this bag of words. The words in the bag, however, might be enhanced with rich syntactic information that could aid in reconstructing the target sentence. This approach to lexical selection and

Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pages 152–159,
Prague, Czech Republic, June 2007.

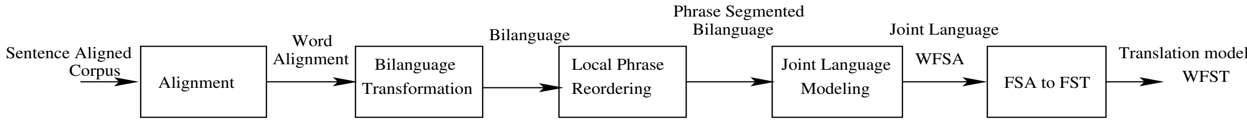

Figure 1: Training phases for our system

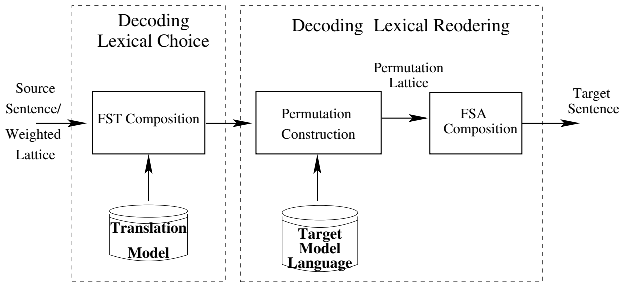

Figure 2: Decoding phases for our system

sentence reconstruction has the potential to circumvent limitations of word-alignment based methods for translation between languages with significantly different word order (e.g. English-Japanese).

In this paper, we present the details of training a global lexical selection model using classification techniques and sentence reconstruction models using permutation automata. We also present a stochastic finite-state transducer (SFST) as an example of an approach that relies on local associations and use it to compare and contrast our approach.

2 SFST Training and Decoding

In this section, we describe each of the components of our SFST system shown in Figure 1. The SFST approach described here is similar to the one described in (Bangalore and Riccardi, 2000) which has subsequently been adopted by (Banchs et al., 2005).

2.1 Word Alignment

The first stage in the process of training a lexical selection model is obtaining an alignment function (f) that given a pair of source ($s_1 s_2 \ldots s_n$) and target ($t_1 t_2 \ldots t_m$) language sentences, maps source language word subsequences into target language word subsequences, as shown below.

$$\forall i \exists j (f(s_i) = t_j \vee f(s_i) = \epsilon) \qquad (1)$$

For the work reported in this paper, we have used the GIZA++ tool (Och and Ney, 2003) which implements a string-alignment algorithm. GIZA++ alignment however is asymmetric in that the word mappings are different depending on the direction of alignment – source-to-target or target-to-source. Hence in addition to the functions f as shown in Equation 1 we train another alignment function g :

$$\forall j \exists i (g(t_j) = s_i \vee g(t_j) = \epsilon) \qquad (2)$$

English: I need to make a collect call
Japanese: 私は　コレクト　コールを かける　必要があります
Alignment: 1 5 0 3 0 2 4

Figure 3: Example bilingual texts with alignment information

I:私は need:必要があります to:ϵ make:コールを a:ϵ collect_コレクト call_かける

Figure 4: Bilanguage strings resulting from alignments shown in Figure 3.

2.2 Bilanguage Representation

From the alignment information (see Figure 3), we construct a bilanguage representation of each sentence in the bilingual corpus. The bilanguage string consists of source-target symbol pair sequences as shown in Equation 3. Note that the tokens of a bilanguage could be either ordered according to the word order of the source language or ordered according to the word order of the target language.

$$\begin{aligned} B^f &= b_1^f \, b_2^f \, \ldots \, b_m^f \qquad (3) \\ b_i^f &= (s_{i-1}; s_i, f(s_i)) \;\; if \;\; f(s_{i-1}) = \epsilon \\ &= (s_i, f(s_{i-1}); f(s_i)) \;\; if \;\; s_{i-1} = \epsilon \\ &= (s_i, f(s_i)) \;\; otherwise \end{aligned}$$

Figure 4 shows an example alignment and the source-word-ordered bilanguage strings corresponding to the alignment shown in Figure 3.

We also construct a bilanguage using the alignment function g similar to the bilanguage using the alignment function f as shown in Equation 3.

Thus, the bilanguage corpus obtained by combining the two alignment functions is $B = B_f \cup B_g$.

2.3 Bilingual Phrases and Local Reordering

While word-to-word translation only approximates the lexical selection process, phrase-to-phrase mapping can greatly improve the translation of collocations, recurrent strings, etc. Using phrases also allows words within the phrase to be reordered into the correct target language order, thus partially solving the reordering problem. Additionally, SFSTs can take advantage of phrasal correlations to improve the computation of the probability $P(W_S, W_T)$.

The bilanguage representation could result in some source language phrases to be mapped to ϵ

(empty target phrase). In addition to these phrases, we compute subsequences of a given length k on the bilanguage string and for each subsequence we reorder the target words of the subsequence to be in the same order as they are in the target language sentence corresponding to that bilanguage string. This results in a retokenization of the bilanguage into tokens of source-target phrase pairs.

2.4 SFST Model

From the bilanguage corpus B, we train an n-gram language model using standard tools (Goffin et al., 2005). The resulting language model is represented as a weighted finite-state automaton ($S \times T \rightarrow [0,1]$). The symbols on the arcs of this automaton ($s_i\_t_i$) are interpreted as having the source and target symbols (s_i:t_i), making it into a weighted finite-state transducer ($S \rightarrow T \times [0,1]$) that provides a weighted string-to-string transduction from S into T :

$$T^* = \underset{T}{argmax}\ P(s_i, t_i | s_{i-1}, t_{i-1} \ldots s_{i-n-1}, t_{i-n-1})$$

2.5 Decoding

Since we represent the translation model as a weighted finite-state transducer ($TransFST$), the decoding process of translating a new source input (sentence or weighted lattice (I_s)) amounts to a transducer composition ($\circ$) and selection of the best probability path ($BestPath$) resulting from the composition and projecting the target sequence (π_1).

$$T^* = \pi_1(BestPath(I_s \circ TransFST)) \quad (4)$$

However, we have noticed that on the development corpus, the decoded target sentence is typically shorter than the intended target sentence. This mismatch may be due to the incorrect estimation of the back-off events and their probabilities in the training phase of the transducer. In order to alleviate this mismatch, we introduce a negative word insertion penalty model as a mechanism to produce more words in the target sentence.

2.6 Word Insertion Model

The word insertion model is also encoded as a weighted finite-state automaton and is included in the decoding sequence as shown in Equation 5. The word insertion FST has one state and $|\sum_T|$ number of arcs each weighted with a λ weight representing the word insertion cost. On composition as shown in Equation 5, the word insertion model penalizes or rewards paths which have more words depending on whether λ is positive or negative value.

$$T^* = \pi_1(BestPath(I_s \circ TransFST \circ WIP)) \quad (5)$$

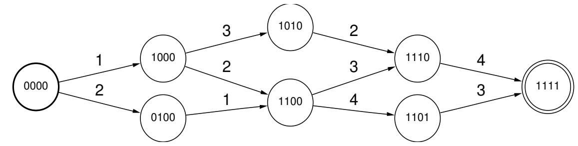

Figure 5: Locally constraint permutation automaton for a sentence with 4 words and window size of 2.

2.7 Global Reordering

Local reordering as described in Section 2.3 is restricted by the window size k and accounts only for different word order within phrases. As permuting non-linear automata is too complex, we apply global reordering by permuting the words of the best translation and weighting the result by an n-gram language model (see also Figure 2):

$$T^* = BestPath(perm(T') \circ LM_t) \quad (6)$$

Even the size of the minimal permutation automaton of a linear automaton grows exponentially with the length of the input sequence. While decoding by composition simply resembles the principle of memoization (i.e. here: all state hypotheses of a whole sentence are kept in memory), it is necessary to either use heuristic forward pruning or constrain permutations to be within a local window of adjustable size (also see (Kanthak et al., 2005)). We have chosen to constrain permutations here. Figure 5 shows the resulting minimal permutation automaton for an input sequence of 4 words and a window size of 2.

Decoding ASR output in combination with global reordering uses n-best lists or extracts them from lattices first. Each entry of the n-best list is decoded separately and the best target sentence is picked from the union of the n intermediate results.

3 Discriminant Models for Lexical Selection

The approach from the previous section is a generative model for statistical machine translation relying on local associations between source and target sentences. Now, we present our approach for a *global* lexical selection model based on discriminatively trained classification techniques. Discriminant modeling techniques have become the dominant method for resolving ambiguity in speech and other NLP tasks, outperforming generative models. Discriminative training has been used mainly for translation model combination (Och and Ney, 2002) and with the exception of (Wellington et al., 2006; Tillmann and Zhang, 2006), has not been used to directly train parameters of a translation model. We expect discriminatively trained global lexical selection models to outperform generatively trained local lexical selection models as well as provide a framework for incorporating rich morpho-syntactic information.

Statistical machine translation can be formulated as a search for the best target sequence that maximizes $P(T|S)$, where S is the source sentence and T is the target sentence. Ideally, $P(T|S)$ should be estimated directly to maximize the conditional likelihood on the training data (discriminant model). However, T corresponds to a sequence with a exponentially large combination of possible labels, and traditional classification approaches cannot be used directly. Although Conditional Random Fields (CRF) (Lafferty et al., 2001) train an exponential model at the sequence level, in translation tasks such as ours the computational requirements of training such models are prohibitively expensive.

We investigate two approaches to approximating the string level global classification problem, using different independence assumptions. A comparison of the two approaches is summarized in Table 1.

3.1 Sequential Lexical Choice Model

In the first approach, we formulate a sequential local classification problem as shown in Equations 7. This approach is similar to the SFST approach in that it relies on local associations between the source and target words(phrases). We can use a conditional model (instead of a joint model as before) and the parameters are determined using discriminant training which allows for richer conditioning context.

$$P(T|S) = \prod_{i=1}^{N} P(t_i|\Phi(S,i)) \tag{7}$$

where $\Phi(S,i)$ is a set of features extracted from the source string S (shortened as Φ in the rest of the section).

3.2 Bag-of-Words Lexical Choice Model

The sequential lexical choice model described in the previous section treats the selection of a lexical choice for a source word in the local lexical context as a classification task. The data for training such models is derived from word alignments obtained by e.g. GIZA++. The decoded target lexical items have to be further reordered, but for closely related languages the reordering could be incorporated into correctly ordered target phrases as discussed previously.

For pairs of languages with radically different word order (e.g. English-Japanese), there needs to be a global reordering of words similar to the case in the SFST-based translation system. Also, for such differing language pairs, the alignment algorithms such as GIZA++ perform poorly.

These observations prompted us to formulate the lexical choice problem *without* the need for word alignment information. We require a sentence aligned corpus as before, but we treat the target sentence as a bag-of-words or BOW assigned to the source sentence. The goal is, given a source sentence, to estimate the probability that we find a given word in the target sentence. This is why, instead of producing a target sentence, what we initially obtain is a target bag of words. Each word in the target vocabulary is detected independently, so we have here a very simple use of binary static classifiers. Training sentence pairs are considered as positive examples when the word appears in the target, and negative otherwise. Thus, the number of training examples equals the number of sentence pairs, in contrast to the sequential lexical choice model which has one training example for each token in the bilingual training corpus. The classifier is trained with n-gram features ($BOgrams(S)$) from the source sentence. During decoding the words with conditional probability greater than a threshold θ are considered as the result of lexical choice decoding.

$$BOW_T^* = \{t|P(t|BOgrams(S)) > \theta\} \tag{8}$$

For reconstructing the proper order of words in the target sentence we consider all permutations of words in BOW_T^* and weight them by a target language model. This step is similar to the one described in Section 2.7. The BOW approach can also be modified to allow for length adjustments of target sentences, if we add optional deletions in the final step of permutation decoding. The parameter θ and an additional word deletion penalty can then be used to adjust the length of translated outputs. In Section 6, we discuss several issues regarding this model.

4 Choosing the classifier

This section addresses the choice of the classification technique, and argues that one technique that yields excellent performance while scaling well is *binary maximum entropy (Maxent)* with *L1-regularization*.

4.1 Multiclass vs. Binary Classification

The Sequential and BOW models represent two different classification problems. In the sequential model, we have a *multiclass* problem where each class t_i is exclusive, therefore, all the classifier outputs $P(t_i|\Phi)$ must be jointly optimized such that

Table 1: A comparison of the sequential and bag-of-words lexical choice models

	Sequential Lexical Model	**Bag-of-Words Lexical Model**
Output target	Target word for each source position i	Target word given a source sentence
Input features	$BOgram(S, i-d, i+d)$: bag of n-grams in source sentence in the interval $[i-d, i+d]$	$BOgram(S, 0, \|S\|)$: bag of n-grams in source sentence
Probabilities	$P(t_i \| BOgram(S, i-d, i+d))$ Independence assumption between the labels	$P(BOW(T) \| BOgram(S, 0, \|S\|))$
Number of classes	One per target word or phrase	
Training samples	One per source token	One per sentence
Preprocessing	Source/Target *word* alignment	Source/Target *sentence* alignment

$\sum_i P(t_i|\Phi) = 1$. This can be problematic: with one classifier per word in the vocabulary, even allocating the memory during training may exceed the memory capacity of current computers.

In the BOW model, each class can be detected independently, and two different classes can be detected at the same time. This is known as the 1-vs-other scheme. The key advantage over the multiclass scheme is that not all classifiers have to reside in memory at the same time during training which allows for parallelization. Fortunately for the sequential model, we can decompose a multiclass classification problem into separate 1-vs-other problems. In theory, one has to make an additional independence assumption and the problem statement becomes different. Each output label t is projected into a bit string with components $b_j(t)$ where probability of each component is estimated independently:

$$P(b_j(t)|\Phi) = 1 - P(\bar{b}_j(t)|\Phi) = \frac{1}{1 + e^{-(\lambda_j - \lambda_{\bar{j}}) \cdot \Phi}}$$

In practice, despite the approximation, the 1-vs-other scheme has been shown to perform as well as the multiclass scheme (Rifkin and Klautau, 2004). As a consequence, we use the same type of binary classifier for the sequential and the BOW models.

The excellent results recently obtained with the SEARN algorithm (Daume et al., 2007) also suggest that binary classifiers, when properly trained and combined, seem to be capable of matching more complex *structured* output approaches.

4.2 Geometric vs. Probabilistic Interpretation

We separate the most popular classification techniques into two broad categories:

- Geometric approaches maximize the width of a separation margin between the classes. The most popular method is the Support Vector Machine (SVM) (Vapnik, 1998).

- Probabilistic approaches maximize the conditional likelihood of the output class given the input features. This logistic regression is also called Maxent as it finds the distribution with *maximum entropy* that properly estimates the average of each feature over the training data (Berger et al., 1996).

In previous studies, we found that the best accuracy is achieved with non-linear (or kernel) SVMs, at the expense of a high test time complexity, which is unacceptable for machine translation. Linear SVMs and regularized Maxent yield similar performance. In theory, Maxent training, which scales linearly with the number of examples, is faster than SVM training, which scales quadratically with the number of examples. In our first experiments with lexical choice models, we observed that Maxent slightly outperformed SVMs. Using a single threshold with SVMs, some classes of words were over-detected. This suggests that, as theory predicts, SVMs do not properly approximate the posterior probability. We therefore chose to use Maxent as the best probability approximator.

4.3 L1 vs. L2 regularization

Traditionally, Maxent is regularized by imposing a Gaussian prior on each weight: this L2 regularization finds the solution with the smallest possible weights. However, on tasks like machine translation with a very large number of input features, a Laplacian L1 regularization that also attempts to maximize the number of zero weights is highly desirable.

A new L1-regularized Maxent algorithms was proposed for density estimation (Dudik et al., 2004) and we adapted it to classification. We found this algorithm to converge faster than the current state-of-the-art in Maxent training, which is L2-regularized L-BFGS (Malouf, 2002)[1]. Moreover, the number of trained parameters is considerably smaller.

5 Data and Experiments

We have performed experiments on the IWSLT06 Chinese-English training and development sets from

[1]We used the implementation available at http://homepages.inf.ed.ac.uk/s0450736/maxent_toolkit.html

Table 2: Statistics of training and development data from 2005/2006 (* = first of multiple translations only).

	Training (2005)		Dev 2005		Dev 2006	
	Chinese	English	Chinese	English	Chinese	English
Sentences	46,311		506		489	
Running Words	351,060	376,615	3,826	3,897	5,214	6,362*
Vocabulary	11,178	11,232	931	898	1,136	1,134*
Singletons	4,348	4,866	600	538	619	574*
OOVs [%]	-	-	0.6	0.3	0.9	1.0
ASR WER [%]	-	-	-	-	25.2	-
Perplexity	-	-	33	-	86	-
# References	-	-	16		7	

2005 and 2006. The data are traveler task expressions such as seeking directions, expressions in restaurants and travel reservations. Table 2 presents some statistics on the data sets. It must be noted that while the 2005 development set matches the training data closely, the 2006 development set has been collected separately and shows slightly different statistics for average sentence length, vocabulary size and out-of-vocabulary words. Also the 2006 development set contains no punctuation marks in Chinese, but the corresponding English translations have punctuation marks. We also evaluated our models on the Chinese speech recognition output and we report results using 1-best with a word error rate of 25.2%.

For the experiments, we tokenized the Chinese sentences into character strings and trained the models discussed in the previous sections. Also, we trained a punctuation prediction model using Maxent framework on the Chinese character strings in order to insert punctuation marks into the 2006 development data set. The resulting character string with punctuation marks is used as input to the translation decoder. For the 2005 development set, punctuation insertion was not needed since the Chinese sentences already had the true punctuation marks.

In Table 3 we present the results of the three different translation models – FST, Sequential Maxent and BOW Maxent. There are a few interesting observations that can be made based on these results. First, on the 2005 development set, the sequential Maxent model outperforms the FST model, even though the two models were trained starting from the same GIZA++ alignment. The difference, however, is due to the fact that Maxent models can cope with increased lexical context[2] and the parameters of the model are discriminatively trained. The more surprising result is that the BOW Maxent model significantly outperforms the sequential Maxent model.

[2]We use 6 words to the left and right of a source word for sequential Maxent, but only 2 preceding source and target words for FST approach.

The reason is that the sequential Maxent model relies on the word alignment, which, if erroneous, results in incorrect predictions by the sequential Maxent model. The BOW model does not rely on the word-level alignment and can be interpreted as a discriminatively trained model of dictionary lookup for a target word in the context of a source sentence.

Table 3: Results (mBLEU) scores for the three different models on the transcriptions for development set 2005 and 2006 and ASR 1-best for development set 2006.

	Dev 2005	Dev 2006	
	Text	Text	ASR 1-best
FST	51.8	19.5	16.5
Seq. Maxent	53.5	19.4	16.3
BOW Maxent	59.9	19.3	16.6

As indicated in the data release document, the 2006 development set was collected differently compared to the one from 2005. Due to this mismatch, the performance of the Maxent models are not very different from the FST model, indicating the lack of good generalization across different genres. However, we believe that the Maxent framework allows for incorporation of linguistic features that could potentially help in generalization across genres. For translation of ASR 1-best, we see a systematic degradation of about 3% in mBLEU score compared to translating the transcription.

In order to compensate for the mismatch between the 2005 and 2006 data sets, we computed a 10-fold average mBLEU score by including 90% of the 2006 development set into the training set and using 10% of the 2006 development set for testing, each time. The average mBLEU score across these 10 runs increased to 22.8.

In Figure 6 we show the improvement of mBLEU scores with the increase in permutation window size. We had to limit to a permutation window size of 10 due to memory limitations, even though the curve has not plateaued. We anticipate using pruning techniques we can increase the window size further.

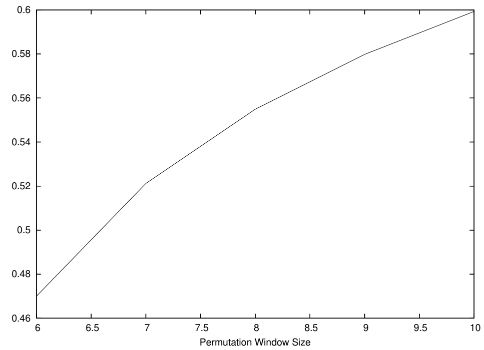

Figure 6: Improvement in mBLEU score with the increase in size of the permutation window

5.1 United Nations and Hansard Corpora

In order to test the scalability of the global lexical selection approach, we also performed lexical selection experiments on the United Nations (Arabic-English) corpus and the Hansard (French-English) corpus using the SFST model and the BOW Maxent model. We used 1,000,000 training sentence pairs and tested on 994 test sentences for the UN corpus. For the Hansard corpus we used the same training and test split as in (Zens and Ney, 2004): 1.4 million training sentence pairs and 5432 test sentences. The vocabulary sizes for the two corpora are mentioned in Table 4. Also in Table 4, are the results in terms of F-measure between the words in the reference sentence and the decoded sentences. We can see that the BOW model outperforms the SFST model on both corpora significantly. This is due to a systematic 10% relative improvement for open class words, as they benefit from a much wider context. BOW performance on close class words is higher for the UN corpus but lower for the Hansard corpus.

Table 4: Lexical Selection results (F-measure) on the Arabic-English UN Corpus and the French-English Hansard Corpus. In parenthesis are F-measures for open and closed class lexical items.

Corpus	Vocabulary		SFST	BOW
	Source	Target		
UN	252,571	53,005	64.6 (60.5/69.1)	69.5 (66.2/72.6)
Hansard	100,270	78,333	57.4 (50.6/67.7)	60.8 (56.5/63.4)

6 Discussion

The BOW approach is promising as it performs reasonably well despite considerable losses in the transfer of information between source and target language. The first and most obvious loss is about word position. The only information we currently use to restore the target word position is the target language model. Information about the grammatical role of a word in the source sentence is completely lost. The language model might fortuitously recover this information if the sentence with the correct grammatical role for the word happens to be the maximum likelihood sentence in the permutation automaton.

We are currently working toward incorporating syntactic information on the target words so as to be able to recover some of the grammatical role information lost in the classification process. In preliminary experiments, we have associated the target lexical items with supertag information (Bangalore and Joshi, 1999). Supertags are labels that provide linear ordering constraints as well as grammatical relation information. Although associating supertags to target words increases the class set for the classifier, we have noticed that the degradation in the F-score is on the order of 3% across different corpora. The supertag information can then be exploited in the sentence construction process. The use of supertags in phrase-based SMT system has been shown to improve results (Hassan et al., 2006).

A less obvious loss is the number of times a word or concept appears in the target sentence. *Function words* like "the" and "of" can appear many times in an English sentence. In the model discussed in this paper, we index each occurrence of the function word with a counter. In order to improve this method, we are currently exploring a technique where the function words serve as attributes (e.g. definiteness, tense, case) on the contentful lexical items, thus enriching the lexical item with morpho-syntactic information.

A third issue concerning the BOW model is the problem of *synonyms* – target words which translate the same source word. Suppose that in the training data, target words t_1 and t_2 are, with equal probability, translations of the same source word. Then, in the presence of this source word, the probability to detect the corresponding target word, which we assume is 0.8, will be, because of discriminant learning, split equally between t_1 and t_2, that is 0.4 and 0.4. Because of this synonym problem, the BOW threshold θ has to be set lower than 0.5, which is observed experimentally. However, if we set the threshold to 0.3, both t_1 and t_2 will be detected in the target sentence, and we found this to be a major source of undesirable insertions.

The BOW approach is different from the parsing based approaches (Melamed, 2004; Zhang and Gildea, 2005; Cowan et al., 2006) where the translation model tightly couples the syntactic and lexical items of the two languages. The decoupling of the two steps in our model has the potential for generating paraphrased sentences not necessarily isomorphic to the structure of the source sentence.

7 Conclusions

We view machine translation as consisting of lexical selection and lexical reordering steps. These two steps need not necessarily be sequential and could be tightly integrated. We have presented the weighted finite-state transducer model of machine translation where lexical choice and a limited amount of lexical reordering are tightly integrated into a single transduction. We have also presented a novel approach to translation where these two steps are loosely coupled and the parameters of the lexical choice model are discriminatively trained using a maximum entropy model. The lexical reordering model in this approach is achieved using a permutation automaton. We have evaluated these two approaches on the 2005 and 2006 IWSLT development sets and shown that the techniques scale well to Hansard and UN corpora.

References

H. Alshawi, S. Bangalore, and S. Douglas. 1998. Automatic acquisition of hierarchical transduction models for machine translation. In *ACL*, Montreal, Canada.

R.E. Banchs, J.M. Crego, A. Gispert, P. Lambert, and J.B. Marino. 2005. Statistical machine translation of euparl data by using bilingual n-grams. In *Workshop on Building and Using Parallel Texts*. ACL.

S. Bangalore and A. K. Joshi. 1999. Supertagging: An approach to almost parsing. *Computational Linguistics*, 25(2).

S. Bangalore and G. Riccardi. 2000. Stochastic finite-state models for spoken language machine translation. In *Proceedings of the Workshop on Embedded Machine Translation Systems*, pages 52–59.

A.L. Berger, Stephen A. D. Pietra, D. Pietra, and J. Vincent. 1996. A Maximum Entropy Approach to Natural Language Processing. *Computational Linguistics*, 22(1):39–71.

P. Brown, S.D. Pietra, V.D. Pietra, and R. Mercer. 1993. The Mathematics of Machine Translation: Parameter Estimation. *Computational Linguistics*, 16(2):263–312.

D. Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the ACL Conference*, Ann Arbor, MI.

B. Cowan, I. Kucerova, and M. Collins. 2006. A discriminative model for tree-to-tree translation. In *Proceedings of EMNLP*.

H. Daume, J. Langford, and D. Marcu. 2007. Search-based structure prediction. *submitted to Machine Learning Journal*.

M. Dudik, S. Phillips, and R.E. Schapire. 2004. Performance Guarantees for Regularized Maximum Entropy Density Estimation. In *Proceedings of COLT'04*, Banff, Canada. Springer Verlag.

V. Goffin, C. Allauzen, E. Bocchieri, D. Hakkani-Tur, A. Ljolje, S. Parthasarathy, M. Rahim, G. Riccardi, and M. Saraclar. 2005. The AT&T WATSON Speech Recognizer. In *Proceedings of ICASSP*, Philadelphia, PA.

H. Hassan, M. Hearne, K. Sima'an, and A. Way. 2006. Syntactic phrase-based statistical machine translation. In *Proceedings of IEEE/ACL first International Workshop on Spoken Language Technology (SLT)*, Aruba, December.

S. Kanthak, D. Vilar, E. Matusov, R. Zens, and H. Ney. 2005. Novel reordering approaches in phrase-based statistical machine translation. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, pages 167–174, Ann Arbor, Michigan.

J. Lafferty, A. McCallum, and F. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML*, San Francisco, CA.

R. Malouf. 2002. A comparison of algorithms for maximum entropy parameter estimation. In *Proceedings of CoNLL-2002*, pages 49–55. Taipei, Taiwan.

I. D. Melamed. 2004. Statistical machine translation by parsing. In *Proceedings of ACL*.

F. J. Och and H. Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of ACL*.

F.J. Och and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Ryan Rifkin and Aldebaro Klautau. 2004. In defense of one-vs-all classification. *Journal of Machine Learning Research*, pages 101–141.

C. Tillmann and T. Zhang. 2006. A discriminative global training algorithm for statistical mt. In *COLING-ACL*.

V.N. Vapnik. 1998. *Statistical Learning Theory*. John Wiley & Sons.

B. Wellington, J. Turian, C. Pike, and D. Melamed. 2006. Scalable purely-discriminative training for word and tree transducers. In *AMTA*.

D. Wu. 1997. Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Corpora. *Computational Linguistics*, 23(3):377–404.

K. Yamada and K. Knight. 2001. A syntax-based statistical translation model. In *Proceedings of* 39^{th} *ACL*.

R. Zens and H. Ney. 2004. Improvements in phrase-based statistical machine translation. In *Proceedings of HLT-NAACL*, pages 257–264, Boston, MA.

H. Zhang and D. Gildea. 2005. Stochastic lexicalized inversion transduction grammar for alignment. In *Proceedings of ACL*.