Feature-Rich Statistical Translation of Noun Phrases

Philipp Koehn and **Kevin Knight**
Information Sciences Institute
Department of Computer Science
University of Southern California
koehn@isi.edu, knight@isi.edu

Abstract

We define noun phrase translation as a subtask of machine translation. This enables us to build a dedicated noun phrase translation subsystem that improves over the currently best general statistical machine translation methods by incorporating special modeling and special features. We achieved 65.5% translation accuracy in a German-English translation task vs. 53.2% with IBM Model 4.

1 Introduction

Recent research in machine translation challenges us with the exciting problem of combining statistical methods with prior linguistic knowledge. The power of statistical methods lies in the quick acquisition of knowledge from vast amounts of data, while linguistic analysis both provides a fitting framework for these methods and contributes additional knowledge sources useful for finding correct translations.

We present work that successfully defines a subtask of machine translation: the translation of noun phrases. We demonstrate through analysis and experiments that it is feasible and beneficial to treat noun phrase translation as a subtask. This opens the path to dedicated modeling of other types of syntactic constructs, e.g., verb clauses, where issues of subcategorization of the verb play a big role.

Focusing on a narrower problem allows not only more dedicated modeling, but also the use of computationally more expensive methods.

We go on to tackle the task of noun phrase translation in a maximum entropy reranking framework. Treating translation as a reranking problem instead of as a search problem enables us to use features over the full translation pair. We integrate both empirical and symbolic knowledge sources as features into our system which outperforms the best known methods in statistical machine translation.

Previous work on defining subtasks within statistical machine translation has been performed on, e.g., noun-noun pair (Cao and Li, 2002) and named entity translation (Al-Onaizan and Knight, 2002).

2 Noun Phrase Translation as a Subtask

In this work, we consider both noun phrases and prepositional phrases, which we will refer to as NP/PPs. We include prepositional phrases for a number of reasons. Both are attached at the clause level. Also, the translation of the preposition often depends heavily on the noun phrase (*in the morning*). Moreover, the distinction between noun phrases and prepositional phrases is not always clear (note the Japanese *bunsetsu*) or hard to separate (German joining of preposition and determiner into one lexical unit, e.g., *ins* $\sim$ *in das* $\rightarrow$ in the).

2.1 Definition

We define the NP/PPs in a sentence as follows:

> Given a sentence s and its syntactic parse tree t, the NP/PPs of the sentence s are the subtrees t_i that contain at least one noun and no verb, and are not part of a larger subtree that contains no verb.

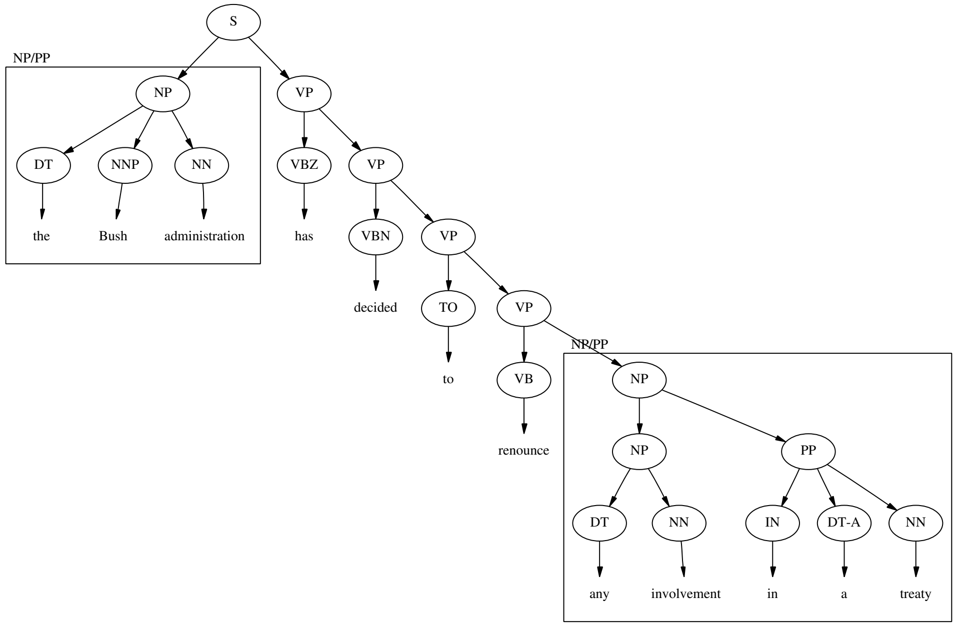

Figure 1: The noun phrases and preposition phrases (NP/PPs) addressed in this work

The NP/PPs are the maximal noun phrases of the sentence, not just the base NPs. This definition excludes NP/PPs that consist of only a pronoun. It also excludes noun phrases that contain relative clauses. NP/PPs may have connectives such as *and*.

For an illustration, see Figure 1.

2.2 Translation of NP/PPs

To understand the behavior of noun phrases in the translation process, we carried out a study to examine how they are translated in a typical parallel corpus. Clearly, we cannot simply expect that certain syntactic types in one language translate to equivalent types in another language. Equivalent types might not even exist.

This study answers the questions:

- Do human translators translate noun phrases in foreign texts into noun phrases in English?
- If all noun phrases in a foreign text are translated into noun phrases in English, is an acceptable sentence translation possible?
- What are the properties of noun phrases which cannot be translated as noun phrases without rendering the overall sentence translation unacceptable?

Using the Europarl corpus[1], we consider a translation task from German to English. We marked the NP/PPs in the German side of a small 100 sentence parallel corpus manually. This yielded 168 NP/PPs according to our definition.

We examined if these units are realized as noun phrases in the English side of the parallel corpus. This is the case for 75% of the NP/PPs.

Second, we tried to construct translations of these NP/PPs that take the form of NP/PPs in English in an overall acceptable translation of the sentence. We could do this for 98% of the NP/PPs.

The four exceptions are:

- *in Anspruch* genommen; Gloss: take *in demand*
- *Abschied* nehmen; take *good-bye*
- *meine Zustimmung* geben; give *my agreement*
- *in der Hauptsache*; *in the main-thing*

The first three cases are noun phrases or prepositional phrases that merge with the verb. This is similar to the English construction *make an observation*, which translates best into some languages as a verb equivalent to *observe*. The fourth example, literally translated as *in the main thing*, is best translated as *mainly*.

[1] Available at `http://www.isi.edu/~koehn/`

Why is there such a considerable discrepancy between the number of noun phrases that *can* be translated as noun phrases into English and noun phrases that *are* translated as noun phrases?

The main reason is that translators generally try to translate the meaning of a sentence, and do not feel bound to preserve the same syntactic structure. This leads them to sometimes arbitrarily restructure the sentence. Also, occasionally the translations are sloppy.

The conclusion of this study is: Most NP/PPs in German are translated to English as NP/PPs. Nearly all of them, 98%, *can* be translated as NP/PPs into English. The exceptions to this rule should be treated as special cases and handled separately.

We carried out studies for Chinese-English and Portuguese-English NP/PPs with similar results.

2.3 The Role of External Context

One interesting question is if external context is necessary for the translation of noun phrases. While the sentence and document context may be available to the NP/PP subsystem, the English output is only assembled later and therefore harder to integrate.

To address this issue, we carried out a manual experiment to check if humans can translate NP/PPs without any external context. Using the same corpus of 168 NP/PPs as in the previous section, a human translator translated 89% of the noun phrases correctly, 9% had the wrong leading preposition, and only 2% were mistranslated with the wrong content word meaning.

Picking the right phrase start (e.g., preposition or determiner) can sometimes only be resolved when the English verb is chosen and its subcategorization is known. Otherwise, sentence context does not play a big role: Word choice can almost always be resolved within the internal context of the noun phrase.

2.4 Integration into an MT System

The findings of the previous section indicate that NP/PP translation can be conceived as a separate subsystem of a complete machine translation system – with due attention to special cases. We will now estimate the importance of such a system.

As a general observation, we note that NP/PPs cover roughly half of the words in news or similar texts. All nouns are covered by NP/PPs. Nouns are the biggest group of open class words, in terms of the number of distinct words. Constantly, new nouns are added to the vocabulary of a language, be it by borrowing foreign words such as *Fahrvergnügen* or *Zeitgeist*, or by creating new words from acronyms such as *AIDS*, or by other means. In addition to new words, new phrases with distinct meanings are constantly formed: *web server*, *home page*, *instant messaging*, etc. Learning new concepts from text sources when they become available is an elegant solution for this knowledge acquisition problem.

System	Correct	BLEU
Basic MT system	7%	0.16
NP/PPs translated in isolation	8%	0.17
Perfect NP/PP translation	24%	0.35

Table 1: Integration of an NP/PP subsystem: Correct sentence translations and BLEU score

In a preliminary study, we assess the impact of an NP/PP subsystem on the quality of an overall machine translation system. We try to answer the following questions:

- What is the impact on a machine translation system if noun phrases are translated in isolation?
- What is the performance gain for a machine translation system if an NP/PP subsystem provides perfect translations of the noun phrases?

We built a subsystem for NP/PP translation that uses the same modeling as the overall system (IBM Model 4), but is trained on only NP/PPs. With this system, we translate the NP/PPs in isolation, without the assistance of sentence context. These translations are fixed and provided to the general machine translation system, which does not change the fixed NP/PP translation.

In a different experiment, we also provided correct translations (motivated by the reference translation) for the NP/PPs to the general machine translation system. We carried out these experiments on the same 100 sentence corpus as in the previous sections. The 164 translatable NP/PPs are marked and translated in isolation.

The results are summarized in Table 1. Treating NP/PPs as isolated units, and translating them in iso-

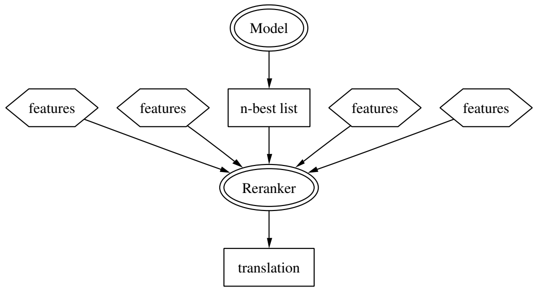

Figure 2: Design of the noun phrase translation subsystem: The base model generates an n-best list that is rescored using additional features

lation with the same methods as the overall system has little impact on overall translation quality. In fact, we achieved a slight improvement in results due to the fact that NP/PPs are consistently translated as NP/PPs. A perfect NP/PP subsystem would triple the number of correctly translated sentences. Performance is also measured by the BLEU score (Papineni et al., 2002), which measures similarity to the reference translation taken from the English side of the parallel corpus.

These findings indicate that solving the NP/PP translation problem would be a significant step toward improving overall translation quality, even if the overall system is not changed in any way. The findings also indicate that isolating the NP/PP translation task as a subtask does not harm performance.

3 Framework

When translating a foreign input sentence, we detect its NP/PPs and translate them with an NP/PP translation subsystem. The best translation (or multiple best translations) is then passed on to the full sentence translation system which in turn translates the remaining parts of the sentence and integrates the chosen NP/PP translations.

Our NP/PP translation subsystem is designed as follows: We train a translation system on a NP/PP parallel corpus. We use this system to generate an n-best list of possible translations. We then rescore this n-best list with the help of additional features. This design is illustrated by Figure 2.

3.1 Evaluation

To evaluate our methods, we automatically detected all of the 1362 NP/PPs in 534 sentences from parts of the Europarl corpus which are not already used as training data. Our evaluation metric is human assessment: Can the translation provided by the system be part of an acceptable translation of the whole sentence? In other words, the noun phrase has to be translated correctly given the sentence context.

The NP/PPs are extracted in the same way that NP/PPs are initially detected for the acquisition of the NP/PP training corpus. This means that there are some problems with parse errors, leading to sentence fragments extracted as NP/PPs that cannot be translated correctly. Also, the test corpus contains all detected NP/PPs, even untranslatable ones, as discussed in Section 2.2.

3.2 Acquisition of an NP/PP Training Corpus

To train a statistical machine translation model, we need a training corpus of NP/PPs paired with their translation. We create this corpus by extracting NP/PPs from a parallel corpus.

First, we word-align the corpus with Giza++ (Och and Ney, 2000). Then, we parse both sides with syntactic parsers (Collins, 1997; Schmidt and Schulte im Walde, 2000)[2]. Our definition easily translates into an algorithm to detect NP/PPs in a sentence.

Recall that in such a corpus, only part of the NP/PPs are translated as such into the foreign language. In addition, the word-alignment and syntactic parses may be faulty. As a consequence, initially only 43.4% of all NP/PPs could be aligned. We raise this number to 67.2% with a number of automatic data cleaning steps:

- NP/PPs that partially align are broken up
- Systematic parse errors are fixed
- Certain word types that are inconsistently tagged as nouns in the two languages are harmonized (e.g., the German *wo* and the English *today*).
- Because adverb + NP/PP constructions (e.g., *specifically this issue* are inconsistently parsed,

[2]English parser available at `http://www.ai.mit.edu/people/mcollins/code.html`, German parser available at `http://www.ims.uni-stuttgart.de/projekte/gramotron/SOFTWARE/LoPar-en.html`

we always strip the adverb from these constructions.

- German verbal adjective constructions are broken up if they involve arguments or adjuncts (e.g., *der von mir* gegessene *Kuchen* = *the by me* eaten *cake*), because this poses problems more related to verbal clauses.
- Alignment points involving punctuation are stripped from the word alignment. Punctuation is also stripped from the edges of NP/PPs.

A total of 737,388 NP/PP pairs are collected from the German-English Europarl corpus as training data.

Certain German NP/PPs consistently do not align to NP/PPs in English (see the example in Section 2.2). These are detected at this point. The obtained data of unaligned NP/PPs can be used for dealing with these special cases.

3.3 Base Model

Given the NP/PP corpus, we can use any general statistical machine translation method to train a translation system for noun phrases. As a baseline, we use an IBM Model 4 (Brown et al., 1993) system[3] with a greedy decoder[4] (Germann et al., 2001).

We found that phrase based models achieve better translation quality than IBM Model 4. Such models segment the input sequence into a number of (non-linguistic) phrases, translate each phrase using a phrase translation table, and allow for reordering of phrases in the output. No phrases may be dropped or added.

We use a phrase translation model that extracts its phrase translation table from word alignments generated by the Giza++ toolkit. Details of this model are described by Koehn et al. (2003).

To obtain an n-best list of candidate translations, we developed a beam search decoder. This decoder employs hypothesis recombination and stores the search states in a search graph – similar to work by Ueffing et al. (2002) – which can be mined with standard finite state machine methods[5] for n-best lists.

[3]Available at `http://www-i6.informatik.rwth-aachen.de/~och/software/GIZA++.html`

[4]Available at `http://www.isi.edu/licensed-sw/rewrite-decoder/`

[5]We use the Carmel toolkit available at `http://www.isi.edu/licensed-sw/carmel/`

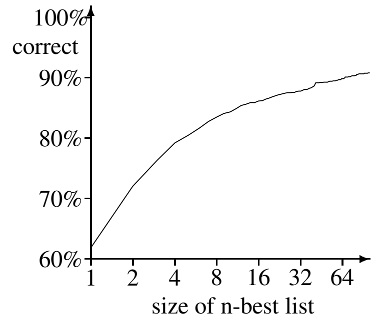

Figure 3: Acceptable NP/PP translations in n-best list for different sizes n

3.4 Acceptable Translations in the n-Best List

One key question for our approach is how often an acceptable translation can be found in an n-best list. The answer to this is illustrated in Figure 3: While an acceptable translation comes out on top for only about 60% of the NP/PPs in our test corpus, one can be found in the 100-best list for over 90% of the NP/PPs[6]. This means that rescoring has the potential to raise performance by 30%.

What are the problems with the remaining 10% for which no translation can be found? To investigate this, we carried out an error analysis of these NP/PPs. Results are given in Table 2. The main sources of error are unknown words (34%) or words for which the correct translation does not occur in the training data (14%), and errors during tagging and parsing that lead to incorrectly detected NP/PPs (28%).

There are also problems with NP/PPs that require complex syntactic restructuring (7%), and NP/PPs that are too long, so an acceptable translation could not be found in the 100-best list, but only further down the list (6%). There are also NP/PPs that cannot be translated as NP/PPs into English (2%), as discussed in Section 2.2.

3.5 Maximum Entropy Reranking

Given an n-best list of candidates and additional features, we transform the translation task from a search problem into a reranking problem, which we address using a maximum entropy approach.

As training data for finding feature values, we collected a development corpus of 683 NP/PPs. Each

[6]Note that these numbers are obtained after compound splitting, described in Section 4.1

Error	Frequency
Unknown Word	34%
Tagging or parsing error	28%
Unknown translation	14%
Complex syntactic restructuring	7%
Too long	6%
Untranslatable	2%
Other	9%

Table 2: Error analysis for NP/PPs without acceptable translation in 100-best list

NP/PP comes with an n-best list of candidate translations that are generated from our base model and are annotated with accuracy judgments. The initial features are the logarithm of the probability scores that the model assigns to each candidate translation: the language model score, the phrase translation score and the reordering (distortion) score.

The task for the learning method is to find a probability distribution $p(e|f)$ that indicates if the candidate translation e is an accurate translation of the input f. The decision rule to pick the best translation is $e_{\text{best}} = \text{argmax}_e p(e|f)$.

The development corpus provides the empirical probability distribution by distributing the probability mass over the acceptable translations $\{e_{a_i}\}$: $\tilde{p}(e_{a_i}|f) = |\{e_{a_i}\}|^{-1}$. If none of the candidate translations for a given input f is acceptable, we pick the candidates that are closest to reference translations measured by minimum edit distance.

We use a maximum entropy framework to parametrize this probability distribution as $p_\lambda(e|f) = \exp \sum_i \lambda_i h_i(f,e)$ where the h_i's are the feature values and the λ_i's are the feature weights.

Since we have only a sample of the possible translations e for the given input f, we normalize the probability distribution, so that $\sum_i p_\lambda(e_i|f) = 1$ for our sample $\{e_i\}$ of candidate translations.

Maximum entropy learning finds a set of feature values λ_i so that $E_{p_\lambda}[h_i] = E_{\tilde{p}}[h_i]$ for each feature h_i. These expectations are computed as sums over all candidate translations e for all inputs f: $\sum_{(f,e)} \tilde{p}(f) p_\lambda(e|f) h_i(f,e) = \sum_{(f,e)} \tilde{p}(f) \tilde{p}(e|f) h_i(f,e)$.

A nice property of maximum entropy training is that it converges to a global optimum. There are a number of methods and tools available to carry out this training of feature values. We use the toolkit[7] developed by Malouf (2002). Berger et al. (1996) and Manning and Schütze (1999) provide good introductions to maximum entropy learning.

Note that any other machine learning, such as support vector machines, could be used as well. We chose maximum entropy for its ability to deal with both real-valued and binary features. This method is also similar to work by Och and Ney (2002), who use maximum entropy to tune model parameters.

4 Properties of NP/PP Translation

We will now discuss the properties of NP/PP translation that we exploit in order to improve our NP/PP translation subsystem. The first of these (compounding of words) is addressed by preprocessing, while the others motivate features which are used in n-best list reranking.

4.1 Compound Splitting

Compounding of words, especially nouns, is common in a number of languages (German, Dutch, Finnish, Greek), and poses a serious problem for machine translation: The word *Aktionsplan* may not be known to the system, but if the word were broken up into *Aktion* and *Plan*, the system could easily translate it into *action plan*, or *plan for action*.

The issues for breaking up compounds are: Knowing the morphological rules for joining words, resolving ambiguities of breaking up a word (*Hauptsturm* $\rightarrow$ *Haupt-Turm* or *Haupt-Sturm*), and finding the right level of splitting granularity (*Frei-Tag* or *Freitag*).

Here, we follow an approach introduced by Koehn and Knight (2003): First, we collect frequency statistics over words in our training corpus. Compounds may be broken up only into known words in the corpus. For each potential compound we check if morphological splitting rules allow us to break it up into such known words.

Finally, we pick a splitting option (perhaps not breaking up the compound at all). This decision is based on the frequency of the words involved.

[7] Available at `http://www-rohan.sdsu.edu/~malouf/pubs.html`

Specifically, we pick the splitting option S with highest geometric mean of word frequencies of its n parts p_i: $S_{\text{best}} = \text{argmax}_S(\prod_{p_i \in S} \text{count}(p_i))^{\frac{1}{n}}$

The German side of both the training and testing corpus is broken up in this way. The base model is trained on a compound-split corpus, and input is broken up before being passed on to the system.

This method works especially well with our phrase-based machine translation model, which can recover more easily from too eager or too timid splits than word-based models. After performing this type of compound splitting, hardly any errors occur with respect to compounded words.

4.2 Web n-Grams

Generally speaking, the performance of statistical machine translation systems can be improved by better translation modeling (which ensures correspondence between input and output) and language modeling (which ensures fluent English output). Language modeling can be improved by different types of language models (e.g., syntactic language models), or additional training data for the language model.

Here, we investigate the use of the web as a language model. In preliminary studies we found that 30% of all 7-grams in new text can be also found on the web, as measured by consulting the search engine Google[8], which currently indexes 3 billion web pages. This is only the case for 15% of 7-grams generated by the base translation system.

There are various ways one may integrate this vast resource into a machine translation system: By building a traditional n-gram language model, by using the web frequencies of the n-grams in a candidate translation, or by checking if all n-grams in a candidate translation occur on the web.

We settled on using the following binary features: Does the candidate translation as a whole occur in the web? Do all n-grams in the candidate translation occur on the web? Do all n-grams in the candidate translation occur at least 10 times on the web? We use both positive and negative features for n-grams of the size 2 to 7.

We were not successful in improving performance by building a web n-gram language model or using the actual frequencies as features. The web may be too noisy to be used in such a straight-forward way without significant smoothing efforts.

[8] `http://www.google.com/`

4.3 Syntactic Features

Unlike in decoding, for reranking we have the complete candidate translation available. This means that we can define features that address any property of the full NP/PP translation pair. One such set of features is syntactic features.

Syntactic features are computed over the syntactic parse trees of both input and candidate translation. For the input NP/PPs, we keep the syntactic parse tree we inherit from the NP/PP detection process. For the candidate translation, we use a part-of-speech tagger and syntactic parser to annotate the candidate translation with its most likely syntactic parse tree.

We use the following three syntactic features:

- Preservation of the number of nouns: Plural nouns generally translate as plural nouns, while singular nouns generally translate as singular
- Preservation of prepositions: base prepositional phrases within NP/PPs generally translate as prepositional phrases, unless there is movement involved. BaseNPs generally translate as baseNPs. German genitive baseNP are treated as basePP.
- Within a baseNP/PP the determiner generally agree in number with the final noun (e.g., not: this nice green flowers).

The features are realized as integers, i.e., how many nouns did not preserve their number during translation?

These features encode relevant general syntactic knowledge about the translation of noun phrases. They constitute soft constraints that may be overruled by other components of the system.

5 Results

As described in Section 3.1, we evaluate the performance of our NP/PP translation subsystem on a blind test set of 1362 NP/PPs extracted from 534 sentences. The contributions of different components of our system are displayed in Table 3.

Starting from the IBM Model 4 baseline, we achieve gains using our phrase-based translation model (+5.5%), applying compound splitting to

System	NP/PP Correct		BLEU
IBM Model 4	724	53.2%	0.172
Phrase Model	800	58.7%	0.188
Compound Splitting	838	61.5%	0.195
Re-Estimated Param.	858	63.0%	0.197
Web Count Features	881	64.7%	0.198
Syntactic Features	892	65.5%	0.199

Table 3: Improving noun phrase translation with special modeling and additional features: Correct NP/PPs and BLEU score for overall sentence translation

training and test data (+2.8%), re-estimating the weights for the system components using the maximum entropy reranking frame-work (+1.5%), adding web count features (+1.7%) and syntactic features (+0.8%). Overall we achieve an improvement of 12.3% over the baseline. Improvements of 2.5% are statistically significant given the size of our test corpus.

Table 3 also provides scores for overall sentence translation quality. The chosen NP/PP translations are integrated into a general IBM Model 4 system that translates whole sentences. Performance is measured by the BLEU score, which measures similarity to a reference translation. As reference translation we used the English side of the parallel corpus. The BLEU scores track the improvements of our components, with an overall gain of 0.027.

6 Conclusions

We have shown that noun phrase translation can be separated out as a subtask. Our manual experiments show that NP/PPs can almost always be translated as NP/PPs across many languages, and that the translation of NP/PPs usually does not require additional external context.

We also demonstrated that the reduced complexity of noun phrase translation allows us to address the problem in a maximum entropy reranking framework, where we only consider the 100-best candidates of a base translation system. This enables us to introduce any features that can be computed over a full translation pair, instead of being limited to features that can be integrated into the search algorithm of the decoder, which only has access to partial translations.

We improved performance of noun phrase translation by 12.3% by using a phrase translation model, a maximum entropy reranking method and addressing specific properties of noun phrase translation: compound splitting, using the web as a language model, and syntactic features. We showed not only improvement on NP/PP translation over best known methods, but also improved overall sentence translation quality.

Our long term goal is to address additional syntactic constructs in a similarly dedicated fashion. The next step would be verb clauses, where modeling of the subcategorization of the verb is important.

References

Al-Onaizan, Y. and Knight, K. (2002). Translating named entities using monolingual and bilingual resources. In *Proceedings of ACL*.

Berger, A. L., Pietra, S. A. D., and Pietra, V. J. D. (1996). A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–69.

Brown, P. F., Pietra, S. A. D., Pietra, V. J. D., and Mercer, R. L. (1993). The mathematics of statistical machine translation. *Computational Linguistics*, 19(2):263–313.

Cao, Y. and Li, H. (2002). Base noun phrase translation using web data and the EM algorithm. In *Proceedings of CoLing*.

Collins, M. (1997). Three generative, lexicalized models for statistical parsing. In *Proceedings of ACL 35*.

Germann, U., Jahr, M., Knight, K., Marcu, D., and Yamada, K. (2001). Fast decoding and optimal decoding for machine translation. In *Proceedings of ACL 39*.

Koehn, P. and Knight, K. (2003). Empirical methods for compound splitting. In *Proceedings of EACL*.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase based translation. In *Proceedings of HLT/NAACL*.

Malouf, R. (2002). A comparison of algorithms for maximum entropy parameter estimation. In *Proceedings of CoNLL*.

Manning, C. D. and Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*. MIT Press.

Och, F. J. and Ney, H. (2000). Improved statistical alignment models. In *Proceedings of ACL*, pages 440–447, Hongkong, China.

Och, F. J. and Ney, H. (2002). Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of ACL*.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL*.

Schmidt, H. and Schulte im Walde, S. (2000). Robust German noun chunking with a probabilistic context-free grammar. In *Proceedings of COLING*.

Ueffing, N., Och, F. J., and Ney, H. (2002). Generation of word graphs in statistical machine translation. In *Proceedings of EMNLP*.