Natural Language Understanding Using Statistical Machine Translation

Klaus Macherey, Franz Josef Och, Hermann Ney

Lehrstuhl für Informatik VI, Computer Science Department
RWTH Aachen - University of Technology
D - 52056 Aachen, Germany
{k.macherey,och,ney}@informatik.rwth-aachen.de

Abstract

Over the past years, automatic dialogue systems and telephone-based machine inquiry systems have received increasing attention. In addition to an automatic speech recognizer and a dialogue manager, such systems consist of a natural language understanding (NLU) component. Some of the most investigated approaches to NLU are rule-based methods as *Stochastic Grammars*, which are often written manually. However, the sole usage of rule-based methods can turn out to be inflexible and the problem of reusability occurs. When extending the application scenario or changing the application's domain itself, a large part of the set of rules often must be rewritten. Therefore, techniques are desirable which help to reduce the manual effort when building up an NLU component for a new domain.

In this paper we investigate an approach to NLU, which is derived from the field of statistical machine translation. Starting from a conceptual annotated corpus, we describe the problem of NLU as a translation from a source sentence to a formal-language target sentence. Doing this, we will mainly focus on the quality of different alignment models between source and target sentences. Even though the usage of grammars cannot be totally avoided in NLU-systems, it is our goal to reduce their employment and learn the dependencies between words and their meaning automatically. Experiments were performed on the Philips in-house TABA corpus, which is a text corpus in the domain of a German train timetable information system.

1. Introduction

One of the most prominent applications of speech recognition over the past years are automatic dialogue systems. In addition to a speech recognition component and a dialogue manager, an automatic dialogue system also includes a *Natural Language Understanding* (NLU) unit in order to map a user's utterance to a machine-readable form, so that it can be processed in some further steps. Several approaches to semantic analysis of input utterances have been proposed. Among them, rule or grammar based approaches have been the most investigated subject of research in the field of NLU (see e.g. [1, 2]) and are often incorporated in actual dialogue systems. However, rule-based methods can turn out to be inflexible, when, for example, dealing with spontaneous speech effects. In case of changing the domain of the application, the whole set of rules often has to be rewritten. Other approaches to NLU are more data-oriented like the hidden understanding models, as proposed by [3], which are inspired by hidden Markov models for decoding the meaning of an utterance, or the CHRONUS system, as introduced by [4]. Both methods are based on the well known statistical source channel paradigm. Another method as proposed by IBM, which also bases on the source channel paradigm, is derived from the field of statistical machine translation. Given a natural source sentence $f_1^J = f_1 \dots f_j \dots f_J$ one seeks the (formal) target language sentence $e_1^I = e_1 \dots e_i \dots e_I$ among all possible strings, that maximizes $Pr(e_1^I|f_1^J)$ [5, 6]:

$$\begin{aligned} \hat{e}_1^I &= \operatorname*{argmax}_{e_1^I} \left\{ Pr(e_1^I|f_1^J) \right\} \\ &= \operatorname*{argmax}_{e_1^I} \left\{ Pr(f_1^J|e_1^I) \cdot Pr(e_1^I) \right\} \qquad (1) \end{aligned}$$

In this paper we will investigate a similar approach, but instead of modelling word clumps as done in [6], we use a method called *alignment templates*. Alignment templates have been proven to be very effective in statistical machine translation because they allow many-to-many alignments between source and target words [7]. Once a conceptual annotated corpus is given, the alignment templates approach is able to learn the model's free parameters entirely from the training data.

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. For a specific application, one must always try to find a compromise between accuracy and the complexity of the implementation [8]. Different semantic representations have been proposed. Among them, case frames [9], semantic frames [10], semantic graphs [11], and variants of tree-based concepts [3] as well as flat concepts [4] 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.

The remainder of this paper is organized as follows: in section 2 we will briefly describe our used concept language representing the meaning of a sentence. Section 3 outlines different alignment models and introduces the alignment templates approach. In section 4 we present results for the German Philips in-house TABA corpus. Section 5 closes with a summary and an outlook for future works.

2. Concepts and attributes

To represent the meaning of an utterance, we use a set of flat concepts. A *concept* is defined as the smallest unit of meaning that is relevant to a specific task [4]. Figure 1 depicts an example of a concept-based meaning representation for the 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

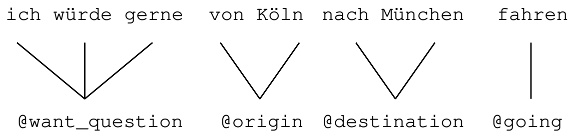

Figure 1: Example of a concept-based meaning representation for the utterance 'I would like to go from Cologne to Munich'.

alignments between source and target words. As can be derived from Figure 1, a single-word based translation from source to target words does not appear ingenious, since the meaning of a sentence phrase only becomes evident when considering the contiguous words of a single word f_j. For example, the affiliation of the city name *Köln* to the concept *@origin* is only apparent, if looking at the preceeding preposition *von* in the source sentence. To cope with this problem the alignment templates approach allows many-to-many alignments between source and target phrases. By using this approach word contexts and local reorderings are explicitly taken into account [7].

In order to perform an sql-query it is not only sufficient to determine the sequence of concepts, but also the *values* of the concepts which are called *attributes*. For example, the value of the concept *@origin* in Figure 1 is *Köln*. The values of a concept can be derived by mapping the 'relevant' aligned words onto the associated attributes. Such a mapping can be realized by a simple concept-dependent keyword mapping or a set of rules describing the meaning of the semantic unit (as is often done, when dealing with time and date expressions). If this mapping is performed independently of the contiguous concepts, the exact determination of the segment borders for each concept is necessary. An example of a misaligned word-sequence is depicted in Figure 2. In this case it is not possible

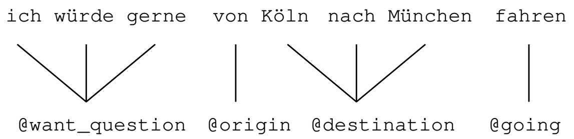

Figure 2: Example of a misaligned sentence for the utterance 'I would like to go from Cologne to Munich'.

to extract the value *Köln* of the concept *@origin*, when considering each concept on its own. Therefore, we will have a closer look at the alignment quality which is essential for this step.

3. Alignment models

When rewriting the probability $Pr(f_1^J|e_1^I)$ of equation 1 by introducing the 'hidden' alignments $a_1^J = a_1 \dots a_j \dots a_J$, with $a_j \in \{1, \dots, I\}$, we obtain:

$$\begin{aligned} Pr(f_1^J|e_1^I) &= \sum_{a_1^J} Pr(f_1^J, a_1^J|e_1^I) \qquad (2) \\ &= \sum_{a_1^J} \prod_{j=1}^{J} Pr(f_j, a_j|f_1^{j-1}, a_1^{j-1}, e_1^I) \end{aligned}$$

The different alignment models we are looking at result from different decompositions of $Pr(f_1^J, a_1^J|e_1^I)$.

3.1. HMM alignment

If we assume a first-order dependence for the alignments a_j in equation 2 and restrict the dependent quantities of the translation probability only to a_j, we arrive at the HMM alignment:

$$Pr(f_1^J|e_1^I) = \sum_{a_1^J} \prod_{j=1}^{J} [p(a_j|a_{j-1}, I) \cdot p(f_j|e_{a_j})] \qquad (3)$$

3.2. Model 1 and Model 2

Replacing the dependence in equation 3 from a_{j-1} to j, we obtain a 'zero-order' Hidden Markov Model which is similar to Model 2 as proposed by [5]. Assuming a uniform alignment probability $p(i|j, I) = \frac{1}{I}$, we obtain Model 1.

3.3. Model 3, Model 4, and Model 5

Since a detailed description of Models 3 to 5 would go beyond the scope of this paper we only sketch some basic ideas. For further details see [5]. In Model 3 and 4 the *fertilities* of target words are introduced. That is, the probability $p(\phi_i|e_i)$ that the target word e_i is aligned to

$$\phi_i \equiv \phi(e_i) = \sum_{j=1}^{J} \delta(i, a_j) \qquad (4)$$

source words is explicitly modelled (here, $\delta(\cdot, \cdot)$ denotes the Kronecker-function). Like Model 2, Model 3 is a zero-order alignment model including additional fertility parameters. A special problem of Model 3 and 4 concerns the *deficiency* of the model, that is, the same position can be chosen twice in the source string. Also a position before the first or beyond the last position may be chosen in Model 4. The deficiency of both models is removed in Model 5 by keeping track of vacant positions in the source string.

3.4. Alignment templates

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

$$\begin{aligned} e_1^I = \widetilde{e}_1^K, &\quad \widetilde{e}_k = e_{i_{k-1}+1}, \dots, e_{i_k}, \quad k = 1, \dots, K \\ f_1^J = \widetilde{f}_1^K, &\quad \widetilde{f}_k = f_{j_{k-1}+1}, \dots, f_{j_k}, \quad k = 1, \dots, K \end{aligned}$$

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

$$\begin{aligned} Pr(f_1^J|e_1^J) &= Pr(\widetilde{f}_1^K|\widetilde{e}_1^K) \\ &= \sum_{\widetilde{a}_1^K} Pr(\widetilde{f}_1^K, \widetilde{a}_1^K|\widetilde{e}_1^K) \\ &= \sum_{\widetilde{a}_1^K} \prod_{k=1}^{K} p(\widetilde{a}_k|\widetilde{a}_1^{k-1}, K) \cdot p(\widetilde{f}_k|\widetilde{e}_{a_k}) \\ &= \sum_{\widetilde{a}_1^K} \prod_{k=1}^{K} p(\widetilde{a}_k|\widetilde{a}_{k-1}, K) \cdot p(\widetilde{f}_k|\widetilde{e}_{a_k}) \qquad (5) \end{aligned}$$

By introducing an alignment template $z = (\widetilde{e}\,', \widetilde{f}\,', \widetilde{a}\,')$ we obtain the following equation:

$$p(\widetilde{f}|\widetilde{e}) = \sum_{z} p(z|\widetilde{e}) \cdot p(\widetilde{f}|z, \widetilde{e})$$

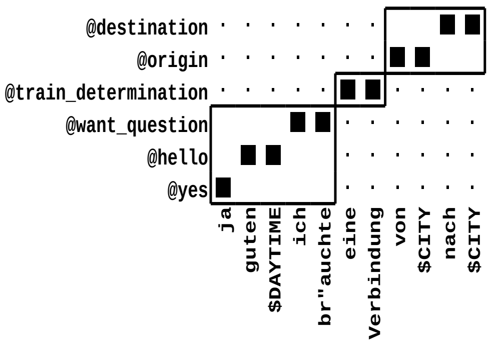

Figure 3: Example of alignment templates for the sentence pair 'Yes, hello, I need a connection from Hamburg to Munich | @yes @hello @want_question @train_determination @origin @destination' from the German TABA training corpus.

So far, we have three different probabilities which have to be modelled and trained:

- the phrase alignment probability $p(\widetilde{a}_k|\widetilde{a}_{k-1}, K)$,
- the probability of applying an alignment template $p(z|\widetilde{e})$,
- and the phrase translation probability $p(\widetilde{f}|z, \widetilde{e})$.

The phrase translation probability $p(\widetilde{f}|z, \widetilde{e})$ of using an alignment is decomposed according to the following equations:

$$p(\widetilde{f}|(\widetilde{e}\,', \widetilde{f}\,', \widetilde{a}\,'), \widetilde{e}) = \delta(\widetilde{e}, \widetilde{e}\,') \cdot \delta(\widetilde{f}, \widetilde{f}\,') \cdot \cdot \prod_{j=1}^{J} p(f_j|\widetilde{a}\,', \widetilde{e}),$$

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

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

$$p(i|j; \widetilde{a}\,') = \frac{\widetilde{a}\,'(i,j)}{\sum_{i\,'} \widetilde{a}\,'(i\,',j)},$$

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

For further details concerning the alignment templates approach and its training the reader is referred to [7]. Figure 3 sets an example of some alignment templates applied to a sentence pair from the German TABA corpus.

3.5. Alignment quality

Within the scope of NLU, we measure the quality of an alignment as the number of words which have been mapped onto the correct concepts. The number of falsely aligned words then defines the *concept alignment error rate* (C-AER):

$$\text{C-AER} := \frac{\sum_{r=1}^{R} \sum_{j=1}^{J_r} \left[1 - \delta\left((f_{r,j}, c^{ref}_{r,a_j})(f_{r,j}, c^{tst}_{r,a_j})\right)\right]}{\sum_{r=1}^{R} \sum_{j=1}^{J_r} 1} \qquad (6)$$

Here, $r = 1, \ldots, R$ denotes the sentence index and $j = 1, \ldots, J_r$ denotes the position in the source sentence. The label *ref* indicates the reference concept, *tst* denotes the concept of the testing string, respectively. The C-AER can serve as an important hint to the quality of the concept/attributes mapping.

4. Results

Experiments were performed on the German in-house Philips TABA corpus[1] which is a text corpus in the domain of a train-timetable information system [2]. Along with the bilingual annotation consisting of the source and target sentences, the corpus also provides the affiliated alignments between source words and concepts. Note that these alignments were only used as reference alignments to compute the alignment error rate, as defined in equation 6, and not as additional information during training. The target language consists of 27 concepts including a filler concept. In order to improve translation, 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 2 shows an excerpt of different categories which were used for both the training and the testing corpus. To simplify training and search effort during the translation process, the maximum length l of the alignment templates is restricted to an upper bound. In our experiments we have chosen a maximum length of 7. Figure 4 shows the effect of the maximally allowed length l w.r.t. different error criteria. As can be derived from the plot an alignment template length of $l \geq 5$ does not improve the results significantly. The corpus allocation is summarized in table 1.

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

		German	Concept
Train	Sentences	25 009	
	Words	87 213	48 325
	Voc.Entries	1 911	27
Test	Sentences	8 015	
	Words	22 963	12 745
	OOV	285	0
	PP (zerogram LM)	–	27

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,
- and the *concept-alignment error rate*.

As can be derived from Table 3, alignment Model 4 outperforms Models 1 to 3. This effect results from the explicitly modelled fertilities of the target words. For example, it is more likely that a concept like *@origin* generates two words ('from $CITY') than any other numbers of words. Table 4 shows the fertility probabilities for some concepts. Nevertheless, the deficiency of

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

Table 2: Excerpt of categories used in the TABA corpus.

Category	Examples
\$CITY	• Aachen • Köln
\$DAYTIME	• Morgen • Vormittag
\$WEEKDAY	• Montag • Dienstag
\$MONTH	• Januar • Februar
\$CARDINAL	• erster • zweiter
\$NUMBER	• null • eins

Table 3: Effect of alignment models on different error rates. All results were obtained using categorizations.

Alignment Model	CER[%]	SER[%]	C-AER[%]
Model 1	6.0	7.9	16.8
HMM	6.6	8.9	15.5
Model 3	6.1	8.0	10.5
Model 4	4.7	5.0	5.3
Model 5	4.2	4.3	4.3

Model 4 is an additional source of error. Therefore, Model 5 clearly outperforms the other alignment models.

5. Summary and outlook

In this paper we have investigated a statistical machine translation approach to natural language understanding. Choosing an appropriate target language representing the meaning of a source sentence, we have used different alignment models in order to train the free parameters of the alignment templates. Since we regard the process of NLU as two separate steps, we have defined a concept-alignment error rate in order to quantify the number of false alignments. Experiments were performed on the Philips in-house TABA corpus which is a corpus in the domain of train timetable information systems [2].

For future investigations we will test our approach on the ATIS corpus. Since the ATIS corpus does not include a concept-based representation language, we first have to generate the target language.

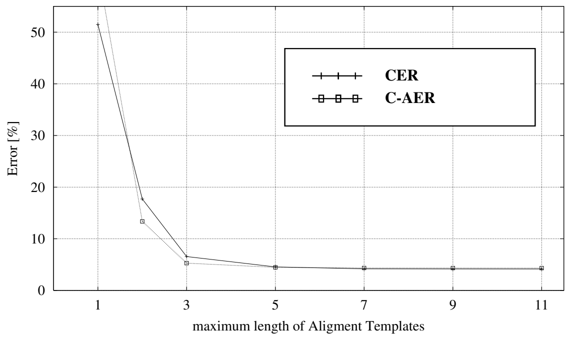

Figure 4: Effect of the alignment template length on the concept error rate and the concept-alignment error rate.

Table 4: Fertility probabilities for some concepts.

Concept	Fertility ϕ_i	$p(\phi_i \vert e_i)$
@origin	1	0.006
	2	**0.907**
	3	0.085
	≥ 4	0.002
@yes	**1**	**0.935**
	2	0.042
	3	0.023
	≥ 4	0.0
@date	1	0.420
	2	0.276
	3	0.276
	4	0.027
	≥ 5	0.001

Acknowledgement. The authors thank Philips Forschungslaboratorien Aachen for kindly providing the TABA corpus.

6. References

[1] S. Seneff, "TINA: A Natural Language System for Spoken Language Applications," *Computational Linguistics*, vol. 18, no. 1, pp. 61–86, 1992.

[2] H. Aust, *Sprachverstehen und Dialogmodellierung in natürlichsprachlichen Informationssystemen*. Ph.D. Thesis, Computer Science Department, RWTH Aachen, 1998.

[3] S. Miller, R. Bobrow, R. Ingria, and R. Schwartz, "Hidden Understanding Models of Natural Language," in *Proceedings of the Association of Computational Linguistics*, pp. 25–32, June 1994.

[4] E. Levin and R. Pieraccini, "Concept-Based Spontaneous Speech Understanding System," in *EuroSpeech'95*, vol. 2, pp. 555–558, September 1995.

[5] P. F. Brown, S. A. Della Pietra, V. J. Della Pietra, and R. L. Mercer, "The Mathematics of Machine Translation: Parameter Estimation," *Computational Linguistics*, vol. 19, no. 2, pp. 263–311, 1993.

[6] M. Epstein, K. Papineni, S. Roukos, T. Ward, and S. Della Pietra, "Statistical Natural Language Understanding Using Hidden Clumpings," in *ICASSP'96*, vol. 1, pp. 176–179, May 1996.

[7] F. J. Och, C. Tillmann, and H. Ney, "Improved Alignment Models for Statistical Machine Translation," in *EMNLP'99*, pp. 20–28, June 1999.

[8] R. Kuhn and R. de Mori, "Sentence Interpretation," in *Spoken Dialogues with Computers* (R. de Mori, ed.), ch. 14, pp. 485–522, Academic Press, 1998.

[9] S. Issar and W. Ward, "CMU's Robust Spoken Language Understanding System," in *EuroSpeech'93*, vol. 3, pp. 2147–2149, September 1993.

[10] S. K. Bennacef, H. Bonnea-Maynard, J. L. Gauvain, L. F. Lamel, and W. Minker, "A Spoken Language System for Information Retrieval," in *ICSLP'94*, pp. 1271–1274, September 1994.

[11] M. Bates, R. Bobrow, R. Ingria, S. Peters, and D. Stallard, "Advances in BBN's Spoken Language System," in *Proceedings of the Spoken Language Technology Workshop*, pp. 43–47, March 1994.