Machine Learning, 60, 135–158, 2005

Maximum Entropy Modeling: A Suitable Framework to Learn Context-Dependent Lexicon Models for Statistical Machine Translation*

ISMAEL GARCÍA-VAREA ivarea@info-ab.uclm.es
Dpto. de Informática, Univ. de Castilla-La Mancha, Campus Universitario s/n, 02071 Albacete, Spain

FRANCISCO CASACUBERTA fcn@dsic.upv.es
Dpto. de Sistemas Informáticos y Computación, Instituto Tecnológico de Informática, Univ. Politécnica de Valencia, Camino de Vera, s/n, 46071Valencia,Spain

Editors: Dan Roth and Pascale Fung

Abstract. Current statistical machine translation systems are mainly based on statistical word lexicons. However, these models are usually context-independent, therefore, the disambiguation of the translation of a source word must be carried out using other probabilistic distributions (distortion distributions and statistical language models). One efficient way to add contextual information to the statistical lexicons is based on maximum entropy modeling. In that framework, the context is introduced through feature functions that allow us to automatically learn context-dependent lexicon models.

In a first approach, maximum entropy modeling is carried out after a process of learning standard statistical models (alignment and lexicon). In a second approach, the maximum entropy modeling is integrated in the expectation-maximization process of learning standard statistical models.

Experimental results were obtained for two well-known tasks, the French–English Canadian Parliament HANSARDS task and the German–English VERBMOBIL task. These results proved that the use of maximum entropy models in both approaches, can help to improve the performance of the statistical translation systems.

Keywords: statistical machine translation, maximum entropy modeling, context-dependent lexicon models

Abbreviations: ME: Maximum Entropy, SMT: Statistical Machine Translation, EM: Expectation–Maximization.

1. Introduction

The performance of a statistical machine translation system depends directly on the quality of the lexicon, the alignment and the target language models used. Most of the statistical machine translation systems are based on single-word alignment models, as described in Brown et al. (1993), which are trained by using existing parallel corpora. Typically, the

*This work has been partially supported by the European Union under grant IST-2001-32091 and by the Spanish CICYT under project TIC-2003-08681-C02-02. The experiments on the VERBMOBIL task were done when the first author was a visiting scientist at RWTH Aachen-Germany.

lexicon models used in these systems do not include any linguistic or contextual information, which often yields inadequate alignments in pairs of sentences. Those lexicon models lack context information that can be extracted from the same parallel corpus. This additional information could be:

– Simple context information: Information of the words surrounding a word pair.
– Syntactic information: Part-of-speech (POS) information, syntactic constituent, sentence mood.
– Semantic information: disambiguation information (e.g. from WordNet), current/previous speech or dialog act, etc.

To include this additional information within the statistical framework, we use the maximum entropy approach. Other alternatives are based on extended lexicon models such as phrase-based models (Tomás & Casacuberta,2001, 2003).

The ME approach has been applied in natural language processing and machine translation to a variety of tasks during the last few years. In Berger, Della Pietra, and Della Pietra (1996), this approach is applied to the so-called IBM Candide system to build context-dependent models, to compute automatic sentence splitting and to improve word reordering in translation. Similar techniques are used in Papineni et al. (1998), and Foster (2000b) for so-called direct translation models instead of those proposed in Brown et al. (1993). In Foster (2000a), two methods for incorporating information about the relative position of bilingual word pairs into a ME translation model are described. In Och and Ney (2002), a general framework for SMT based on direct ME models is presented, where the different components of a SMT system (i.e. lexicon, alignment/distortion, and fertility models) are treated as feature functions, and can be combined in different ways. Other authors have also applied this approach to language modeling (Rosenfeld, 1996; Ratnaparkhi, 1997; Martin, Ney, & Zaplo, 1999; Peters & Klakow, 1999; Khudanpur & Wu, 1999, 2000), to natural language understanding (Bender et al., 2003), to natural language parsing (Charniak, 1999; Ratnaparkhi, 1999), and to POS tagging (Ratnaparkhi, 1996).

In this paper, we define a set of context-dependent ME lexicon models. We show how to learn these models in order to generally improve existing statistical machine translation systems in a way similar to the one in García-Varea et al. (2001) where ME models are used in a completely decoupled way to reduce translation test perplexities and translation errors by means of a rescoring algorithm. On the other hand, as in García-Varea et al. (2002), we present a procedure for an efficient training of these ME models within the conventional EM training to the family of statistical alignment models proposed in Brown et al. (1993). In each iteration of the training process, the set of ME models is automatically generated by using the set of possible word-alignments between each pair of sentences. The ME models are trained with the Generalized Iterative Scaling (GIS) algorithm (Darroch & Ratcliff, 1972) and then used in the next iteration of the EM training process in order to recompute a new set of parameters of the translation models. In contrast to Och and Ney (2002), where the ME approach is used to replace source-channel paradigm while preserving the translation probability component, here ME is used to enhance the translation probability component itself.

The paper is structured as follows: in Section 2, a short overview of a statistical machine translation systems is presented; next, in Section 3, the ME principle is stated as well as all the issues concerning the definition of context-dependent ME models. In Section 4, a first experimentation is presented in order to set some parameters to be used in future experiments. In Section 5, we go further and we present how the ME models can be completely integrated into a conventional EM training of statistical alignment models, paying special attention to the effectiveness of the integration and also presenting some alignment quality experiments. Finally, the main conclusions of this work and future direction plan are presented.

Experimental results are given for two well-known tasks the French–English Canadian Parliament HANSARDS task and the German–English VERBMOBIL task. The experimentation presented demonstrates that the use of ME models in the framework of single-word based statistical alignment models can help to improve the performance of the system.

2. Statistical machine translation

The goal of the translation process in statistical machine translation can be formulated as follows: A source language string $\mathbf{f} = f_1^J = f_1 \dots f_J$ is to be translated into a target language string $\mathbf{e} = e_1^I = e_1 \dots e_I$. Every target string is regarded as a possible translation for the source language string with maximum a-posteriori probability $\Pr(\mathbf{e} \mid \mathbf{f})$. According to Bayes' decision rule, we have to choose the target string that maximizes the product of both the target language model $\Pr(\mathbf{e})$ and the string translation model $\Pr(\mathbf{f} \mid \mathbf{e})$.

Many existing systems for statistical machine translation (Berger et al., 1994; Wang & Waibel 1997, Tillmann et al., 1997, Nießen et al., 1998) use a special way of structuring the string translation model like the one proposed by Brown et al. (1993): the correspondence between the words in the source and the target string is described by alignments that assign one target word position to each source word position. The lexicon probability $p(f \mid e)$ of a certain target word e to occur in the target string is assumed to depend basically only on the source word f aligned to it. These alignment models are similar to the concept of hidden Markov models (HMM) in speech recognition. The alignment mapping is $j \to i = a_j$ from source position j to target position $i = a_j$. The alignment $\mathbf{a} = a_1^J$ may contain alignments $a_j = 0$ with the 'empty' word e_0 to account for source words that are not aligned to any target word. In statistical alignment models, $\Pr(\mathbf{f}, \mathbf{a} \mid \mathbf{e})$, the alignment $\mathbf{a}$ is introduced as a hidden variable.

The translation probability $\Pr(\mathbf{f}, \mathbf{a} \mid \mathbf{e})$ can be rewritten as follows:

$$\begin{aligned}\Pr(\mathbf{f}, \mathbf{a} \mid \mathbf{e}) &= \prod_{j=1}^{J} \Pr\left(f_j, a_j \middle| f_1^{j-1}, a_1^{j-1}, e_1^I\right) \\ &= \prod_{j=1}^{J} \Pr\left(a_j \middle| f_1^{j-1}, a_1^{j-1}, e_1^I\right) \cdot \Pr\left(f_j \middle| f_1^{j-1}, a_1^{j}, e_1^I\right)\end{aligned}$$

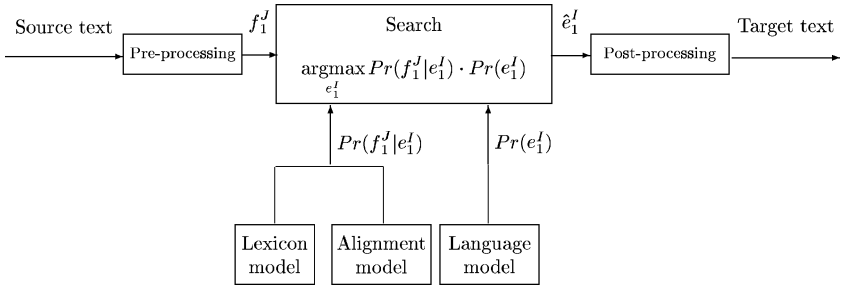

Figure 1. Architecture of the translation approach based on Bayes' decision rule.

Typically, the search is performed using the so-called maximum approximation:

$$\begin{aligned}\hat{e}_1^I &= \arg\max_{e_1^I} \left\{ \Pr(e_1^I) \cdot \sum_{a_1^J} \Pr(f_1^J, a_1^J | e_1^I) \right\} \\ &\approx \arg\max_{e_1^I} \left\{ \Pr(e_1^I) \cdot \max_{a_1^J} Pr(f_1^J, a_1^J | e_1^I) \right\} \end{aligned} \tag{2}$$

The search space consists of the set of all possible target language strings e_1^I and all possible alignments a_1^J.

The overall architecture of the statistical translation approach based on Bayes' decision rule is depicted in Figure (1).

3. Maximum entropy modeling

3.1. Motivation

Typically, the probability $\Pr(f_j \mid f_1^{j-1}, a_1^j, e_1^I)$ in Eq. (1) is approximated by a lexicon model $p(f_j \mid e_{a_j})$ by dropping the dependencies on f_1^{j-1}, a_1^{j-1}, $e_1^{a_{j-1}}$, and $e_{a_{j+1}}^I$. Obviously, this simplification is not true for many natural language phenomena. On the other hand, we could think that the random process that generates the word f given e not only depends on e but also on some contextual information where the pair (f, e) appears. The straightforward approach to include more dependencies in the lexicon model would be to add additional dependencies (e.g. $p(f_j|e_{a_j}, e_{a_{j-1}})$). This approach would yield a significant data sparseness problem. For this reason, we define a set of context-dependent ME lexicon models, which easily allow us to take into account relevant contextual information. Also, as we will see later, these ME models can be directly integrated into a conventional EM training of the statistical alignment models.

In this case, the role of ME is to build a stochastic model that efficiently takes a larger context into account. In the remainder of the paper, we shall use $p_e(f \mid x)$ to denote the probability that the ME model (which is associated to e) assigns to f in the context x.

3.2. *Maximum entropy principle*

In the ME approach, we denote all properties that we deem to be useful by so-called feature functions $\phi_{e,k}(x, f), k = 1, \ldots, K_e$. For example, let us suppose that the k-th feature for word e tries to model the existence or absence of a specific word e'_k in the context of an English word e, which can be translated by f'_k. We can express this dependence using the following feature function:

$$\phi_{e,k}(x, f) = \begin{cases} 1 & \text{if } f = f'_k \text{ and } e'_k \in x \\ 0 & \text{otherwise} \end{cases}. \tag{3}$$

Consequently the k-th feature for word e has the pair (f'_k, e'_k) associated to it.

The ME principle suggests that the optimal parametric form of a model $p_e(f \mid x)$ taking into account the feature functions $\phi_{e,k}, k = 1, \ldots, K_e$ is given by (Berger, Della Pietra, & Della Pietra, 1996):

$$p_e(f \mid x) = \frac{1}{Z_{\Lambda_e}(x)} \exp\left(\sum_{k=1}^{K_e} \lambda_{e,k}\phi_{e,k}(x, f)\right). \tag{4}$$

Here, $Z_{\Lambda_e}(x)$ is a normalization factor. The resulting model has an exponential form with free parameters $\Lambda_e \equiv \{\lambda_{e,k}, k = 1, \ldots, K_e\}$ (Berger, Della Pietra & Della Pietra, 1996). The parameter values that maximize the likelihood for a given training corpus can be computed using the so-called GIS algorithm or its improved version IIS (Della Pietra, Della Pietra, & Lafferty, 1997).

It is important to stress that, in principle, we obtain one ME model for each target language word e. To avoid data sparseness problems for rarely seen words, we use only words that have been seen a certain number of times.

3.3. *Contextual information and training events*

In order to train the ME model $p_e(f \mid x)$ associated to a target word e, we need to construct a corresponding training sample from the entire bilingual corpus, depending on the contextual information that we want to use. To construct this sample, we need to know the word-to-word alignment between each sentence pair in the corpus. That is obtained using the Viterbi alignment provided by a translation model as described in Brown et al. (1993). Specifically, we use the Viterbi alignment that was produced by IBM Model 5. To obtain the Viterbi alignment, we use the program GIZA++ (Och, 2000), which is an extension of the training program available in EGYPT (Al-Onaizan et al., 1999).

Once we know the alignment of each target word, we need to define the contextual information to be extracted. As in Berger, Della Pietra, and Della Pietra (1996), we use a window of 3 words to the left and 3 words to the right of the target word as contextual information. As in García-Varea et al. (2001), in addition to a dependence on the words themselves, we also use a dependence on the word classes. Thereby, we improve the generalization of the models and include some semantic and syntactic information; we also

use a dependence on the source context. More specifically, we use the following contextual information:

– Target context: As in Berger, Della Pietra, and Della Pietra (1996), we take into account a window of 3 words to the left and to the right of the target word considered. With these contexts, we try to catch some linguistic phenomena like the article-noun pairs or the interchange adjective-noun to noun-adjective between target and source.
– Source context: In addition, we consider a window of 3 words to the left of the source word f which is connected to e according to the Viterbi alignment. Obviously, the word f in question is considered, too. With this information we can take into account the information that the language model provides us, like for example bistant n-grams or the well-known trigger pairs (Tillmann & Ney, 1996, 1997; Zhou & Lua, 1998).
– Word classes: Instead of using a dependency on the word identity, we also include a dependency on word classes. By doing this, we improve the generalization of the models and include some semantic and syntactic information. The word classes are computed automatically using another statistical training procedure (Och, 1999), which often produces word classes including words with the same semantic meaning in the same class.

A training event, namely (f, e, x), for a specific target word e, is composed by three items:

– The specific word e.
– The source word f aligned to e.
– The context x in which the aligned pair (f,e) appears.

The number of occurrences of the event in the training corpus is also considered as a part of the training events. We will refers to this number as a *count* of the training event.

3.4. Feature definition and selection

Table 1 summarizes the feature functions that we use for a specific pair of aligned words (f_j, e_i): Category 1 features give rise to constraints that enforce equality between the probability of any source translation f_j of e_i according to the model and the probability of that translation in the empirical distribution. A ME model that uses only category 1 features predicts each source translation f_j with the probability $\tilde{p}_e(f_j)$ determined by the empirical data. This is exactly the distribution employed in the translation model described in Brown et al. (1993). Categories 2 and 3 describe features that also depend on an additional word e' that appears one position to the left or to the right of e_i, respectively. The features of category 4 and 5 depend on an additional target word e' that appears in any position of the context x. In categories 6 and 7 the source context is used instead of the target context. Analogous features are defined using the word class associated to each word instead of the word identity. In our experiments, 50 non-ambiguous word classes were used for each language. A more intuitive idea about these categories is shown in Table 1.

Table 1. Meaning of different feature categories where □ represents a specific target word (to be placed in •) and ◇ represents a specific source word, where k has the pair(◇, □) associated to it.

Category	$\phi_{e_i,k}(x, f_j) = 1$ if and only if ...
1	$f_j = \Diamond$
2	$f_j = \Diamond$ and $\square \in$			•	e_i			
3	$f_j = \Diamond$ and $\square \in$				e_i	•		
4	$f_j = \Diamond$ and $\square \in$	•	•	•	e_i			
5	$f_j = \Diamond$ and $\square \in$				e_i	•	•	•
6	$f_j = \Diamond$ and $\Diamond \in$			•	f_j			
7	$f_j = \Diamond$ and $\Diamond \in$	•	•	•	f_j			

The number of possible features that can be used according to the source and target vocabularies and word classes is huge. In other words, the size of the source vocabulary for features of Category 1 and the size of the source vocabulary times the size of the target vocabulary for the rest of the feature categories. In order to reduce the number of features, we perform a threshold-based feature selection. Any feature that occurs less than T times is not used. The aim of the feature selection is two-fold. First, we obtain smaller models by using fewer features. Secondly, we hope to avoid overfitting on the training data.

4. Experimental study

At this point, we are able to use the ME models defined in the previous section in the framework of a statistical machine translation system.

The first question we should answer, taking in to account the definition of the ME context-dependent lexicon models in the previous section, is how these ME models could help to improve existing SMT systems?

In this section, we try to answer this question by means of two different experiments:

– The first one is to carry out a comparison of different models with and without the use of the ME models. We do this by performing a model perplexity experiment.
– The second one is a translation experiment by using the ME lexicon models as a post-process to the output of a SMT system by means of a hypothesis rescoring algorithm.

In addition, these experiments will give us an idea of what kind of contextual information is useful and help us to establish a reasonable value of the threshold T for future experiments.

Table 2. Corpus characteristics for the VERBMOBIL task.

		German	English
Training	Sentences	50,000	
	Running Words	454,619	482,350
	Vocabulary	7,456	4,420
Validation	Sentences	8,073	
	Running Words	64,904	67,571
	Vocabulary	2,579	1,666
Test	Sentences	147	
	Running Words	1,968	2,173
	PP (trigr. LM)	(40.3)	28.8

As commented above, the starting point to make use of the ME models within the framework of a statistical machine translation system is to have an aligned parallel corpus to make it possible to consider all aspects concerning the training of the specific ME models. Thus, we use the IBM Model 5 Viterbi alignment provided by the program GIZA++ (Och, 2000).

For the experiments shown in this section, we used the well-known VERBMOBIL task (Wahlster, 2000). The VERBMOBIL task is a German-English speech translation task in the domain of appointment scheduling, travel planning, and hotel reservation. The task is difficult because it consists of spontaneous speech and the syntactic structures of the sentences are less restricted and highly variable. The training, validation and test corpora characteristics for this task are described in Table 2.

4.1. A model perplexity experiment

In order to obtain the threshold T, we compared the training and validation corpora perplexities for various thresholds. The different thresholds used in the experiments ranged from 0 to 512. The threshold is used as a cut-off for the number of occurrences that a specific feature must appear. So a cut-off of 0 means that all features observed in the training data are used. A cut-off of 32 means those features that appear 32 times or more are considered to train the maximum entropy models.

Another thing to take into account is the fact that not all words in the vocabulary can provide valuable information to the models. Therefore, for training the ME models, we selected the English words that appear at least 150 times in the training sample. The number of such words was 348 (in total) of the 4,673 words contained in the English vocabulary. Table 3 shows the different number of features considered for the 348 English words selected using different thresholds and contexts.

In choosing a reasonable threshold, we have to balance the number of features and the observed perplexity.

Table 3. Number of features used according to different cut-off thresholds. The number of features used when only the English context is considered is shown in the second column of the table. The third column corresponds to English, German and Word-Class contexts.

T	English	English + German
0	846,121	1,581,529
2	240,053	500,285
4	153,225	330,077
8	96,983	210,795
16	61,329	131,323
32	40,441	80,769
64	28,147	49,509
128	21,469	31,8050
256	18,511	22,947
512	17,193	19,027

Table 4. Training and Validation perplexities using different contextual information and different thresholds T. The reference perplexities obtained with the basic translation IBM Model 5 are TrainPP = 10.38 and ValiPP = 13.22.

	English		English + German	
T	TrainPP	ValiPP	TrainPP	ValiPP
0	5.03	11.39	4.60	9.28
2	6.59	10.37	5.70	8.94
4	7.09	10.28	6.17	8.92
8	7.50	10.39	6.63	9.03
16	7.95	10.64	7.07	9.30
32	8.38	11.04	7.55	9.73
64	9.68	11.56	8.05	10.26
128	9.31	12.09	8.61	10.94
256	9.70	12.62	9.20	11.80
512	10.07	13.12	9.69	12.45

The training and validation perplexities are shown in Table 4, where English stands for only English context and English + German stands for English, German and Word-Class contexts.

As expected, the perplexity reduction in the validation corpus was lower than in the training corpus; however in both cases, better perplexities were obtained using the ME models. The best value was obtained when a threshold of 4 was used.

We expected to observe strong overfitting effects when a cut-off for features was too small got used. Yet, for most words, the best validation corpus perplexity was observed when we used all features including those that occurred only once.

4.2. A translation experiment

In order to make use of the ME models in a completely decoupled way within a statistical translation system, we implemented a *rescoring algorithm*. This algorithm takes the standard lexicon model (not using maximum entropy) and the 348 models obtained with the ME training as input. For a hypothesis sentence e_1^I and a corresponding alignment a_1^J, the algorithm modifies the probability/score $\Pr(f_1^J, a_1^J \mid e_1^I)$ according to the refined maximum entropy lexicon model. In other words, the algorithm divides this by the probability associated to the conventional translation model ($p(f \mid e)$) and multiplies that by the probability computed with the ME models ($p_e(f \mid x)$) for each word pair (f, e) which is aligned.

We carried out some experiments with a N-best list of hypotheses provided by a translation system (Tillmann & Ney, 2000) in order to make a rescoring of each i-th hypothesis and reorder the list according to the new probability/score computed with the refined lexicon models. Unfortunately, the N-best extraction algorithm is sub-optimal, i.e. the true best N translations were not extracted. We had to use a limit of only 10 translations per sentence.

For the evaluation of the translation quality, we used the automatically computable Word Error Rate (WER). The WER corresponds to the edit distance between the produced translation and one predefined reference translation. A shortcoming of the WER is the fact that it requires a perfect word order. This is particularly a problem for the VERBMOBIL task, where the word order of the German-English sentence pair can be quite different. As a result, the word order of the automatically generated target sentence can be different from that of the target sentence but still be acceptable. Therefore, the WER measure alone can be misleading. In order to overcome this problem, we introduced an additional measure, the position-independent word error rate (PER). This measure compares the words in the two sentences *without* taking the word order into account. Words that have no matching counterparts are counted as substitution errors. Depending on whether the translated sentence is longer or shorter than the target translation, the remaining words result in either insertion or deletion errors in addition to substitution errors. The PER is guaranteed to be less than or equal to the WER.

We used the top-10 list of hypotheses provided by the translation system described in Tillmann and Ney (2000), which is a dynamic programming-like search algorithm based on the IBM Model 4 statistical alignment model. These 10-best hypotheses were rescored using the ME models and sorted according to the new maximum entropy probability/score. The translation results in terms of error rates are shown in Table 5.

We observed that the translation quality improves slightly with respect to the WER and PER. Table 5 shows that the translation quality results are quite small compared to the improvements in perplexity. We tested these results using the Wilcox statistical hypothesis test and we can not conclude that these results have statistical significance. However, they do show a consistent tendency towards improvement. We attribute this to the fact that the algorithm used to compute the N-best lists is suboptimal. However, we consider that this

Table 5 Translation results for the VERBMOBIL Test-147 for different contextual information and different thresholds using the top-10 translations. The baseline translation results for IBM Model 4 are WER=54.80 and PER=43.07.

	English		English + German	
T	WER	PER	WER	PER
0	54.57	42.98	54.02	42.48
2	54.16	42.43	54.07	42.71
4	54.53	42.71	54.11	42.75
8	54.76	43.21	54.39	43.07
16	54.76	43.53	54.02	42.75
32	54.80	43.12	54.53	42.94
64	54.21	42.89	54.53	42.89
128	54.57	42.98	54.67	43.12
256	54.99	43.12	54.57	42.89
512	55.08	43.30	54.85	43.21

Table 6 Four examples showing the translation obtained with the IBM Model 4 (M4) and the ME model for a given German source sentence. SRC stands for source sentence, in that case German.

SRC:	Danach wollten wir eigentlich noch Abendessen gehen.
M4:	We actually concluding dinner together.
ME:	Afterwards we wanted to go to dinner.
SRC:	Bei mir oder bei Ihnen?
M4:	For me or for you?
ME:	At your or my place?
SRC:	Das wäre genau das richtige.
M4:	That is exactly it spirit.
ME:	That is the right thing.
SRC:	Ja, das sieht bei mir eigentlich im Januar ziemlich gut aus.
M4:	Yes, that does not suit me in January looks pretty good.
ME:	Yes, that looks pretty good for me actually in January.

experiment shows that ME models could improve translation quality if optimal and larger *N*-best lists or word translation graphs were used, or if the ME models were used in a completely integrated system instead of using the rescoring algorithm.

Table 6 shows some examples where the translation obtained with the rescoring procedure was better than the best hypothesis provided by the translation system.

5. Integration of the ME lexicon models within the training of statistical alignment models

As discussed in previous section, to make use of the ME models, we need to know the word-by-word alignment of a parallel corpus for one main reason: to be able to extract the contextual training events to carry out the training of the refined ME lexicon models. However, why not directly use the already proven more valuable contextual information provided by ME models to obtain better statistical alignment models?

In this section, we want to go further; that is, to show how the training of the ME models can be directly integrated within the Expectation–Maximization (EM) training of conventional statistical alignment models. Other works that combine iterative scaling and EM techniques for ME estimation have been carried out, like (Lauritzen, 1995) for graphical association models and (Riezler et al., 2000) for constrained-based grammars. Also, in Wang, Schwurmans, and Zhao (2004) the maximum entropy principle with latent variables is presented.

In the remainder of the section, first, as an introduction, we show how a conventional EM training of statistical alignment models is performed. Then, we show how the EM-ME training integration can be done and how to carry it out in an efficient way. We also present experimental results in order to show the alignment quality obtained for the different approaches.

5.1. Conventional EM training (review)

In this section, we describe the training of the model parameters. Every model has a specific set of free parameters. For example, the parameters θ for Model 4 (Brown et al. 1993) consist of lexicon, alignment and fertility parameters:

$$\theta = \Big\{ \{p(f \mid e)\}, \{p_{=1}(\Delta j)\}, \{p_{>1}(\Delta j)\}, \{p(\phi \mid e)\}, p_1 \Big\}. \tag{5}$$

To train the model parameters θ, we pursue a maximum likelihood approach using a parallel training corpus consisting of S sentence pairs $\{(\mathbf{f}_s, \mathbf{e}_s) : s = 1, \ldots, S\}$:

$$\hat{\theta} = \arg\max_{\theta} \prod_{s=1}^{S} \sum_{\mathbf{a}} p_\theta(\mathbf{f}_s, \mathbf{a} \mid \mathbf{e}_s). \tag{6}$$

We do this by applying the EM algorithm. The different models are trained in succession on the same data, where the final parameter values of a simpler model serve as the starting point for a more complex model.

In the E-step, the lexicon parameter counts for one sentence pair $(\mathbf{f}, \mathbf{e})$ are calculated:

$$c(f \mid e; \mathbf{f}, \mathbf{e}) = N(\mathbf{f}, \mathbf{e}) \cdot \sum_{\mathbf{a}} \Pr(\mathbf{a}|\mathbf{f}, \mathbf{e}) \sum_{j} \delta(f, f_j)\delta(e, e_{a_j}). \tag{7}$$

Here, $N(\mathbf{f}, \mathbf{e})$ is the training corpus count of the sentence pair $(\mathbf{f}, \mathbf{e})$. In the M-step, we want to compute the lexicon parameters $p(f \mid e)$ that maximize the likelihood on the training corpus. This results in the following re-estimation (Brown et al., 1993):

$$p(f \mid e) = \frac{\sum_s c(f \mid e; \mathbf{f}^{(s)}, \mathbf{e}^{(s)})}{\sum_{s,f} c(f \mid e; \mathbf{f}^{(s)}, \mathbf{e}^{(s)})} \quad . \tag{8}$$

Similarly, the alignment and fertility probabilities can be estimated for all other alignment models (see Brown et al. (1993) for details). When bootstrapping from a simpler model to a more complex model, the simpler model is used to weight the alignments and the counts are accumulated for the parameters of the more complex model.

5.2. EM-ME training integration

Using a ME lexicon model for a target word e, we have to train the model parameters $\Lambda_e \equiv \{\lambda_{e,k} : k = 1, \ldots, K_e\}$ instead of the parameters $\{p(f \mid e)\}$. We pursue the following approach. In the E-step, we perform a refined count collection for the lexicon parameters:

$$c(f \mid e, x; \mathbf{f}, \mathbf{e}) = N(\mathbf{f}, \mathbf{e}) \cdot \sum_{\mathbf{a}} Pr(\mathbf{a} \mid \mathbf{f}, \mathbf{e}) \sum_j \delta(f, f_j)\delta(e, e_{a_j})\delta(x, x_{j,a_j}) \quad . \tag{9}$$

Here, x_{j,a_j} should denote the ME context that surrounds f_j and e_{a_j}. In the M-step, we want to compute the lexicon parameters that maximize the likelihood:

$$\hat{\Lambda}_e = \arg\max_{\Lambda_e} \prod_{f,x} c(f \mid e, x; \mathbf{f}, \mathbf{e}) \cdot \log p_e(f \mid x) \quad . \tag{10}$$

Hence, the refined lexicon counts $c(f \mid e, x; \mathbf{f}, \mathbf{e})$ are the weights of the set of training samples (f, e, x), which are used to train the ME models.

The re-estimation of the alignment and fertility probabilities does not change if we use a ME lexicon model.

Thus, we obtain the following steps of each iteration for the EM algorithm:

1. E-step:
   (a) Collect counts for alignment and fertility parameters.
   (b) Collect refined lexicon counts and ME training events generation.

2. M-step:
   (a) Re-estimate alignment and fertility parameters.
   (b) Perform GIS training for ME lexicon parameters.

With respect to a conventional EM loop, steps 1b and 2b involve overhead in space and computation time. In the next subsection, we outline how to address this overhead to make the integrated training as efficient as possible.

5.3. *Efficient training*

As an introduction to the integration of the ME training within the EM algorithm, let us suppose that we are in the k-th iteration of the EM process. In the E-step of this iteration, for every sentence pair in the training corpus (f_1^J, e_1^I) and every possible alignment between them, we need to use the ME lexicon probability of every word pair (f_j, e_i) computed in the $(k-1)$-th iteration. We will then need a priori to recompute $p_{e_i}(f_j \mid x)$ for every computation of $\Pr(\mathbf{f}, \mathbf{a} \mid \mathbf{e})$ (and also J times for the J words). To perform this computation efficiently, we precompute all possible $p_{e_i}(f_j \mid x)$ in a translation matrix $(I \times J)$ of ME lexicon probabilities. Thus, each time we need to compute this probability, we only need to access the corresponding matrix element. In the E-step, the specific ME training events (f_j, e_i, x) generation of current iteration is also performed.

After the E-step is carried out for each sentence pair in the corpus, we have all possible ME events (f, e, x) for each word e. Then, with these events, in the M-step we perform a GIS training for every e word we considered (a priori) relevant to our problem and obtain the set of Λ_e parameters that define our specific ME model.

The following factors specially contribute to the computational overhead introduced by the ME lexicon models:

1. The computation time of the translation matrix. This involves an increase of one order of magnitude.
2. The computation of the GIS training for each word e to be modeled by ME; in the worst case, when all words e from the target vocabulary are used. In the experiments, the computation time of the GIS algorithm ranges from 5 to 10 seconds on average. This could yield an increase of two orders of magnitude depending on the number of ME models to be considered.

Hence, the additional time consumption directly depends on the number of words e to be modeled by ME. As described at the end of Section 3.4, we developed a ME model for every word that appeared in the training corpus more than a fixed number of times. In our experiments, this word selection yielded only about 5–10% of the vocabulary.

With respect to the space overhead, we will need to store every possible ME training event (f, e, x), that is, every possible combination of e from the target vocabulary, f from the source vocabulary and context x. Obviously this requires a huge quantity of memory, as the word selection described above also plays an important role in the efficiency of space overhead.

The number of training events is an important factor for the computational overhead of the GIS algorithm. Therefore, a pruning of these training events is also applied. As described in the previous subsection, the refined lexicon count (*fractional counts* (Brown et al., 1993)) are the weights of the ME training events. We prune those training events with fractional counts smaller than 1.0 in order to avoid very rare events. Very rare events are thereby discarded and the ME training is faster, and a better parameter estimation is performed. In addition, the space overhead is also reduced.

In the following, we suggest a simplified approach which reduces the overhead required by this approach. First, we perform a regular training of the EM algorithm. Then, after the

Table 7 Time consumption in seconds of different approaches per EM iteration (on average for the five IBM models). # of e means the number of target words to be modeled by ME after the counting-based word selection.

Task	Size of train.	# of e	Conv. train	ME-train	Simp. ME-train
VERBMOBIL	0.5 K	29	1	29	1.5
	8 K	84	18	235	68
	35 K	209	60	2,290	675
HANSARDS	0.5 K	15	2.5	29	3
	8 K	80	35	1,180	100
	128 K	1,214	655	16,890	6,870

final iteration, we perform the ME training of the ME lexicon parameters but use only the Viterbi alignment of each sentence pair instead of the set of all possible alignments. Finally, a new EM training is performed where the lexicon parameters are fixed to the ME lexicon models obtained previously. In this case, the more informative contextual information is also used but in a decoupled way. It is important to stress that in this approximation only one ME training is needed. Interestingly, the alignment quality obtained with this simplification and the fully integrated approach are practically the same.

The time consumption in seconds per iteration of the EM algorithm of the different approaches are shown in Table 7. This experiment was carried out on a Pentium-III machine at 600 MHz. The results are presented for the corpora described in Table 8, regarding the EM training of the experiments presented in Section 5.4.

5.4. Alignment quality experimental results

We present results on the VERBMOBIL task and the HANSARDS task. The French–English HANSARDS task consists of debates in the Canadian Parliament. This task has a very large vocabulary of more than 100,000 French words.

The corpus statistics for these experiments are shown in Table 8. The number of running words and the vocabularies are based on full-form words including the punctuation marks. For these experiments we used a subset of the complete corpus of the VERBMOBIL corpus because of the availability of the annotation scheme (described below) and in order to make the results comparable to previously related work as in Och and Ney (2003).

We produced smaller training corpora by randomly choosing 500, 8,000 and 34,000 sentences from the VERBMOBIL task and 500, 8,000 and 128,000 sentences from the HANSARDS task.

To train the context-dependent statistical alignment models, we extended the publicly available toolkit GIZA++ (Och, 2000)(which will be also available shortly). The training of the ME models was carried out using the YASMET toolkit (Och, 2001). For the experiments shown below we considered a value of T to the optimal value obtained in the previous

Table 8 Corpus characteristics for alignment quality experiments.

		VERBMOBIL		HANSARDS	
		German	English	French	English
Training	Sentences	34,446		128,000	
	Running Words	329,625	343,076	2,120,212	11,309,283
	Vocabulary	5,936	3,505	37,532	29,414

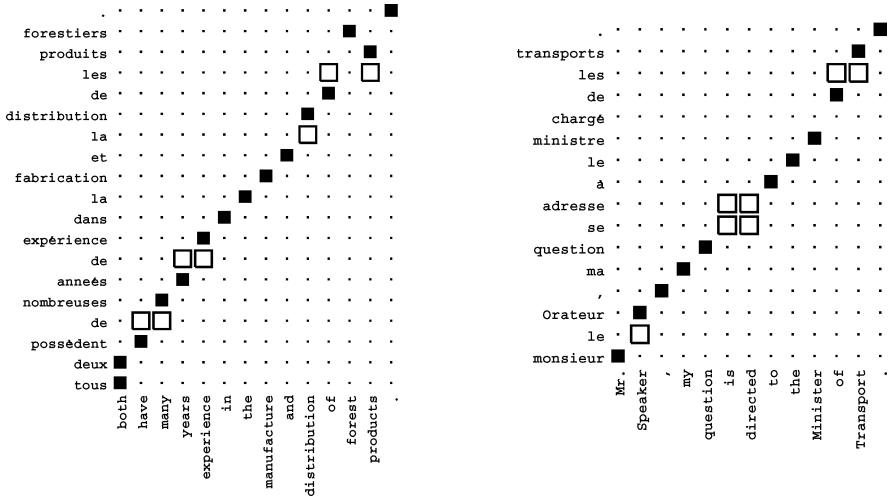

Figure 2. Two examples of manual alignments with *S(ure)* (■) and *P(ossible)* (□) connections

experiments, that is a value of 4. Also, only target context (that is English context) plus word-classes were used as contextual information for the ME models.

5.4.1. Evaluation methodology. We use the same annotation scheme for single-word based alignments and a corresponding evaluation criterion as described in Och and Ney (2000). The annotation scheme explicitly allows for ambiguous alignments. The people performing the annotation were allowed to specify two different kinds of alignments: a $S(ure)$ alignment, which is used for alignments that are certain or unambiguous and a $P(ossible)$ alignment, which is used for ambiguous or doubtful alignments. The P label is used particularly to align words within idiomatic expressions, free translations, and missing function words ($S \subseteq P$).

The reference alignment thus obtained may contain many-to-one and one-to-many relationships. Figure 2 shows two examples (of the HANSARDS task) of manually aligned sentences with S and P labels.

The quality of an alignment $A = \{(j, a_j) \mid a_j > 0\}$ is then computed by appropriately redefined precision and recall measures and the alignment error rate, which is derived from the well known F-measure:

Table 9 AER [%] on Hansards task.

		Size of training corpus		
Training scheme	Model	0.5 K	8 K	128 K
1^5	IBM1	48.0	35.1	29.2
	IBM1 + ME	47.7	32.7	22.5
$1^5 2^5$	IBM2	46.0	29.2	21.9
	IBM2 + ME	44.7	28.0	19.0
$1^5 2^5 3^3$	IBM3	43.2	27.3	20.8
	IBM3 + ME	42.5	26.4	17.2
$1^5 2^5 3^3 4^3$	IBM4	41.8	24.9	17.4
	IBM4 + ME	41.5	24.3	14.1
$1^5 2^5 3^3 4^3 5^3$	IBM5	41.5	24.8	16.2
	IBM5 + ME	41.5	24.5	14.3

Table 10 AER [%] on VERBMOBIL task.

		Size of training corpus		
Training scheme	Model	0.5 K	8 K	34 K
1^5	IBM1	27.7	19.2	17.6
	IBM1 + ME	24.6	16.6	13.7
$1^5 2^5$	IBM2	26.8	15.7	13.5
	IBM2 + ME	25.3	14.1	10.8
$1^5 2^5 3^3$	IBM3	25.6	13.7	10.8
	IBM3 + ME	24.1	11.6	8.8
$1^5 2^5 3^3 4^3$	IBM4	23.6	10.0	7.7
	IBM4 + ME	22.8	9.3	7.0
$1^5 2^5 3^3 4^3 5^3$	IBM5	22.6	9.9	7.2
	IBM5 + ME	22.3	9.6	6.8

$$recall = \frac{|\ A \cap S|}{|S|},\ precision = \frac{|A \cap P|}{|A|}, AER(S, P; A) = 1 - \frac{|A \cap S| + |A \cap P|}{|A| + |S|}$$

Thus, a recall error can only occur if a $S(ure)$ alignment is not found. A precision error can only occur if the alignment found is not even $P(ossible)$.

The set of sentence pairs, for which the manual alignment was produced, was randomly selected from the training corpus. It should be emphasized that all the training was done in a completely unsupervised way, i.e. no manual alignments were used. From this point of view, there is no need to have a separate test corpus.

Table 11 Training perplexity for HANSARDS task for different training sizes and schemes.

		Size of training corpus		
Training scheme	Model	0.5 K	8 K	128 K
1^5	IBM1	34.7	47.3	72.9
	IBM1 + ME	33.9	43.4	51.9
$1^5 2^5$	IBM2	11.1	25.5	40.2
	IBM2 + ME	10.5	23.5	30.9
$1^5 2^5 3^3$	IBM3	21.3	43.3	69.8
	IBM3 + ME	20.8	39.2	49.8
$1^5 2^5 3^3 4^3$	IBM4	47.4	72.8	104.1
	IBM4 + ME	46.1	65.7	68.1
$1^5 2^5 3^3 4^3 5^3$	IBM5	8.2	13.2	19.9
	IBM5 + ME	8.1	11.9	14.1

Table 12 Training perplexity for VERBMOBIL task for different training sizes and schemes.

		Size of training corpus		
Training scheme	Model	0.5 K	8 K	34 K
1^5	IBM1	22.2	27.6	29.9
	IBM1 + ME	20.7	24.0	24.5
$1^5 2^5$	IBM2	7.7	14.6	16.9
	IBM2 + ME	7.1	13.0	14.2
$1^5 2^5 3^3$	IBM3	12.9	23.1	26.8
	IBM3 + ME	11.7	19.8	20.2
$1^5 2^5 3^3 4^3$	IBM4	24.0	32.9	35.0
	IBM4 + ME	22.0	27.7	26.9
$1^5 2^5 3^3 4^3 5^3$	IBM5	5.3	7.1	7.7
	IBM5 + ME	4.9	6.1	6.1

5.4.2. Alignment quality results. The assessment of the conventional and the integrated approaches was performed by comparing the quality of the word alignments produced by the models of each approach. A measure of this quality can be computed from the deviation that can be observed in the above word alignments with respect to a reference word alignment obtained manually. According to the experiments we have carried out so far, the differences in alignment quality of the ME integration training with respect to the simplification proposed at the end of Section 5.3 were small. Taking into account the high time consumption of the ME integration, the results presented below were computed by using the simplified approach.

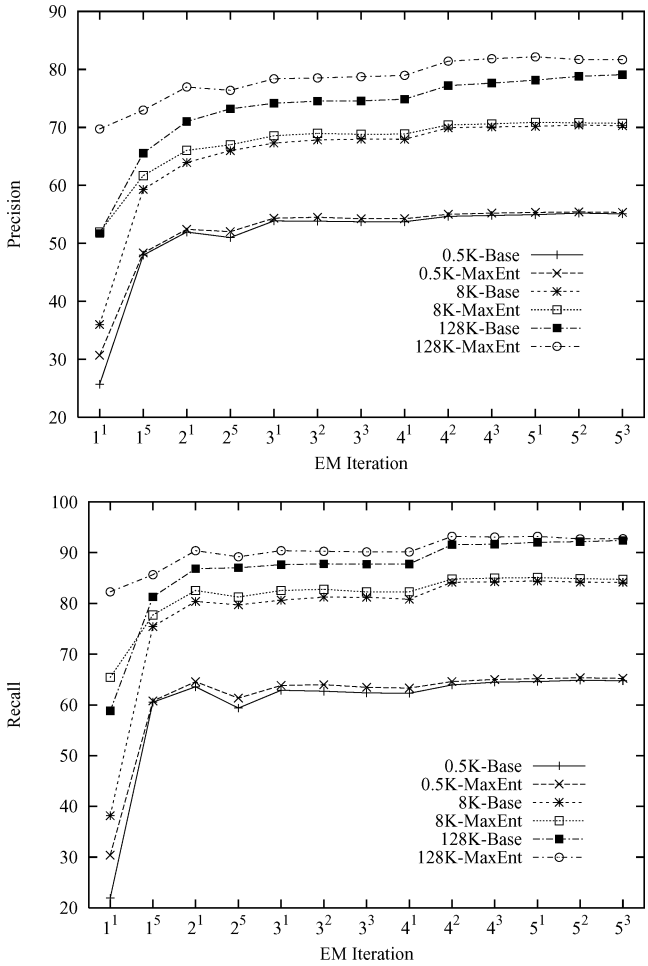

Figure 3. Precision and Recall [%] results for HANSARDS task for different corpus sizes, for every iteration of the training scheme.

Tables 9 and 10 shows the alignment quality for different training sample sizes of the HANSARDS and VERBMOBIL tasks respectively. These tables show the baseline AER for different training schemes and the corresponding values when the integration of the ME is done. The training scheme was defined in accordance with the number of iterations performed for each model (4^3 means 3 iterations of IBM Model 4).

The precision and recall results for the HANSARDS task (with and without ME training) for different sizes of the training corpus are shown in Figure 3. The corresponding precision and recall results for the VERBMOBIL task are shown in Figure 4. In all these figures, `Base` stands for the conventional training, and `MaxEnt` stands for the integrated maximum entropy training.

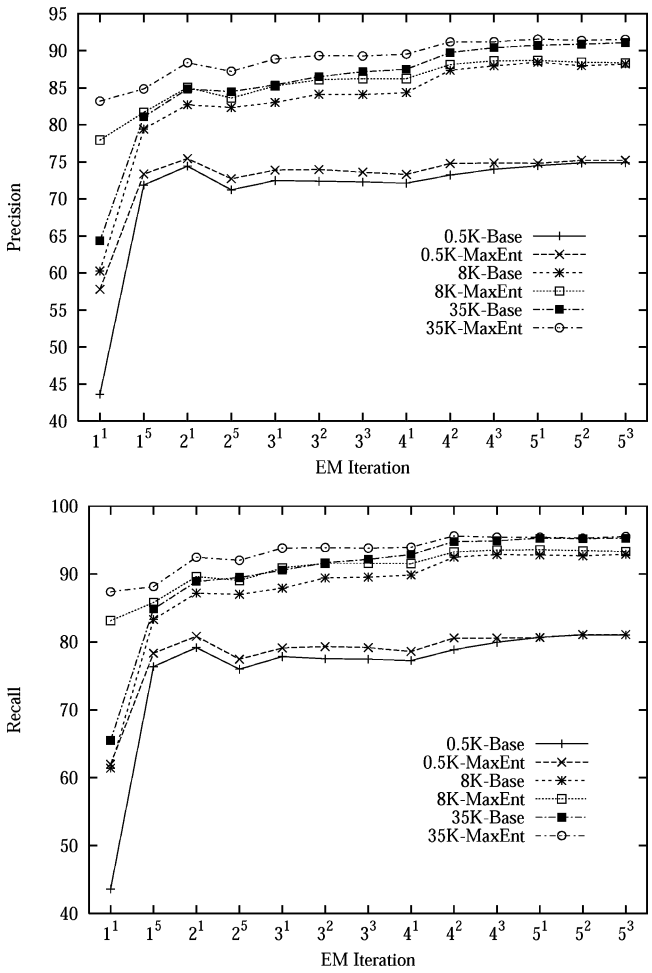

Figure 4. Precision and Recall [%] results for VERBMOBIL task for different corpus sizes, for every iteration of the training scheme.

In Tables 11 and 12, the training perplexity for different training schemes (with and without the use of the integration of the ME) are shown. These values corroborate the perplexity experiments carried out in the previous section; that is, in all cases, better model perplexity is obtained when the ME integration is used.

We observe that the alignment error rate improved when using the context-dependent lexicon models. For the VERBMOBIL task, the improvements were smaller than for the HANSARDS task, which might be due to the fact that the baseline alignment quality was already very good. It can be seen that greater improvements were obtained for the simpler models.

As can be observed in the precision and recall figures for both tasks, in all cases, higher precision and higher recall were obtained when using the context-dependent lexicon models. It can also be observed that the differences were small when small sizes of the training corpora were used, being practically the same for the 500-sentence training corpus.

As expected, the ME training plays a more important role when larger sizes of the corpus are used. For the smallest corpora, the number of training events for the ME models was very low, so it is not possible to disambiguate some translations/alignments for different contexts. For larger sizes of the corpora, greater improvements were obtained. Therefore, we expect to obtain better improvements when using even larger corpora.

After observing the common alignment errors, we plan to include more discriminant features that would provide greater improvements. We also expect improvements by performing a refined modeling of the rare/infrequent words, which are currently not taken into account by present ME models.

5.4.3. Conclusions. In this section, we have shown how an efficient and straightforward integration of ME context-dependent models within a maximum likelihood training of statistical translation models can be done.

The results presented here are directly comparable with those presented in Och and Ney, (2003). We obtained better results when using ME context-dependent models when they are applied to the family of the IBM alignment models. In that work, the authors used the HMM alignment model presented in (Och & Ney, 2000) instead of IBM Model 2 in the training scheme as a starting point to train the fertility models (IBM models 3, 4 and 5) yielding slightly better results. According to those results the alignment quality when using the HMM model instead of IBM Model 2 is better. We also plan to include the ME models within the EM training of the HMM alignment model in order to obtain even better results.

According to the experiments, it is clear that the use of the refined lexicon models within the EM training also helped to the rest of the models involved in the training, that is fertility and alignment/distortion models. This is actually due to the fact that the information of the alignment/distortion and fertility parameters is also used during the estimation of the lexicon parameters, as can be seen in Brown et al. (2003).

We evaluated the quality of the alignments obtained with this new training scheme comparing the results with the baseline results. In all cases better alignment quality was obtained using the context-dependent lexicon model.

It might be possible to design a bootstrapping strategy to combine the experiments of the previous section with the alignment models obtained with the integration. Moreover, we consider it more valuable to include the ME models within the search process stated in Eq. (2) in order to build completely integrated ME statistical machine translation systems.

6. Concluding remarks

In this paper, we have introduced refined lexicon models for statistical machine translation by using maximum entropy models. We have been able to obtain a significantly better test corpus perplexity. Despite the fact that the translation results were not statistically significant, we did obtain a slight but consistent improvement in translation quality. We

believe that by performing a rescoring on translation word graphs, we will be able to obtain a greater improvement in translation quality.

In addition, we have presented an efficient and straightforward integration of ME context-dependent models within a maximum likelihood training of statistical translation models. We have also evaluated the quality of the alignments obtained with this new training scheme comparing the results with the baseline results. In all cases, better alignment quality was obtained when using the context-dependent lexicon model.

For the future we plan to:

– Investigate the inclusion of more features in the ME model, such us dependencies on other source and target words, syntactic features such as POS tags and syntactic constituents, semantic features, and features that go beyond sentence boundaries.
– Investigate more refined feature selection methods in order to make the maximum entropy models smaller and better generalizing.
– Include ME refined lexicon models into the EM training of the HMM alignment model in order to evaluate the usefulness of the ME models in other statistical alignment models which are different from the classical IBM models.
– Design ME alignment and fertility models. This will permit easy integration of more dependencies, such as second-order alignment models, without running into the problem of an unmanageable number of alignment parameters.
– Integrate the ME models within the search process in order to have a completely integrated ME statistical machine translation system.

Acknowledgments

The authors wish to thank the anonymous reviewers for their criticisms and suggestions, and to Prof. Dr.-Ing. Hermann Ney and Dr. rer. nat. Franz J. Och from the Lehrstuhl für Informatik VI RWTH-Aachen for their invaluable help in the development of this work as well as allowing us to use some of their tools and corpora.

References

Al-Onaizan, Y., Curin, J., Jahr, M., Knight, K., Lafferty, J. D., Melamed, I. D., Purdy, D., Och, F. J., Smith, N. A., & Yarowsky, D. (1999). Statistical machine translation, Final Report, JHU Workshop. `http://www.clsp.jhu.edu/ws99/projects/mt/final_report/mt-final-report.ps`.

Bender, O., Macherey, K., Och, F. J., & Ney, H. (2003). Comparison of alignment templates and maximum entropy models for natural Language understanding. In *Proc. of the 10th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 11–18). Budapest, Hungary.

Berger, A. L., Brown, P. F., Della Pietra, S. A., Della Pietra, V. J., Gillett, J. R., Lafferty, J. D., Printz, H., & Ureš, L. (1994). The candide system for machine translation. In *Proc. ARPA Workshop on Human Language Technology* (pp. 157–162). Plainsboro, NJ.

Berger, A. L., Della Pietra, S. A., & Della Pietra, V. J. (1996). A maximum entropy approach to natural language processing. *Computational Linguistics, 22:1*, 39–72.

Brown, P. F., Della Pietra, S. A., Della Pietra, V. J., & Mercer, R. L. (1993). The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics, 19:2*, 263–311.

Charniak, E. (1999). A maximum-entropy-inspired parser. Technical ReportCS-99-12.

Darroch, J. N. & Ratcliff, D. (1972). Generalized iterative scaling for log-linear models. *Annals of Mathematical Statistics, 43*, 1470–1480.

Della Pietra, S. A., Della Pietra, V. J., & Lafferty, J. D. (1997). Inducing features in random fields. *IEEE Trans. on Pattern Analysis and Machine Inteligence, 19:4*, 380–393.

Foster, G. (2000a). Incorporating position information into a maximum entropy/minimum divergence translation model. In *Fourth Conf. on Computational Language Learning (CoNLL)* (pp. 37–52). Lisbon, Portugal.

Foster, G. (2000b). A maximum entropy/minimum divergence translation model. In *Proc. of the 38th Annual Meeting of the Association for Computational Linguistics (ACL)* (pp. 37–44). Hong Kong.

García-Varea, I., Och, F. J., Ney, H., & Casacuberta, F. (2001). Refined lexicon models for statistical machine translation using a maximum entropy approach. In *Proc. of the 39th Annual Meeting of the Association for Computational Linguistics (ACL)* (pp. 204–211). Toulouse, France.

García-Varea, I., Och, F. J., Ney, H., & Casacuberta, F. (2002). Efficient integration of maximum entropy lexicon models within the training of statistical alignment models. In C. Richardson (Ed.), *Machine Translation: From research to real users* (pp. 161–168). Lecture Notes in Artificial Intelligence. Springer-Verlag, The Association for Machine Translation in the Americas AMTA-2002 Conference. Tiburon, California.

Khudanpur, S., & Wu, J. (1999). A maximum entropy language modelto integrate N-grams and topic dependencies for conversational speech recognition. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing* (pp. 553–556). Phoenix, USA

Khudanpur, S. & Wu, J. (2000). Maximum entropy techniques for exploiting syntactic, semantic and collocational dependencies in language modeling.. *Computer, Speech and Language, 14*, 355–372.

Lauritzen, S. L. (1995). The EM algorithm for graphical association models with missing data.. *Computational Statistics & Data Analysis, 19:2*, 191–201.

Martin, S., Ney, H., & Zaplo, J. (1999). Smoothing methods in maximum entropy language modeling. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing* (pp. 545–548). Phoenix, AR.

Nießen, S., Vogel, S., Ney, H., & Tillmann, C. (1998). A DP-based search algorithm for statistical machine translation. In *COLING-ACL '98: 30th Annual Meeting of the Association for Computational Linguistics and 17th Int. Conf. on Computational Linguistics* (pp. 960-967). Montreal, Canada.

Och, F. J. (1999). An efficient method for determining bilingual word classes. In *EACL '99: Ninth Conf. of the Europ. Chapter of the Association for Computational Linguistics* (pp. 71–76). Bergen, Norway.

Och, F. J. (2000). GIZA++: Training of statistical translation models. `http://www-i6.informatik.rwth-aachen.de/~och/software/GIZA++.html`

Och, F. J. (2001). YASMET: Toolkit for conditional maximum entropy models. `http://www-i6.informatik.rwth-aachen.de/~och/software/YASMET.html`

Och, F. J., & Ney, H. (2000). A comparison of alignment models for statistical machine translation. In *COLING '00: The 18th Int. Conf. on Computational Linguistics* (pp. 1086–1090). Saarbrücken, Germany.

Och, F. J., & Ney, H. (2002). Discriminative training and maximum entropy models for statistical machine translation. In *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)* Philadelphia, PA.

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

Papineni, K. A., Roukos, S., & Ward, R. T. (1998). Maximum likelihood and discriminative training of direct translation models. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing* (pp. 189–192). Seattle, WA.

Peters, J., & Klakow, D. (1999). Compact maximum entropy language models, In *Proc. of the IEEE Workshop on Automatic Speech Recognition and Understanding*. Keystone, CO.

Ratnaparkhi, A. (1996). A maximum entropy model for part-of-speech tagging. In E. Brill, & K. Church (Eds.), *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 133–142). Somerset, New Jersey: Association for Computational Linguistics.

Ratnaparkhi, A. (1997). A simple introduction to maximum entropy models for natural language processing

Ratnaparkhi, A. Learning to parse natural language with maximum entropy models. *Machine Learning, 34*, 151.

Riezler, S., Prescher, D., Khun, J., & Johnson, M. (2000). Lexicalized stochastic modeling of constraing-based grammars using log-Linear measures and EM-training. In *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)* Hong Kong.

Rosenfeld, R. (1996). A maximum entropy approach to adaptive statistical language modeling. *Computer, Speech and Language, 10*, 187–228.

Tillmann, C., & Ney, H. (1996). Selection criteria for word trigger pairs in language modelling. *In Grammatical Inference: Learning Syntax from Sentences, 3rd International Colloquium, ICGI-96, Montpellier, France, 1996, Proceedings* (pp. 95–106). Vol. 1147. Berlin Springer.

Tillmann, C., & Ney, H. (1997). Word trigger and the EM algorithm. In *Proc. Workshop Computational Natural-Language Learning* (pp. 117-124). Madrid, Spain.

Tillmann, C., & Ney, H. (2000). Word re-ordering and DP-based search in statistical machine translation. In *Procs. Workshop on Computational Natural Language Learning (CoNLL)* (pp. 850–856). Saarbrücken, Germany.

Tillmann, C., Vogel, S., Ney, H., & Zubiaga, A. (1997). A DP-based search using monotone alignments in statistical translation. In *Proc. 35th Annual Conf. of the Association for Computational Linguistics* (pp. 289–296). Madrid, Spain.

Tomás, J., & Casacuberta, F. (2001). Monotone statistical translation using word groups. In *Procs. of the Machine Translation Summit VIII* (pp. 357–361). Santiago de Compostela, Spain.

Tomás, J., & Casacuberta, F. (2003). Combining phrase-based and template-based models in statistical machine translation. In F. Perales, A. Campillo, N. P. de la Blanca, & A. Sanfeliu (Eds.), *Pattern recognition and image analisys* (pp. 1021-1031). Vol. 2652 of *Lecture Notes in Computer Science*. Springer-Verlag, 1st Iberian Conference, IbPRIA-2003.

W. Wahlster (Ed.) *Verbmobil: Foundations of speech-to-speech translations* Berlin, Germany: Springer Verlag.

Wang, S., Schuurmans, D., & Zhao, Y. (2004). The latent maximum entropy principle. In submission.

Wang, Y.-Y., & Waibel, A. (1997). Decoding algorithm in statistical translation. In *Proc. 35th Annual Conf. of the Association for Computational Linguistics* (pp. 366–372). Madrid, Spain.

Zhou, G., & Lua, K. (1998). Word Association and MI-TRigger-based Language Modeling. In *COLING-ACL '98: 36th Annual Meeting of the Association for Computational Linguistics and 17th Int. Conf. on Computational Linguistics* (pp. 1465–1471).

Received October 7, 2003
Revised June 29, 2004
Accepted November 10, 2004