Refined Lexicon Models for Statistical Machine Translation using a Maximum Entropy Approach

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

Franz J. Och and **Hermann Ney**
Lehrstuhl für Inf. VI
RWTH Aachen
Ahornstr., 55
D-52056 Aachen, Germany
{och|ney}@cs.rwth-aachen.de

Francisco Casacuberta
Dpto. de Sist. Inf. y Comp.
Inst. Tecn. de Inf. (UPV)
Avda. de Los Naranjos, s/n
46071 Valencia, Spain
fcn@iti.upv.es

Abstract

Typically, the lexicon models used in statistical machine translation systems do not include any kind of linguistic or contextual information, which often leads to problems in performing a correct word sense disambiguation. One way to deal with this problem within the statistical framework is to use maximum entropy methods. In this paper, we present how to use this type of information within a statistical machine translation system. We show that it is possible to significantly decrease training and test corpus perplexity of the translation models. In addition, we perform a rescoring of N-Best lists using our maximum entropy model and thereby yield an improvement in translation quality. Experimental results are presented on the so-called "Verbmobil Task".

1 Introduction

Typically, the lexicon models used in statistical machine translation systems are only single-word based, that is one word in the source language corresponds to only one word in the target language.

Those lexicon models lack from context information that can be extracted from the same parallel corpus. This additional information could be:

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

To include this additional information within the statistical framework we use the maximum entropy approach. This approach has been applied in natural language processing to a variety of tasks. (Berger et al., 1996) applies this approach to the so-called IBM Candide system to build context dependent models, compute automatic sentence splitting and to improve word reordering in translation. Similar techniques are used in (Papineni et al., 1996; Papineni et al., 1998) for so-called direct translation models instead of those proposed in (Brown et al., 1993). (Foster, 2000) describes two methods for incorporating information about the relative position of bilingual word pairs into a maximum entropy translation model. Other authors have applied this approach to language modeling (Rosenfeld, 1996; Martin et al., 1999; Peters and Klakow, 1999). A short review of the maximum entropy approach is outlined in Section 3.

2 Statistical Machine Translation

The goal of the translation process in statistical machine translation can be formulated as follows: A source language string $f_1^J = f_1 \ldots f_J$ is to be translated into a target language string $e_1^I = e_1 \ldots e_I$. In the experiments reported in this paper, the source language is German and the target language is English. Every target string is considered as a possible translation for the input. If we assign a probability $Pr(e_1^I|f_1^J)$ to each pair of strings (e_1^I, f_1^J), then according to Bayes' decision rule, we have to choose the target string that maximizes the product of the target language model $Pr(e_1^I)$ and the string translation model $Pr(f_1^J|e_1^I)$.

Many existing systems for statistical machine translation (Berger et al., 1994; Wang and Waibel, 1997; Tillmann et al., 1997; Nießen et al., 1998) make use of a special way of structuring the string translation model like 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 $p(f|e)$ 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 $j \rightarrow i = a_j$ from source position j to target position $i = a_j$. The alignment 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 $Pr(f_1^J, a_1^J|e_1^I)$, the alignment a_1^J is introduced as a hidden variable.

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

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

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 is depicted in Figure 1.

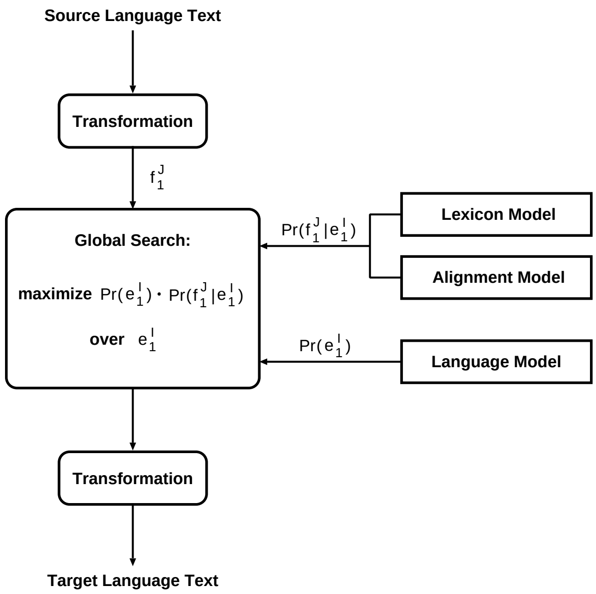

Figure 1: Architecture of the translation approach based on Bayes' decision rule.

3 Maximum entropy modeling

The translation probability $Pr(f_1^J, a_1^J|e_1^I)$ can be rewritten as follows:

$$\begin{aligned}Pr(f_1^J, a_1^J|e_1^I) &= \prod_{j=1}^{J} Pr(f_j, a_j|f_1^{j-1}, a_1^{j-1}, e_1^I) \\ &= \prod_{j=1}^{J} \Big(Pr(a_j|f_1^{j-1}, a_1^{j-1}, e_1^I) \cdot \\ &\quad Pr(f_j|f_1^{j-1}, a_1^j, e_1^I) \Big)\end{aligned}$$

Typically, the probability $Pr(f_j|f_1^{j-1}, a_1^j, e_1^I)$ is approximated by a lexicon model $p(f_j|e_{a_j})$ by dropping the dependencies on f_1^{j-1}, a_1^{j-1}, and e_1^I. Obviously, this simplification is not true for a lot of natural language phenomena. 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.

Here, the role of maximum entropy (ME) is to build a stochastic model that efficiently takes a larger context into account. In the following, we will use $p(f|x)$ to denote the probability that the ME model assigns to f in the context x in order to distinguish this model from the basic lexicon model $p(f|e)$.

In the maximum entropy approach we describe all properties that we feel are useful by so-called feature functions $\phi(x, f)$. For example, if we want to model the existence or absence of a specific word e' in the context of an English word e which has the translation f we can express this dependency using the following feature function:

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

The ME principle suggests that the optimal parametric form of a model $p(f|x)$ taking into account only the feature functions $\phi_k, k = 1, \dots, K$ is given by:

$$p(f|x) = \frac{1}{Z(x)} \exp\left(\sum_{k=1}^{K} \lambda_k \phi_k(x, f)\right)$$

Here $Z(x)$ is a normalization factor. The resulting model has an exponential form with free parameters $\lambda_k, k = 1, \dots, K$. The parameter values which maximize the likelihood for a given training corpus can be computed with the so-called GIS algorithm (general iterative scaling) or its improved version IIS (Pietra et al., 1997; Berger et al., 1996).

It is important to notice that we will have to obtain one ME model for each target word observed in the training data.

4 Contextual information and training events

In order to train the ME model $p_e(f|x)$ associated to a target word e, we need to construct a corresponding training sample from the whole 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 within 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 Model 5. We use the program GIZA++ (Och and Ney, 2000b; Och and Ney, 2000a), which is an extension of the training program available in EGYPT (Al-Onaizan et al., 1999).

Berger et al. (1996) use the words that surround a specific word pair (e, f) as contextual information. The authors propose as context the 3 words to the left and the 3 words to the right of the target word. In this work we use the following contextual information:

- Target context: As in (Berger et al., 1996) we consider a window of 3 words to the left and to the right of the target word considered.

- 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.

- Word classes: Instead of using a dependency on the word identity we include also a dependency on word classes. By doing this, we improve the generalization of the models and include some semantic and syntactic information with. 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, for a specific target word e, is composed by three items:

- The source word f aligned to e.

- The context in which the aligned pair (e, f) appears.

- The number of occurrences of the event in the training corpus.

Table 1 shows some examples of training events for the target word *"which"*.

5 Features

Once we have a set of training events for each target word we need to describe our feature functions. We do this by first specifying a large pool of possible features and then by selecting a subset of "good" features from this pool.

5.1 Features definition

All the features we consider form a triple (*pos*, *label-1*, *label-2*) where:

- *pos*: is the position that *label-2* has in a specific context.

- *label-1*: is the source word f of the aligned word pair (e, f) or the word class of the source word f ($\mathcal{F}(f)$).

- *label-2*: is one word of the aligned word pair (e, f) or the word class to which these words belong ($\mathcal{F}(f), \mathcal{E}(e)$).

Using this notation and given a context x:

$$x = \boxed{e_{i-3} \ldots e_i \ldots e_{i+3}} \boxed{f_{j-3} \ldots f_j}$$

Table 1: Some training events for the English word *"which"*. The symbol "__" is the placeholder of the English word *"which"* in the English context. In the German part the placeholder ("__") corresponds to the word aligned to "which", in the first example the German word *"die"*, the word *"das"* in the second and the word *"was"* in the third. The considered English and German contexts are separated by the double bar "||".The last number in the rightmost position is the number of occurrences of the event in the whole corpus.

Alig. word (f)	Context (x)	# of occur.
die	bar | there | , | __ | I | just | already || nette | Bar | , | __	2
das	hotel | best | , | __ | is | very | centrally || ein | Hotel | , | __	1
was	| now | , | __ | one | do | we || | jetzt | , | __	1

Table 2: Meaning of different feature categories where $\square$ represents a specific target word and $\diamond$ represents a specific source word.

Category	$\phi_{e_i}(x, f_j) = 1$ if and only if ...
1	$f_j = \diamond$
2	$f_j = \diamond$ and $\square \in$ [| | • | e_i | | |]
2	$f_j = \diamond$ and $\square \in$ [| | | e_i | • | |]
3	$f_j = \diamond$ and $\square \in$ [• | • | • | e_i | | |]
3	$f_j = \diamond$ and $\square \in$ [| | | e_i | • | • | •]
6	$f_j = \diamond$ and $\diamond \in$ [| | • | f_j]
7	$f_j = \diamond$ and $\diamond \in$ [• | • | • | f_j]

for the word pair (e_i, f_j), we use the following categories of features:

1. $(0, f_j,)$
2. $(\pm 1, f_j, e')$ and $e' = e_{i\pm 1}$
3. $(\pm 3, f_j, e')$ and $e' \in \{e_{i-3}...e_{i+3}\}$
4. $(\pm 1, \mathcal{F}(f_j), \mathcal{E}(e'))$ and $e' = e_{i\pm 1}$
5. $(\pm 3, \mathcal{F}(f_j), \mathcal{E}(e'))$ and $e' \in \{e_{i-3}...e_{i+3}\}$
6. $(-1, f_j, f')$ and $f' = f_{j-1}$
7. $(-3, f_j, f')$ and $f' \in \{f_{j-3}...f_{j-1}\}$
8. $(-1, \mathcal{F}(f_j), \mathcal{F}(f'))$ and $f' = f_{j-1}$
9. $(-3, \mathcal{F}(f_j), \mathcal{F}(f'))$ and $f' \in \{f_{j-3}..f_{j-1}\}$

Category 1 features depend only on the source word f_j and the target word e_i. A ME model that uses only those, predicts each source translation f_j with the probability $\tilde{p}_e(f_j)$ determined by the empirical data. This is exactly the standard lexicon probability $p(f|e)$ employed in the translation model described in (Brown et al., 1993) and in Section 2.

Category 2 describes features which depend in addition on the word e' one position to the left or to the right of e_i. The same explanation is valid for category 3 but in this case e' could appears in any position of the context x. Categories 4 and 5 are the analogous categories to 2 and 3 using word classes instead of words. In the categories 6, 7, 8 and 9 the source context is used instead of the target context. Table 2 gives an overview of the different feature categories.

Examples of specific features and their respective category are shown in Table 3.

Table 3: The 10 most important features and their respective category and λ values for the English word *"which"*.

Category	Feature	λ
1	(0,was,)	1.20787
1	(0,das,)	1.19333
5	(3,F35,E15)	1.17612
4	(1,F35,E15)	1.15916
3	(3,das,is)	1.12869
2	(1,das,is)	1.12596
1	(0,die,)	1.12596
5	(-3,was,@@)	1.12052
6	(-1,was,@@)	1.11511
9	(-3,F26,F18)	1.11242

5.2 Feature selection

The number of possible features that can be used according to the German and English vocabularies and word classes is huge. In order to reduce the number of features we perform a threshold based feature selection, that is every feature which occurs less than T times is not used. The aim of the feature selection is two-fold. Firstly, we obtain smaller models by using less features, and secondly, we hope to avoid overfitting on the training data.

In order to obtain the threshold T we compare the test corpus perplexity for various thresholds. The different threshold used in the experiments range 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.

We select the English words that appear at least 150 times in the training sample which are in total 348 of the 4673 words contained in the English vocabulary. Table 4 shows the different number of features considered for the 348 English words selected using different thresholds.

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

Table 4: Number of features used according to different cut-off threshold. In the second column of the table are shown the number of features used when only the English context is considered. The third column correspond to English, German and Word-Classes contexts.

	# features used	
T	English	English+German
0	846121	1581529
2	240053	500285
4	153225	330077
8	96983	210795
16	61329	131323
32	40441	80769
64	28147	49509
128	21469	31805
256	18511	22947
512	17193	19027

6 Experimental results

6.1 Training and test corpus

The "Verbmobil Task" is a 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. For the rescoring experiments we use the corpus described in Table 5.

Table 5: Corpus characteristics for translation task.

		German	English
Train	Sentences	58 332	
	Words	519 523	549 921
	Vocabulary	7 940	4 673
Test	Sentences	147	
	Words	1 968	2 173
	PP (trigr. LM)	(40.3)	28.8

To train the maximum entropy models we used the "Ristad ME Toolkit" described in (Ristad, 1997). We performed 100 iteration of the Improved Iterative Scaling algorithm (Pietra et al., 1997) using the corpus described in Table 6,

Table 6: Corpus characteristics for perplexity quality experiments.

		German	English
Train	Sentences	50 000	
	Words	454 619	482 344
	Vocabulary	7 456	4 420
Test	Sentences	8073	
	Words	64 875	65 547
	Vocabulary	2 579	1 666

which is a subset of the corpus shown in Table 5.

6.2 Training and test perplexities

In order to compute the training and test perplexities, we split the whole aligned training corpus in two parts as shown in Table 6. The training and test perplexities are shown in Table 7. As expected, the perplexity reduction in the test corpus is lower than in the training corpus, but in both cases better perplexities are obtained using the ME models. The best value is obtained when a threshold of 4 is used.

We expected to observe strong overfitting effects when a too small cut-off for features gets used. Yet, for most words the best test corpus perplexity is observed when we use all features including those that occur only once.

Table 7: Training and Test perplexities using different contextual information and different thresholds T. The reference perplexities obtained with the basic translation model 5 are TrainPP = 10.38 and TestPP = 13.22.

	English		English+German	
T	TrainPP	TestPP	TrainPP	TestPP
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

6.3 Translation results

In order to make use of the ME models in a statistical translation system we implemented a *rescoring algorithm*. This algorithm take as input the standard lexicon model (not using maximum entropy) and the 348 models obtained with the ME training. For an hypothesis sentence e_1^I and a corresponding alignment a_1^J the algorithm modifies the score $Pr(f_1^J, a_1^J | e_1^I)$ according to the refined maximum entropy lexicon model.

We carried out some preliminary experiments with the N-best lists of hypotheses provided by the translation system in order to make a rescoring of each i-th hypothesis and reorder the list according to the new score computed with the refined lexicon model. Unfortunately, our N-best extraction algorithm is sub-optimal, i.e. not the true best N translations are extracted. In addition, so far we had to use a limit of only 10 translations per sentence. Therefore, the results of the translation experiments are only preliminary.

For the evaluation of the translation quality we use 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 nevertheless acceptable so that the WER measure alone can be misleading. In order to overcome this problem, we introduce as 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. 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 use the top-10 list of hypothesis provided by the translation system described in (Tillmann and Ney, 2000) for rescoring the hypothesis using the ME models and sort them according to the new maximum entropy score. The translation results in terms of error rates are shown in Table 8. We use Model 4 in order to perform the translation experiments because Model 4 typically gives better translation results than Model 5.

We see that the translation quality improves slightly with respect to the WER and PER. The translation quality improvements so far are quite small compared to the perplexity measure improvements. We attribute this to the fact that the algorithm for computing the N-best lists is suboptimal.

Table 8: Preliminary translation results for the Verbmobil Test-147 for different contextual information and different thresholds using the top-10 translations. The baseline translation results for 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 9 shows some examples where the translation obtained with the rescoring procedure is better than the best hypothesis provided by the translation system.

7 Conclusions

We have developed refined lexicon models for statistical machine translation by using maximum entropy models. We have been able to obtain a significant better test corpus perplexity and also a slight improvement in translation quality. We believe that by performing a rescoring on translation word graphs we will obtain a more significant improvement in translation quality.

For the future we plan to investigate more refined feature selection methods in order to make the maximum entropy models smaller and better generalizing. In addition, we want to investigate more syntactic, semantic features and to include features that go beyond sentence boundaries.

References

Yaser Al-Onaizan, Jan Curin, Michael Jahr, Kevin Knight, John Lafferty, Dan Melamed, David Purdy, Franz J. Och, Noah A. Smith, and David Yarowsky. 1999. Statistical machine translation, final report, JHU workshop. `http://www.clsp.jhu.edu/ws99/projects/mt/final_report/mt-final-report.ps`.

A. L. Berger, P. F. Brown, S. A. Della Pietra, et al. 1994. The candide system for machine translation. In *Proc. , ARPA Workshop on Human Language Technology*, pages 157–162.

Adam L. Berger, Stephen A. Della Pietra, and Vincent J. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–72, March.

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

George Foster. 2000. Incorporating position information into a maximum entropy/minimum divergence translation model. In *Proc. of CoNNL-2000 and LLL-2000*, pages 37–52, Lisbon, Portugal.

Sven Martin, Christoph Hamacher, Jörg Liermann, Frank Wessel, and Hermann Ney. 1999. Assessment of smoothing methods and complex stochastic language modeling. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, volume I, pages 1939–1942, Budapest, Hungary, September.

Sonja Nießen, Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1998. A DP-based search algorithm for statistical machine translation. In *COLING-ACL '98: 36th Annual Meeting of the Association for Computational Linguistics and 17th Int. Conf. on Computational Linguistics*, pages 960–967, Montreal, Canada, August.

Franz J. Och and Hermann Ney. 2000a. Giza++: Training of statistical translation models. `http://www-i6.Informatik.RWTH-Aachen.DE/~och/software/GIZA++.html`.

Franz J. Och and Hermann Ney. 2000b. Improved statistical alignment models. In *Proc. of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 440–447, Hongkong, China, October.

Table 9: Four examples showing the translation obtained with the Model 4 and the ME model for a given German source sentence.

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.

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

K.A. Papineni, S. Roukos, and R.T. Ward. 1996. Feature-based language understanding. In *ESCA, Eurospeech*, pages 1435–1438, Rhodes, Greece.

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

Jochen Peters and Dietrich Klakow. 1999. Compact maximum entropy language models. In *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, Keystone, CO, December.

Stephen Della Pietra, Vincent Della Pietra, and John Lafferty. 1997. Inducing features in random fields. *IEEE Trans. on Pattern Analysis and Machine Inteligence*, 19(4):380–393, July.

Eric S. Ristad. 1997. Maximum entropy modelling toolkit. Technical report, Princeton Univesity.

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

Christoph Tillmann and Hermann Ney. 2000. Word re-ordering and dp-based search in statistical machine translation. In *8th International Conference on Computational Linguistics (CoLing 2000)*, pages 850–856, Saarbrücken, Germany, July.

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

Ye-Yi Wang and Alex Waibel. 1997. Decoding algorithm in statistical translation. In *Proc. 35th Annual Conf. of the Association for Computational Linguistics*, pages 366–372, Madrid, Spain, July.