Improving Statistical Natural Language Translation with Categories and Rules

Franz Josef Och and Hans Weber
FAU Erlangen - Computer Science Institute,
IMMD VIII - Artificial Intelligence,
Am Weichselgarten 9, 91058 Erlangen - Tennenlohe, Germany
{faoch,weber}@immd8.informatik.uni-erlangen.de

Abstract

This paper describes an all level approach on statistical natural language translation (SNLT). Without any predefined knowledge the system learns a statistical translation lexicon (STL), word classes (WCs) and translation rules (TRs) from a parallel corpus thereby producing a generalized form of a word alignment (WA). The translation process itself is realized as a beam search. In our method example–based techniques enter an overall statistical approach leading to about 50 percent correctly translated sentences applied to the very difficult English–German VERBMOBIL spontaneous speech corpus.

1 Introduction

In SNLT the transfer itself is realized as a maximization process of the form

$$\text{Trans}(\mathbf{d}) = \text{argmax}_{\mathbf{e}} \; P(\mathbf{e}|\mathbf{d}) \qquad (1)$$

Here $\mathbf{d}$ is a given source language (SL) sentence which has to be translated into a target language (TL) sentence $\mathbf{e}$. In order to model the distributions $P(\mathbf{e}|\mathbf{d})$ all approaches in SNLT use a "divide and conquer" strategy of approximating $P(\mathbf{e}|\mathbf{d})$ by a combination of simpler models. The problem is to reduce parameters in a sufficient way but end up with a model still able to describe the linguistic facts of natural language translation.

The work presented here uses two approximations for $P(\mathbf{e}|\mathbf{d})$. One approximation is used for to gain the relevant parameters in training while a modified formula is subject of decoding translations. In detail, we impose the following modifications with respect to approaches published in the last decade: 1. A refined distance weight for the STL probabilities is used which allows for a good modeling of the effects caused by syntactic phrases. 2. In order to account for collocations a WA technique is used, where one–to–n and n–to–one WAs are allowed. 3. For the translation WCs are used which are constructed using clustering techniques, where the STL forms a part of the optimization criterion. 4. A set of TRs is learned mapping sequences of SL WCs to sequences of TL WCs.

Throughout the paper the four topics above are described in more detail. Finally we report on experimental results produced on the VERBMOBIL corpus.

2 Learning of the Translation Lexicon

In order to determine the STL, we use a statistical model for translation and the EM algorithm to adjust its model parameters. The simple model 1 (Brown et al., 1993) for the translation of a SL sentence $\mathbf{d} = d_1 \ldots d_l$ in a TL sentence $\mathbf{e} = e_1 \ldots e_m$ assumes that every TL word is generated independently as a mixture of the SL words:

$$P(\mathbf{e}|\mathbf{d}) \sim \prod_{j=1}^{m} \sum_{i=0}^{l} t(e_j|d_i) \qquad (2)$$

In the equation above $t(e_j|d_i)$ stands for the probability that e_j is generated by d_i.

The assumption that each SL word influences every TL word with the same strength appears to be too simple. In the refined model 2 (Brown et al., 1993) alignment probabilities $a(i|j,l,m)$ are included to model the effect that the position of a word influences the position of its translation.

The phrasal organization of natural languages is well known and has been described by (Jackendorff, 1977) among many others. The tra-

ditional alignment probabilities depend on absolute positions and do not take that into account, as has already been noted by (Vogel et al., 1996). Therefore we developed a kind of relative weighting probability. The following model — which we will call the model 2′ — makes the weight between the words d_i and e_j dependent on the relative distances between the words d_k which generated the previous word e_{j-1}:

$$s(i|j, e_{j-1}, \mathbf{d}) \sim \sum_{k=0}^{l} d(i-k|l) \cdot t(e_{j-1}|d_k) \quad (3)$$

Here $d(i-k|l)$ is the probability that word d_i influences a word e_j if the previous word e_{j-1} is influenced by d_k. As an effect of such a weight a (phrase–)cluster of words being moved over a long distance receives additional 'cost' only at the ends of the cluster. So we have the final translation probability for model 2′:

$$P(\mathbf{e}|\mathbf{d}) \sim \prod_{j=1}^{m} \sum_{i=0}^{l} t(e_j|d_i) s(i|j, e_{j-1}, \mathbf{d}) \quad (4)$$

The parameters involved can be determined using the EM algorithm (Baum, 1972). The application of this algorithm to the basic problem using a parallel bilingual corpus aligned on the sentence level is described in (Brown et al., 1993).

3 Determining a Word Alignment

The kind of WA we use is more general than the often used WA through a vector, where every TL word is generated by exactly one SL word. We use a matrix $\mathbf{Z}$ for every sentence pair, whose fields describe whether or not two words are aligned. In this approach, multiple words can be aligned to one TL word, which is motivated by collocation phenomena as for instance German compound nouns. Alignments may look like the one in figure 1 according to our method. The matrix $\mathbf{Z}$ contains $i+1$ lines and j rows with binary values. The value $z_{ij} = 1$ ($z_{ij} = 0$) means that the word i influences (not) the word j. In figure 1 every link stands for $z_{ij} = 1$.

The models 1, 2 and 2′ and some similar models can be described in the form

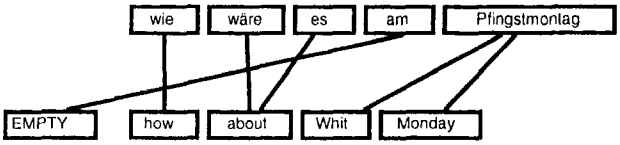

Figure 1: Alignment example.

$$P(\mathbf{e}|\mathbf{d}) \sim \prod_{j=1}^{m} \sum_{i=0}^{l} x_{ij} \quad (5)$$

where the value x_{ij} is the strength of the influence of word d_i to word e_j. We use a threshold $\theta < 1$ in such a way that while the sum $\sum_{k=0}^{s} x_{i_k j}$ of the first s values is smaller than $\theta \cdot \sum_{k=0}^{l} x_{kj}$ we set $z_{i_s j} = 0$. The other values are set to 1. The permutation $i_0, \ldots, i_l$ sorts the x_{ij} so that $x_{i_0 j} < \ldots < x_{i_l j}$.

Interestingly using such a WA technique does not in general lead to the same results when applied from TL to SL and vice versa. If we use $P(\mathbf{e}|\mathbf{d})$ or $P(\mathbf{d}|\mathbf{e})$ we receive different WAs z_{ij}^{ed} and z_{ij}^{de}. Intuitively the relation between the words of the sentences should be symmetric and there should be the same WA. It is possible to enforce the symmetry with $z_{ij} = z_{ij}^{ed} \cdot z_{ij}^{de}$, in order to make a link between two words only if there is a link in both WAs.

It is possible to include the WA into the EM algorithm for the estimation of the model probabilities. This can be done by replacing $t(e_j|d_i)$ by $t(e_j|d_i) \cdot z_{ij}$. The resulting STL becomes much cleaner in the sense that it does not contain so many wrong entries (see section 7).

4 Learning of Translation Rules

The incorporation of TRs adds an "example–based" touch to the statistical approach. In a very naive approach a TR could be represented by a translation example. The obvious advantage is an expectable good quality of the translated sentences. The disadvantage is the fact that almost no sentence can be translated because every corpus would have too few examples — the generalization capability of the naive approach is very limited.

We desired a general kind of TR which does not use explicit linguistic properties of the used languages. In addition the rules should generalize from very sparse data. Therefore it seemed

natural to use WCs and shorter sequences to end up with a set of rather general rules. In order to achieve a good learning performance, all the WCs of a language are pairwise disjoint (see section 5). The function $C(\cdot)$ gives the class of a word or the sequence of WCs of a sequence of words.

Our TRs are triples $(\mathbf{D}, \mathbf{E}, \mathbf{Z})$ where $\mathbf{D}$ is a sequence of SL WCs, $\mathbf{E}$ is a sequence of TL WCs and $\mathbf{Z}$ is a WA matrix between $\mathbf{D}$ and $\mathbf{E}$. For using one rule in the translation process we first rewrite the probability $P(\mathbf{e}|\mathbf{d})$:

$$P(\mathbf{e}|\mathbf{d}) = \sum_{\mathbf{E},\mathbf{Z}} P(\mathbf{E}, \mathbf{Z}|\mathbf{d}) \cdot P(\mathbf{e}|\mathbf{E}, \mathbf{Z}, \mathbf{d}) \qquad (6)$$

In order to simplify the maximization (equation 1) we use only the TR which gives the maximum probability.

During the learning of those TRs we count all extractable rules occurring in the aligned corpus and define the probability $p(\mathbf{E}, \mathbf{Z}|C(\mathbf{d})) \approx P(\mathbf{E}, \mathbf{Z}|\mathbf{d})$ in terms of the relative frequency.

We approximate $P(\mathbf{e}|\mathbf{E}, \mathbf{Z}, \mathbf{d})$ by simpler probabilities, so that we finally need a language model $p(e_j|e_1^{j-1})$, a translation model $p(e_j|\mathbf{d}, \mathbf{Z})$ and a probability $p(e_j|E_j)$. For $p(e_j|e_1^{j-1})$ we use a class–based polygram language model (Schukat-Talamazzini, 1994). For the translation probability $p(e_j|\mathbf{d}, \mathbf{Z})$ we use model 1 and include the information of the WA:

$$p(e_j|\mathbf{d}, \mathbf{Z}) := \sum_{i=0}^{l} t(e_j|d_i) \cdot z_{ij} \qquad (7)$$

Figure 2 shows how the application of those rules works in principle. We arrive at a list of word hypotheses with probabilities for each position. Neglecting the language model, the best decision would be to independently choose the most probable word for every position.

In general the translation of a sentence involves more than one rule and usually there are many rules applicable. An applicable rule is one where the sequence of SL WCs matches a sequence of WCs in the sentence. So in the general case we have to decide for a set of rules we want to apply. This set of rules has to cover the sentence, this means that every word is used in a rule and that no word is used twice or more times. The next step is to decide how to arrange the generated units to get the translated sentence. Finally we have to decide for every position which word to use. We want all those decisions to be optimal in the sense that the following product is maximized:

$$p(\mathbf{e}^{(j_1)} \circ \cdots \circ \mathbf{e}^{(j_L)}) \cdot \prod_{k=1}^{L} p(\mathbf{Z}^{(k)}, \mathbf{E}^{(k)}|C(\mathbf{d}^{(k)}) \cdot p(\mathbf{e}^{(j_k)}|\mathbf{Z}^{(k)}, \mathbf{E}^{(k)}, \mathbf{d}^{(k)}) \qquad (8)$$

Here L is the number of SL units, $\mathbf{d}^{(k)}$ is the k-th SL unit, $\mathbf{e}^{(k)}$ is the k-th TL unit and $j_1, \ldots, j_L$ is a permutation of the numbers $1, \ldots, L$.

5 Learning of Category Systems

During the last decade some publications have discussed the problem of learning WCs using clustering techniques based on maximum likelihood criteria applied to single language corpora. The question which we pose in addition is: Which WCs are suitable for translation? It seems to make sense to require that the used WCs in the two languages are correlated, so that the information about the class of a SL word gives much information about the class of the generated TL word. Therefore it has been argued in (Fung and Wu, 1995) that independently generated WCs are not good for the use in translation.

For the automatic generation of class systems exists a well known procedure (see (Kneser and Ney, 1993), (Och, 1995)) which maximizes the perplexity of the language model for a training corpus by moving one word from a class to another in an iterative procedure. The function $ML(\mathcal{C}|N_{w\to w'})$ which has to be optimized depends only on the count function $N_{w\to w'}$ which counts the frequency that the word w' comes after the word w.

Using two sets of WCs for the TL and SL which are independent (method `INDEP`) does not guarantee that those WCs are much correlated. The resulting WCs have only the property that the information about the class of a word w has much information about the class of the following word w'. We want for the WCs used for translation that the information about the WC of a word has much information about the WC of the translation. For the use of the standard method for optimizing WCs we need only define a count function $N_{d\to e}$, which we do by $N_{d\to e}(d, e) := t(e|d) \cdot n(e)$. In the

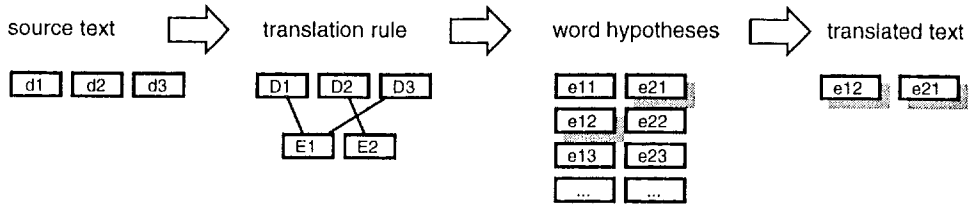

Figure 2: Application of a Rule.

same way a count function $N_{e \to d}$ can be determined and we get the new optimization criterion $ML(C_d \uplus C_e | N_{d \to e} + N_{e \to d})$. The resulting classes are strongly correlated, but rarely contain words with similar syntactic/semantic properties. To arrive at WCs having both (method COMB), we determine TL WCs with the first method and afterwards we determine SL WCs with the second method.

So we can use the well known iterative method to end up with WCs in different languages which are correlated. From those WCs we expect that they are more suitable for building the TRs from section 4 and finally result in a better overall translation performance.

6 Translation as a Search Problem

The problem of finding the translation of a sentence can be viewed as a search problem for a path with minimal cost in a tree. If we apply the negative logarithm to the product of probabilities in equation 8 we arrive at a sum of costs which has to be minimized. The costs stem from the language model, the rule probabilities and the translation probabilities. In the search tree every node represents a partial translation for the first words or a full translation. The leaves of the tree are the nodes where the applied rules define a complete cover of the SL sentence. To reduce the search space we use additional costs for changing the order of the fragments.

We use a beam search strategy (Greer et al., 1982) to find a good path in this tree. To make the search feasible we had to implement some problem specific heuristics.

7 Results

The experiments in this section have all been carried out on the bilingual German–English VERBMOBIL corpus. This corpus consists of spontaneous utterances from negotiation dialogs which had originally been produced in German. For training we used 11 500 randomly chosen sentence pairs.

The first experiment shall be understood as an illustration for our improved technique in generating a STL using the WA in the EM-algorithm. We generated a STL using 10 EM-iterations for model 1 and 10 iterations for model 2′. The whole process took about 4 hours for our corpus. Below are given some STL entries for German words. The probabilities $t(e|d)$ are written in parentheses.

- Tuesday→Dienstag (0.83), den (0.05), COMMA (0.042), am (0.038), dienstags (0.018), der (0.009), also (0.0069), passen (0.0019), diesem (0.0013), steht (0.0012)
- Frankfurt→Frankfurt (0.67), nach (0.12), in (0.081), mit (0.068), um (0.031), habe (0.02), besuchen (0.0078), wiederum (0.0036)

The top positions are always plausible translations. But there are many improper translations produced. When we include the WA in the EM algorithm as described in section 3 we can produce fewer lexicon entries of a much better quality:

- Tuesday→Dienstag (0.97), dienstags (0.029)
- Frankfurt→Frankfurt (1)

The following two corresponding WCs (out of 600) show a typical result of the method COMB to determine correlated WCs:

- Mittwoch, Donnerstag, Freitag, Sonnabend, Frühlingsanfang, Karsamstag, Volkstrauertag, Weihnachtsferien, Sommerschule, Thomas, einschließen
- Wednesday, Thursday, Friday, Thursdays, Fridays, Thomas, Veterans', mourning, national, spending, spring, summer-school

To evaluate the complete system we translated 200 randomly chosen sentences drawn from an independent test corpus and checked manually how many of them constituted acceptable translations. Since we used a spontaneous speech corpus many sentences were grammatically incorrect. A translation is classified 'correct' if the translation is an error-free (spontaneaous speech) utterance and classified 'understandable' if the intention of the utterance is translated. The 100 sentences had a mean sentence length of 10 words. The used STL was generated using model 2' (see section 2).

	correct	understandable
INDEP	46.5 %	64 %
COMB	52 %	71 %

Table 1: Quality of Translation.

Some example translations:

- was hältst du von zweiter Februar nachmittags, nach fünfzehn Uhr → what do you think about the second of February in the afternoon, after three o'clock
- I wanted to fix a time with you for a five-day business trip to Stuttgart → ich wollte mit Ihnen einen Termin ausmachen für eine fünftägige Geschäftsreise nach Stuttgart

8 Conclusions

We have presented a couple of improvements to SNLT. The most important changes are the translation model 2', the representation of WA using a matrix, a method to determine correlated WCs and the use of TRs to constrain search. In the future, the rule mechanism should be extended. So far the rules learned are only loop-free finite state transducers. Still many translation errors stem from the inability to model long distance dependencies. We intend to move to finite state cascades or context free grammars in future work. With respect to the category sets we feel that an additional morphological model could further improve the translation quality. As it stands the system still makes many errors concerning the number of nominals and verbs. This is especially important when the language pairs differ with respect to the productivity of their inflectional systems.

9 Acknowledgements

We have to thank Stefan Vogel from the RWTH Aachen explicitly, for the material he provided and Günther Görz for general promotion. The work is part of the German Joint Project VERBMOBIL. This work was funded by the German Federal Ministry for Research and Technology (BMBF) in the framework of the Verbmobil Project under Grant BMBF 01 IV 701 K 5. The responsibility for the contents of this study lies with the authors.

References

L.E. Baum. 1972. An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of Markov Processes. *Inequalities*, 3:1–8.

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.

P. Fung and D. Wu. 1995. Coerced markov models for cross-lingual lexical-tag relations. In *The Sixth Int. Conf on Theor. and Methodological Issues in Machine Translation*, pages 240–255, Leuven, Belgium, July.

K. Greer, B. Lowerre, and L. Wilcox. 1982. Acoustic Pattern Matching and Beam Searching. In *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing*, pages 1251–1254, Paris.

R. Jackendorff. 1977. X-bar-syntax: A study of phrase structure. In *Linguistic Inquiry Monograph 2.*

R. Kneser and H. Ney. 1993. Improved Clustering Techniques for Class-Based Statistical Language Modelling. In *Eurospeech*, pages 973–976.

F. J. Och. 1995. Maximum-Likelihood-Schätzung von Wortkategorien mit Verfahren der kombinatorischen Optimierung. Studienarbeit, FAU Erlangen-Nürnberg.

E.G. Schukat-Talamazzini. 1994. *Automatische Spracherkennung.* Vieweg, Wiesbaden.

S. Vogel, H. Ney, and C. Tillmann. 1996. HMM-Based Word Alignment in Statistical Translation. In *Proc. Int. Conf. on Computational Linguistics*, pages 836–841, Kopenhagen, August.