Comparison of Alignment Templates and Maximum Entropy Models for Natural Language Understanding

Oliver Bender, Klaus Macherey, Franz Josef Och, and Hermann Ney
Lehrstuhl für Informatik VI, Computer Science Department
RWTH Aachen - University of Technology
D-52056 Aachen, Germany
{bender,k.macherey,och,ney}@informatik.rwth-aachen.de

Abstract

In this paper we compare two approaches to natural language understanding (NLU). The first approach is derived from the field of statistical machine translation (MT), whereas the other uses the maximum entropy (ME) framework. Starting with an annotated corpus, we describe the problem of NLU as a translation from a source sentence to a formal language target sentence. We mainly focus on the quality of the different alignment and ME models and show that the direct ME approach outperforms the alignment templates method.

1 Introduction

The objective of natural language understanding (NLU) is to extract all the information from a natural language based input which are relevant for a specific task. Typical applications using NLU components are spoken dialogue systems (Levin and Pieraccini, 1995) or speech-to-speech translation systems (Zhou et al., 2002).

In this paper we present two approaches for analyzing the semantics of natural language inputs and discuss their advantages and drawbacks. The first approach is derived from the field of statistical machine translation (MT) and is based on the source-channel paradigm (Brown et al., 1993). Here, we apply a method called alignment templates (Och et al., 1999). The alternative approach uses the maximum entropy (ME) framework (Berger et al., 1996). For both frameworks, the objective can be described as follows. Given a natural source sentence $f_1^J = f_1 ... f_j ... f_J$ we choose the formal target language sentence $e_1^I = e_1 ... e_i ... e_I$ with the highest probability among all possible target sentences:

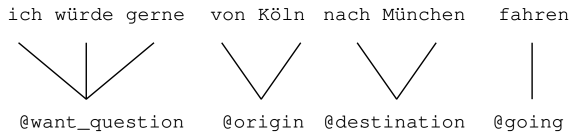

Figure 1: Example of a word / concept mapping.

$$\begin{aligned} \hat{e}_1^I &= \underset{e_1^I}{\operatorname{argmax}} \left\{ Pr(e_1^I | f_1^J) \right\} && (1) \\ &= \underset{e_1^I}{\operatorname{argmax}} \left\{ \frac{Pr(f_1^J | e_1^I) \cdot Pr(e_1^I)}{Pr(f_1^J)} \right\} \\ &= \underset{e_1^I}{\operatorname{argmax}} \left\{ Pr(f_1^J | e_1^I) \cdot Pr(e_1^I) \right\} . && (2) \end{aligned}$$

Using Bayes' theorem, Eq. 1 can be rewritten to Eq. 2, where the denominator can be neglected. The argmax operation denotes the search problem, i.e. the generation of the sequence of formal semantic concepts in the target language. An example is depicted in Figure 1. The main difference between both approaches is that the ME framework directly models the posterior probabilities whereas the statistical machine translation approach applies Bayes' theorem resulting in two distributions: the translation probability $Pr(f_1^J | e_1^I)$ and the language model probability $Pr(e_1^I)$. In the following, we compare both approaches for two NLU tasks which are derived from two different domains and show that the ME approach clearly outperforms the statistical machine translation approach within these settings.

1.1 Related Work

The use of statistical machine translation for NLU tasks was firstly proposed by (Epstein et al., 1996). Whereas (Epstein et al., 1996) model hidden clumpings, we use a method called *alignment templates*. Alignment templates have been proven to be very powerful for statistical machine translation tasks since they allow for many-to-many alignments between source and target words (Och et al., 1999). Alignment templates for NLU tasks were firstly proposed by (Macherey et al., 2001).

Applying ME translation models to NLU has been firstly suggested by (Papineni et al., 1997; Papineni et al., 1998). Here, we use a concept-based meaning representation as formal target language and propose different features and structural constraints in order to improve the NLU results.

The remainder of the paper is organized as follows: in the following section, we briefly describe the concept based meaning representation as used for the NLU task. Section 3 describes the training and search procedure of the alignment templates approach. In section 4, we outline the ME framework and describe the features that were used for the experiments. Section 5 presents results for both the alignment templates approach and the ME framework. For both approaches, experiments were carried out on two different German NLU tasks.

2 Concept-based semantic representation

A crucial decision, when designing an NLU system, is the choice of a suitable semantic representation, since interpreting a user's request requires an appropriate formalism to represent the meaning of an utterance. Different semantic representations have been proposed. Among them, case frames (Issar and Ward, 1993), semantic frames (Bennacef et al., 1994), and variants of hierarchical concepts (Miller et al., 1994) as well as flat concepts (Levin and Pieraccini, 1995) are the most prominent. Since we regard NLU as a special case of a translation problem, we have chosen a flat concept-based target language as meaning representation.

A semantic concept (in the following briefly termed as concept) is defined as the smallest unit of meaning that is relevant to a specific task (Levin and Pieraccini, 1995). Figure 1 depicts an example of a concept-based meaning representation for the input utterance 'I would like to go from Munich to Cologne' from the domain of a German train-timetable information system. The first line shows the source sentence, the last line depicts the target sentence consisting of several concepts, marked by the preceding @-symbol. The connections between the words describe the alignments between source and target words.

3 Alignment Templates

The statistical machine translation approach decomposes $Pr(e_1^I|f_1^J)$ into two probability distributions, the language model probability and the translation probability. The architecture of this method is depicted in figure 2. For the translation approach, we use the same training procedure as for the automatic translation of natural languages. When rewriting the translation probability $Pr(f_1^J|e_1^I)$ by introducing a 'hidden' alignment $a_1^J = a_1 \ldots a_j \ldots a_J$, with $a_j \in \{1, \ldots, I\}$, we obtain:

$$\begin{aligned} & Pr(f_1^J|e_1^I) \\ & \quad = \sum_{a_1^J} Pr(f_1^J, a_1^J|e_1^I) \qquad (3) \\ & \quad = \sum_{a_1^J} \prod_{j=1}^{J} Pr(f_j, a_j|f_1^{j-1}, a_1^{j-1}, e_1^I) \ . \end{aligned}$$

The IBM models as proposed by (Brown et al., 1993) and the HMM model as suggested by (Vogel et al., 1996) result from different decompositions of $Pr(f_1^J, a_1^J|e_1^I)$. For training the alignment model, we train a sequence of models of increasing complexity. Starting from the first model IBM1, we proceed over the HMM model, IBM3 up to IBM5. Using the model IBM5 as a result of the last training step, we use the alignment template approach to model whole word groups.

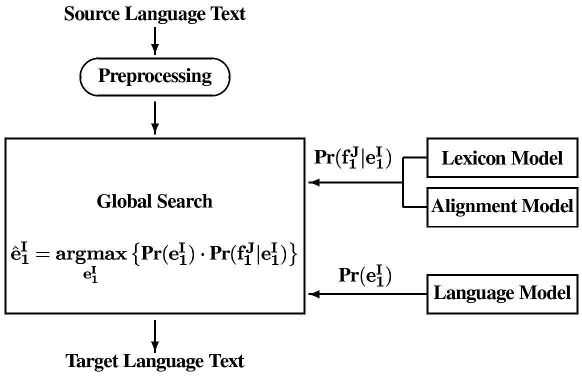

Figure 2: Architecture of the translation approach based on the source-channel paradigm.

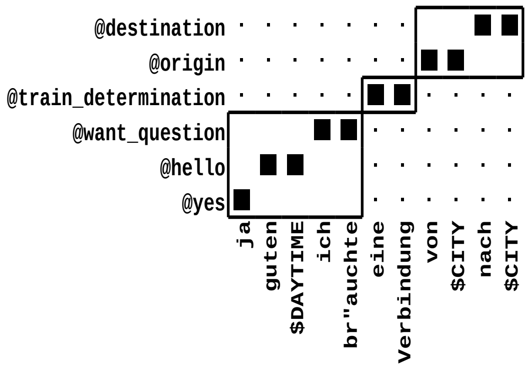

Figure 3: Example of alignment templates for representing a natural sentence as a sequence of concepts.

3.1 Model

The alignment templates approach provides a two-level alignment: a phrase level alignment and a word level alignment within the phrases. As a result, source and target sentence must be segmented into K word-groups, describing the phrases:

$$e_1^I = \widetilde{e}_1^K, \;\; \widetilde{e}_k = e_{i_{k-1}+1}, \ldots, e_{i_k}, \;\; k = 1, \ldots, K$$
$$f_1^J = \widetilde{f}_1^K, \;\; \widetilde{f}_k = f_{j_{k-1}+1}, \ldots, f_{j_k}, \;\; k = 1, \ldots, K$$

By decomposing the translation probability with the above-mentioned definitions, we arrive at:

$$\begin{aligned} & Pr(f_1^J|e_1^J) \\ & \quad = \sum_{\widetilde{a}_1^K} Pr(\widetilde{f}_1^K, \widetilde{a}_1^K|\widetilde{e}_1^K) \\ & \quad \cong \sum_{\widetilde{a}_1^K} \prod_{k=1}^{K} p(\widetilde{a}_k|\widetilde{a}_{k-1}, K) \cdot p(\widetilde{f}_k|\widetilde{e}_{a_k}) \; . \end{aligned}$$

Denote $z = (\widetilde{e}\,', \widetilde{f}\,', \widetilde{a}\,')$ an alignment template, we obtain $p(\widetilde{f}|\widetilde{e}) = \sum_z p(z|\widetilde{e}) \cdot p(\widetilde{f}|z, \widetilde{e})$. The phrase translation probability $p(\widetilde{f}|z, \widetilde{e})$ is decomposed according to the following equation:

$$\begin{aligned} & p(\widetilde{f}|(\widetilde{e}\,', \widetilde{f}\,', \widetilde{a}\,'), \widetilde{e}) \\ & \quad = \delta(\widetilde{e}, \widetilde{e}\,') \cdot \delta(\widetilde{f}, \widetilde{f}\,') \cdot \prod_{j=1}^{J} p(f_j|\widetilde{a}\,', \widetilde{e}) \; , \end{aligned}$$

where $\delta(\cdot, \cdot)$ denotes the Kronecker-function. The probability $p(f_j|\widetilde{a}\,', \widetilde{e})$ can be decomposed in the following way:

$$\begin{aligned} p(f_j|\widetilde{a}\,', \widetilde{e}) &= \sum_{i=0}^{I} p(i|j; \widetilde{a}\,') \cdot p(f_j|e_i) \\ p(i|j; \widetilde{a}\,') &= \frac{\widetilde{a}\,'(i,j)}{\sum_{i\,'} \widetilde{a}\,'(i\,',j)} \; , \end{aligned}$$

$$\widetilde{a}\,'(i,j) \;\; := \;\; \begin{cases} 1 & \text{if } (i,j) \text{ are linked in } \widetilde{a}\,' \\ 0 & \text{otherwise.} \end{cases}$$

3.2 Training

During training, we proceed over all sentence pairs and estimate the probabilities by determining the relative frequencies of applying an alignment template. Figure 3 shows an example of alignment templates computed for a sentence pair from the German TABA corpus.

3.3 Search

If we insert the alignment template model and a standard left-to-right language model in the source-channel approach (Eq. 2), we obtain the following search criterion in maximum approximation which is used in combination with beam search:

$$\begin{aligned} \hat{e}_1^I &= \text{argmax}\{Pr(e_1^I) \cdot Pr(f_1^J|e_1^I)\} \\ &= \underset{e_1^I}{\text{argmax}} \Bigg\{ \max_{K, \tilde{e}_1^K = e_1^I, \tilde{f}_1^K, \tilde{a}_1^K \in \Pi_k, z_1^K} \\ & \quad \Bigg\{ \prod_{i=1}^{I} p(e_i|e_{i-1}) \prod_{k=1}^{K} p(\tilde{a}_k|\tilde{a}_{k-1}) \cdot \\ & \quad \cdot p(z_k|\tilde{e}_{\tilde{a}_k}) \cdot \; p(\tilde{f}_k|z_k, \tilde{e}_{\tilde{a}_k}) \Bigg\} \Bigg\} \; . \qquad (4) \end{aligned}$$

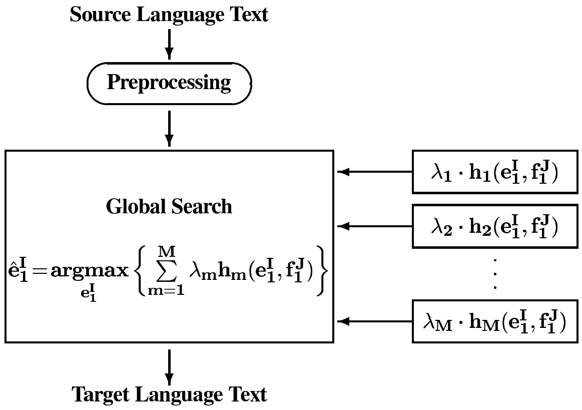

Figure 4: Architecture of the maximum entropy model approach.

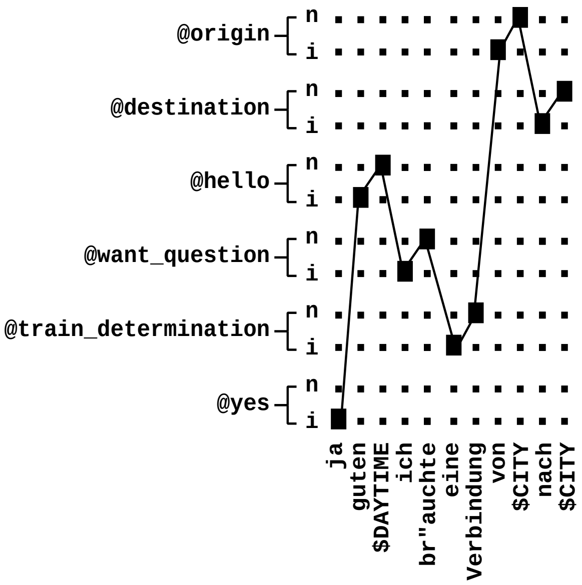

Figure 5: Example of a sentence/concept mapping using maximum entropy ('i' denotes initial concepts, 'n' non-initial concepts resp.).

4 Maximum Entropy Models

As alternative to the source-channel approach, we can directly model the posterior probability $Pr(e_1^I|f_1^J)$. A well-founded framework for doing this is maximum entropy (Berger et al., 1996). In this framework, we have a set of M feature functions $h_m(e_1^I, f_1^J), m = 1, \ldots, M$. For each feature function h_m, there is a model parameter λ_m. The posterior probability can then be modeled as follows:

$$
\begin{aligned}
& Pr(e_1^I|f_1^J) \\
&= p_{\lambda_1^M}(e_1^I|f_1^J) \\
&= \frac{\exp[\sum_{m=1}^{M} \lambda_m h_m(e_1^I, f_1^J)]}{\sum_{e'^I_1} \exp[\sum_{m=1}^{M} \lambda_m h_m(e'^I_1, f_1^J)]} . \qquad (5)
\end{aligned}
$$

The architecture of the ME approach is summarized in Figure 4.

For our approach, we determine the corresponding formal target language concept for each word of a natural language input. Therefore, we distinguish whether a word is an initial or a non-initial word of a concept. This procedure yields a one-to-one translation from source words to formal semantic concepts, i.e. the length of both sequences must be equal ($I = J$). Figure 5 depicts a one-to-one mapping applied to a sentence/concept pair from the German TABA corpus.

Further, we assume that the decisions only depend on a limited window of $f_{j-2}^{j+2} = f_{j-2}...f_{j+2}$ around the current source word f_j and on the two predecessor concepts. Thus, we obtain the following second-order model:

$$
\begin{aligned}
Pr(e_1^I|f_1^J) &= \prod_{j=1}^{J} Pr(e_j|e_1^{j-1}, f_1^J) \\
&\underset{model}{=} \prod_{j=1}^{J} p_{\lambda_1^M}(e_j|e_{j-2}^{j-1}, f_{j-2}^{j+2}) .
\end{aligned}
$$

Transition constraints: Due to the distinction between initial and non-initial concepts, we have to ensure that a non-initial concept must only follow its corresponding initial one. To guarantee this, a straightforward method is to implement a feature function that models the transitions and to set the feature values of all invalid transitions to zero, so that they will be discarded during search.

4.1 Feature functions

We have implemented a set of binary valued feature functions for our system:

Lexical features: The words f_{j-2}^{j+2} are compared to a vocabulary. Words which are not found in the vocabulary are mapped onto an 'unknown word'.

Formally, the feature

$$h_{f,d,e}(e_{j-2}^{j-1}, e_j, f_{j-2}^{j+2}) = \delta(f_{j+d}, f) \cdot \delta(e_j, e) \ , \quad d \in \{-2, ..., 2\} \ ,$$

will fire if the word f_{j+d} matches the vocabulary entry f and if the prediction for the current concept equals e. $\delta(\cdot,\cdot)$ again denotes the Kronecker-function.

Word features: Word characteristics are covered by the word features, which test for:

- Capitalization: These features will fire if f_j is capitalized, has an internal capital letter, or is fully capitalized.

- Pre- and suffixes: If the prefix (suffix) of f_j equals a given prefix (suffix), these features will fire.

Transition features: Transition features model the dependence on the two predecessor concepts:

$$h_{e',d,e}(e_{j-2}^{j-1}, e_j, f_{j-2}^{j+2}) = \delta(e_{j-d}, e') \cdot \delta(e_j, e) \ , \quad d \in \{1, 2\} \ .$$

Prior features: The single concept priors are incorporated by prior features. They just fire for the currently observed concept:

$$h_e(e_{j-2}^{j-1}, e_j, f_{j-2}^{j+2}) = \delta(e_j, e) \ .$$

Compound features: Using the feature functions defined so far, we can only specify features that refer to a single word or concept. To enable also word phrases and word/concept combinations, we introduce the following compound features:

$$\begin{aligned} & h_{\{z_1,d_1\},...,\{z_K,d_K\},e}\,(e_{j-2}^{j-1}, e_j, f_{j-2}^{j+2}) \\ & = \prod_{k=1}^{K} h_{z_k,d_k,e}\,(e_{j-2}^{j-1}, e_j, f_{j-2}^{j+2}) \ , \\ & \quad z_k \in \{f, e'\} \ , \ d_k \in \{-2, ..., 2\} \ . \end{aligned}$$

Feature selection: Feature selection plays a crucial role in the ME framework. In our system we use simple count-based feature reduction. Given a threshold K, we only include those features that have been observed on the training data at least K times. Although this method is not minimal, i.e. the reduced feature set may still contain features that are redundant or non-informative, it turned out to perform well in practice. Experiments were carried out with different thresholds. It turned out that for the NLU task, a threshold of 2 for all features achieved the best results, except for the prefix and suffix features, for which a threshold of 6 yielded best results.

4.2 Training

For the purpose of training, we consider the set of manually annotated and segmented training sentences to form a single long sentence. As training criterion, we use the maximum class posterior probability criterion:

$$\hat{\lambda}_1^M = \operatorname*{argmax}_{\lambda_1^M} \left\{ \sum_{n=1}^{N} \log p_{\lambda_1^M}(e_n|f_n) \right\} \ .$$

This corresponds to maximizing the likelihood of the ME model. The direct optimization of the posterior probability in Bayes' decision rule is referred to as discriminative training in automatic speech recognition since we directly take into account the overlap in the probability distributions. Since the optimization criterion is convex, there is only a single optimum and no convergence problems occur. To train the model parameters λ_1^M we use the Generalized Iterative Scaling (GIS) algorithm (Darroch and Ratcliff, 1972).

In practice, the training procedure tends to result in an overfitted model. To avoid overfitting, (Chen and Rosenfeld, 1999) have suggested a smoothing method where a Gaussian prior on the parameters is assumed. Instead of maximizing the probability of the training data, we now maximize the probability of the training data times the prior probability of the model parameters:

$$\hat{\lambda}_1^M = \operatorname*{argmax}_{\lambda_1^M} \left\{ p(\lambda_1^M) \cdot \sum_{n=1}^{N} p_{\lambda_1^M}(e_n|f_n) \right\} \ ,$$

where

$$p(\lambda_1^M) = \prod_m \frac{1}{\sqrt{2\pi}\sigma} \exp\left[-\frac{\lambda_m^2}{2\sigma^2} \right] \ .$$

4.3 Search

In the test phase, the search is performed using the so called maximum approximation, i.e. the most likely sequence of concepts $\hat{e}_1^I$ is chosen among all possible sequences e_1^I:

$$\begin{aligned} \hat{e}_1^I &= \underset{e_1^I}{\operatorname{argmax}} \left\{ Pr(e_1^I | f_1^J) \right\} \\ &= \underset{e_1^I}{\operatorname{argmax}} \left\{ \sum_{m=1}^{M} \lambda_m h_m(e_1^I, f_1^J) \right\} . \end{aligned}$$

Therefore, the time-consuming renormalization in Eq. 5 is not needed during search. We run a Viterbi search to find the highest probability sequence (Borthwick et al., 1998).

5 Results

Experiments were performed on the German in-house Philips TABA corpus[1] and the German in-house TELDIR corpus[2]. The TABA corpus is a text corpus in the domain of a train timetable information system (Aust et al., 1995). The TELDIR corpus is derived from the domain of a telephone directory assistance. Along with the bilingual annotation consisting of the source and target sentences, the corpora also provide the affiliated alignments between source words and concepts. The corpora allocations are summarized in table 1 and table 2. For the TABA corpus, the target language consists of 27 flat semantic concepts (23 concepts for the TELDIR application, resp.), including a filler concept. Table 3 summarizes an excerpt of the most frequently observed concepts.

In order to improve the quality of both approaches, we used a set of word categories. Since it is unlikely that every city name is observed during training, all city names were mapped onto the category `$CITY{city name}`. Table 4 shows an excerpt of different categories which were used for both the training and the testing corpora.

We have computed three different evaluation criteria:

- The *concept error rate* (CER), which is equally defined to the well known word error rate. The CER describes the ratio of the sum of deleted, inserted, and substituted concepts w.r.t. a Levenshtein-alignment for a given reference concept-string, and the total number of concepts in all reference strings.

- The *sentence error rate* (SER), which is defined as ratio between the number of falsely translated sentences and the total number of sentences w.r.t. the concept-level.

- The *concept-alignment error rate* (C-AER), which is defined as the ratio of the sum of falsely aligned words, i.e. words mapped onto the wrong concept, and the total number of words in the reference (Macherey et al., 2001).

The error rates obtained by using the alignment templates method are summarized in table 5

[1]The TABA corpus was kindly provided by Philips Forschungslaboratorien Aachen.

[2]The data-collection was partially funded by Ericsson Eurolab Deutschland GmbH.

Table 1: Training and testing conditions for the TABA corpus.

		Natural Language	Concept Language
Train	Sentences	25 009	
	Tokens	87 213	48 325
	Vocabulary	1 911	27
	Singletons	621	0
Test	Sentences	8 015	
	Tokens	22 963	12 745
	OOV	283	0
	Trigram PP	–	4.36

Table 2: Training and testing conditions for the TELDIR corpus.

		Natural Language	Concept Language
Train	Sentences	1 189	
	Tokens	6 850	3 356
	Vocabulary	752	23
	Singletons	276	2
Test	Sentences	510	
	Tokens	3 041	1 480
	OOV	194	0
	Trigram PP	–	4.49

Table 3: Excerpt of the most frequently observed concept for the TABA and the TELDIR corpus.

Concept	Example
@origin	von $CITY
@destination	nach $CITY
@person	mit Herrn $SURNAME
@organization	mit der $COMPANY

Table 4: Excerpt of used word categories.

Category	Examples
$CITY	• Berlin
	• Köln
$DAYTIME	• Morgen
	• Vormittag
$COMPANY	• BASF AG
	• Porsche
$SURNAME	• Schlegel
	• Wagner

and table 6. Table 7 and table 8 show the performance of the ME approach for different types of ME features. Starting with only lexical features, we successively extend our model by including additional feature functions. As can be seen from these results, the ME models clearly outperform the alignment models. The quality of the translation approach is achieved within the ME framework by just including lexical and transition features, and is significantly improved by adding further feature functions. Comparing the performance on the TABA task and on the TELDIR task, we see that the error rates are much lower for the TABA task than for the TELDIR task; the reason is due to the very limited training data.

One of the advantages of the ME approach results from the property that the ME framework directly models the posterior probability and allows for integrating structural information by using appropriate feature functions. Furthermore, the ME approach is consistent with the features observed on the training data, but otherwise makes the fewest possible assumptions about the distribution. Since the optimization criterion is convex, there is only a single optimum and no convergence problems occur. Due to the manual annotation using initial and non-initial concepts, we implicitly model a one-to-one alignment from nat-

Table 5: Effect of alignment templates on different error rates for the TABA corpus (Model 5* uses a given alignment in training).

Alignment Model	[%] SER	CER	C-AER
Model 5	4.2	4.3	4.3
Model 5*	3.9	3.9	3.3

Table 6: Effect of alignment templates on different error rates for the TELDIR corpus (Model 5* uses a given alignment in training).

Alignment Model	[%] SER	CER	C-AER
Model 5	16.1	6.9	13.6
Model 5*	14.5	5.9	6.7

Table 7: Dependence on the number of included feature types on different error rates for the TABA corpus.

Feature Types	[%] SER	CER	C-AER
lexical	8.8	6.7	4.6
+ transition	4.3	3.3	3.2
+ prior	2.1	1.6	1.5
+ capitalization	1.8	1.4	1.4
+ pre- & suffixes	1.6	1.2	1.3
+ compound	1.1	0.8	0.9

Table 8: Dependence on the number of included feature types on different error rates for the TELDIR corpus.

Feature Types	[%] SER	CER	C-AER
lexical	17.3	8.4	5.9
+ transition	13.5	5.6	5.4
+ prior	12.7	5.1	4.9
+ capitalization	12.0	4.8	4.9
+ pre- & suffixes	9.6	3.6	4.4
+ compound	9.0	3.6	4.1

ural language words to semantic concepts whereas the translation approach tries to learn the hidden alignment automatically. We investigated the effect of this difference by keeping the segmentation of the training data fixed for the translation approach. This approach is referred to as Model 5*, and the results are shown in table 5 and table 6. As can be seen from these tables, this variant of the translation approach has a somewhat lower error rate, but is still outperformed by the ME approach.

6 Summary

In this paper, we have investigated two approaches for natural language understanding: the alignment templates approach which is based on the source-channel paradigm and the maximum entropy approach which directly models the posterior probability. Both approaches were tested on two different corpora. We have shown that within these settings the maximum entropy method clearly outperforms the alignment templates approach.

References

H. Aust, M. Oerder, F. Seide, and V. Steinbiss. 1995. The Philips automatic train timetable information system. *Speech Communication*, 17:249–262, November.

S. K. Bennacef, H. Bonnea-Maynard, J. L. Gauvain, L. F. Lamel, and W. Minker. 1994. A spoken language system for information retrieval. In *Proc. of the Int. Conf. on Spoken Language Processing (ICSLP'94)*, pages 1271–1274, Yokohama, Japan, September.

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

A. Borthwick, J. Sterling, E. Agichtein, and R. Grisham. 1998. NYU: Description of the MENE named entity system as used in MUC-7. In *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, 6 pages, Fairfax, VA, April. http://www.itl.nist.gov/iaui/894.02/related_projects/muc/.

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

S. Chen and R. Rosenfeld. 1999. A gaussian prior for smoothing maximum entropy models. Technical Report CMUCS-99-108, Carnegie Mellon University, Pittsburgh, PA.

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

M. Epstein, K. Papineni, S. Roukos, T. Ward, and S. Della Pietra. 1996. Statistical natural language understanding using hidden clumpings. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, volume 1, pages 176–179, Atlanta, GA, May.

S. Issar and W. Ward. 1993. CMU's robust spoken language understanding system. In *European Conf. on Speech Communication and Technology*, volume 3, pages 2147–2149, Berlin, Germany, September.

E. Levin and R. Pieraccini. 1995. Concept-based spontaneous speech understanding system. In *European Conf. on Speech Communication and Technology*, volume 2, pages 555–558, Madrid, Spain, September.

K. Macherey, F. J. Och, and H. Ney. 2001. Natural language understanding using statistical machine translation. In *European Conf. on Speech Communication and Technology*, pages 2205–2208, Aalborg, Denmark, September.

S. Miller, R. Bobrow, R. Ingria, and R. Schwartz. 1994. Hidden understanding models of natural language. In *Proceedings of the Association of Computational Linguistics*, pages 25–32, June.

F. J. Och, C. Tillmann, and H. Ney. 1999. Improved alignment models for statistical machine translation. In *Proc. of the Joint SIGDAT Conf. on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 20–28, University of Maryland, College Park, MD, June.

K. A. Papineni, S. Roukos, and R. T. Ward. 1997. Feature-based language understanding. In *European Conf. on Speech Communication and Technology*, pages 1435–1438, Rhodes, Greece, September.

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, Seattle, WA, May.

S. Vogel, H. Ney, and C. Tillmann. 1996. HMM-based word alignment in statistical translation. In *International Conference on Computational Linguistics*, volume 2, pages 836–841, August.

B. Zhou, Y. Gao, J. Sorensen, Z. Diao, and M. Picheny. 2002. Statistical natural language generation for speech-to-speech machine translation systems. In *Proc. of the Int. Conf. on Spoken Language Processing (ICSLP'02)*, pages 1897–1900, Denver, CO, September.