Experiments with a Noun-Phrase driven

Statistical Machine Translation System

Sanjika Hewavitharana, Alon Lavie and Stephan Vogel

Language Technologies Institute
Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh PA 15213
USA
{sanjika,alavie+,vogel+}@cs.cmu.edu

Abstract

This paper presents a noun phrase driven two-level statistical machine translation system. Noun phrases (NPs) are used as the unit of decomposition to build a two level hierarchy of phrases. English noun phrases are identified using a parser. The corresponding translations are induced using a statistical word alignment model. Identified noun phrase pairs in the training corpus are replaced with a tag to produce a NP tagged corpus. This corpus is then used to extract phrase translation pairs. Both NP translations and NP-tagged phrases are used in a two-level translation decoder: NP translations tag NPs in the first level, where NP-tagged phrases match across NPs to produce translations in the second level. The two-level system shows significant improvements over a baseline SMT system. It also produces longer matching phrases due to the generalization introduced by tagging NPs.

1. Introduction

When using statistical machine translation (SMT) systems, we often notice that the phrases used to construct the translations are rather short. On average these phrases are less than two words long. This is in spite of that fact that some phrase extraction methods allow the extraction of arbitrarily long phrases. The main reason for this behavior is data sparseness; long exact matching phrases are relatively rare in the training data. In the decoder, these phrases have to compete with abundant shorter phrases. Due to this reason, Koehn et al. (2003) find that phrases longer than three words give little performance improvement. However, with limited reordering strategies used in most of the statistical machines translation systems, a combination of small short phrases does not always generate the desired translation. Zhang (2005) shows improved translation performance by using phrases of arbitrary length. Hierarchical models, such as the Hiero system (Chiang, 2005), that uses phrases with words as well as subphrases have shown better performance than standard phrase based systems.

In this paper, we investigate a simplified two-level machine translations system that uses a linguistically motivated phrase decomposition. We think noun phrases (NPs) are good candidates for a hierarchical system. Semantically noun phrases describe objects and concepts using one or more nouns and adjectives. The vast majority of words in a language are nouns and hence NPs appear frequently in sentences. Noun phrases can often be translated independently into other languages irrespective of the context they appear. They tend to appear as coherent units in many languages. When using NPs as the unit of decomposition, we force it to be translated as an NP in the target language. Although this might not always be the best choice, as Koehn (2003) shows, it is not a harmful restriction.

We integrate the two-level phrases into a phrase based SMT decoder with minor modifications. It involves the following steps:

- Identify NPs on both sides of the parallel training corpus and generate an NP translation table
- Tag NPs in the training corpus by replacing them with a special tag "@NP"
- Extract phrase translation pairs from the NP-tagged corpus
- Use the extracted NP translation table and NP-tagged phrases in a two-level decoder to translate new sentences.

In the next chapter we describe each of the above steps in more detail. We then present the experimental results and our conclusions.

2. System

We build a translations system that translates Arabic text into English. Our training data consists of parallel corpora primarily of newswire genre available from LDC. Table 1 shows the statistics for the training data after it was pre-processed and English side lower-cased.

2.1 Generating NP translations

The first step is to identify noun phrases in the training corpus. Essentially, we want to identify corresponding NP pairs in Arabic and English sides, and build an NP translation table. To achieve this we use a parser to extract NPs from one side of the text and a word-alignment model to induce the corresponding NPs on the other side of the parallel text.

حملة عسكرية # a military campaign
اتفاق تعاون عسكري # a military cooperation protocol
حكومة وحدة وطنية # a national unity government
بعثة جديدة للامم المتحدة # a new united nations mission
افاد مراسل وكالة فرانس برس # agence france presse correspondent
المنتجات الزراعية والغذائية # agricultural and food products

Figure 1: Sample of NP translation table

As our system translates Arabic text into English, it would be logical to start with Arabic; parsing Arabic side and extracting corresponding English NP translations. However, the Arabic parsers available did not produce desired accuracy. Therefore we use Charniak's parser (Charniak, 2000) to parse English side of the training data. From the resulting parse trees we extract base NPs; i.e. NPs that do not contain other NPs embedded in them. As mentioned in the previous section these NPs are fairly short and are good candidates for a hierarchical system.

	Arabic	English
Sentences	135K	135K
Tokens	3.5M	4.3M
Vocabulary	145K	63K

Table 1: Training data statistics

We generate IBM model 3 (Brown et al., 1993) alignments by running GIZA++ (Och and Ney, 2003) with the parallel text. GIZA++ training is done for both directions and the word alignments are generated by the intersection of the two.

For each English NP, we search the aligned corpus for sentences that contain the NP and read off the alignment as its translation. To compensate for alignment errors we also include partial alignments as follows: We find maximum (*max*) and minimum (*min*) Arabic word indices that are aligned to the words in the English NP. All the Arabic words between *min* and *max* are considered to be the translation of the English NP.

We filter out unbalance NP translation pairs by removing entries that have a length ratio (# Arabic words / # English words) over two. Table 2 shows the details of extracted NPs. As seen in the table, translations for some English NPs are not found due to alignment errors.

English NPs	325K
Translations found	260K
After filtering	205K

Table 2: Extracted NP statistics

A sample of the extracted NP translation table is shown in Figure 1. Each line contains an Arabic NP and its English equivalent separated by a hash mark. An NP in the table may contain multiple translation alternatives.

Table 3 gives the length distribution of extracted NP translations. This was calculated for Arabic and English sides independently. The average length of an Arabic NP is 2 words while the average length of an English NP is 3.

Length of NP (# words)	Arabic	English
1	31K	13K
2	75K	66K
3	52K	73K
4	22K	32K
5	13K	12K
> 5	12K	9K

Table 3: Length distribution of NPs

To evaluate the accuracy of NP translations, and also to estimate how often the Arabic translation of an English NP is indeed an NP, a sample of NP translations was evaluated by an Arabic native speaker. 90% of the resulting Arabic translations were NPs. Out of these NPs, 11% had some irregularities such as missing articles in one side, etc. 10% of Arabic translations were either incorrect or not NPs (e.g. verb phrases).

Figure 2: Tagging of NPs

نشرت في @NP # published in @NP
نشرت في @NP # published on @NP in
@NP افاد # @NP reported
@NP افاد # @NP stated
@NP افاد # @NP said
بالحصول من @NP علي @NP # to get @NP from @NP
بالحصول من @NP علي @NP # get @NP from @NP
واعلن @NP @NP ان @NP # @NP told @NP that @NP
@NP @NP نقلت @NP # @NP that transported @NP to @NP

Figure 3: Examples of extracted NP-tagged phrases

2.2 Tagging NPs in the Training Corpus

Once we have the NP translation table, the next step is to identify and tag NP pairs in the training corpus. From the parse tree, we already know NPs on the English side. We only have to identify NPs on the Arabic side. Although we can directly use the already generated word alignments for this purpose, we choose not to as alignments within individual sentences are less reliable. Instead we use the filtered NP translation table as follows:

For each NP in the English sentence, we look for its Arabic translation in the NP translation table. If any of the alternative translations is present in the Arabic sentence, we tag it as an NP, and replace both NPs with a special tag (@NP). If none of the translations are present in the Arabic sentence, we leave English NP untagged. Figure 2 illustrates this process. This is repeated for all the sentences to generate an NP-tagged training corpus.

How much contraction has the NP-tagging introduced? We looked into the number of unique n-grams in the corpus before and after NP-tagging. Tables 4 and 5 show the comparison for Arabic and English sides of the corpus respectively.

N-gram Type	Original	NP-tagged
unigram	145K	144K
bigram	1425K	1269K
trigrams	2540K	2299K
4-grams	2890K	2713K

Table 4: Unique N-grams in Arabic training corpus before and after NP-tagging

We see a considerable reduction in the types of n-grams.

N-gram Type	Original	NP-tagged
unigram	630K	628K
bigram	841K	770K
trigrams	2293K	2015K
4-grams	3242K	2894K

Table 5: Unique N-grams in English training corpus before and after NP-tagging

We also compared the average length of the corpus before and after NP-tagging. These numbers are given in Table 6. Avg. length of an Arabic sentence has dropped by about 1.3 words. For English sentences, the drop is about 2 words.

Corpus	Original	NP-tagged
Original	28.83	35.89
NP-tagged	27.56	33.40

Table 6: Avg. length of training corpus before and after NP-tagging

2.3 Extract Phrases from NP-tagged Corpus

We use the NP-tagged parallel corpus to extract phrase translation pairs. Our phrase extraction method is similar to Moore (2003) which is a variation of the IBM-1 word alignment model (Brown et al., 1993).

Assuming a source sentence $s_1^I = s_1 \ldots s_I$ in the bilingual corpus contains a phrase $s_{i_1}^{i_2} = s_{i_1} \ldots s_{i_2}$ we are interested in the sequence of words $t_{j_1}^{j_2} = t_{j_1} \ldots t_{j_2}$ from the respective target sentence $t_1^J = t_1 \ldots t_J$ that is the optimal translation for this source phrase. We can estimate the quality of a translation candidate by using the IBM-1 word alignment probabilities between the source and target phrases. If the candidate is actually a good translation of the source phrase we expect higher IBM-1 probabilities between the words in the phrases than if the translation candidate was incorrect.

If we assume that $t_{j_1}^{j_2}$ is the optimal translation for $s_{i_1}^{i_2}$ in this sentence pair we can analogously argue that the words from the sentence pair that are not in these phrases must also be translations of each other. This means the optimal translation for the (non-contiguous) source phrase $s_1 \ldots s_{i_1-1} \quad s_{i_2+1} \ldots s_I$ is $t_1 \ldots t_{j_1-1} \quad t_{j_2+1} \ldots t_J$ and we also expect high probabilities between the words in these two phrases.

Overall the constrained probability for this sentence split can be calculated as:

$$p_{j_1,j_2}(s \mid t) = \prod_{i=1}^{i_1-1} \sum_{j \notin (j_1 \ldots j_2)} p(s_i \mid t_j) \prod_{i=i_1}^{i_2} \sum_{j=j_1}^{j_2} p(s_i \mid t_j) \prod_{i=i_2+1}^{I} \sum_{j \notin (j_1 \ldots j_2)} p(s_i \mid t_j)$$

If we optimize over the target side boundaries j_1 and j_2 we can determine the optimal sentence splitting and the best translation candidate.

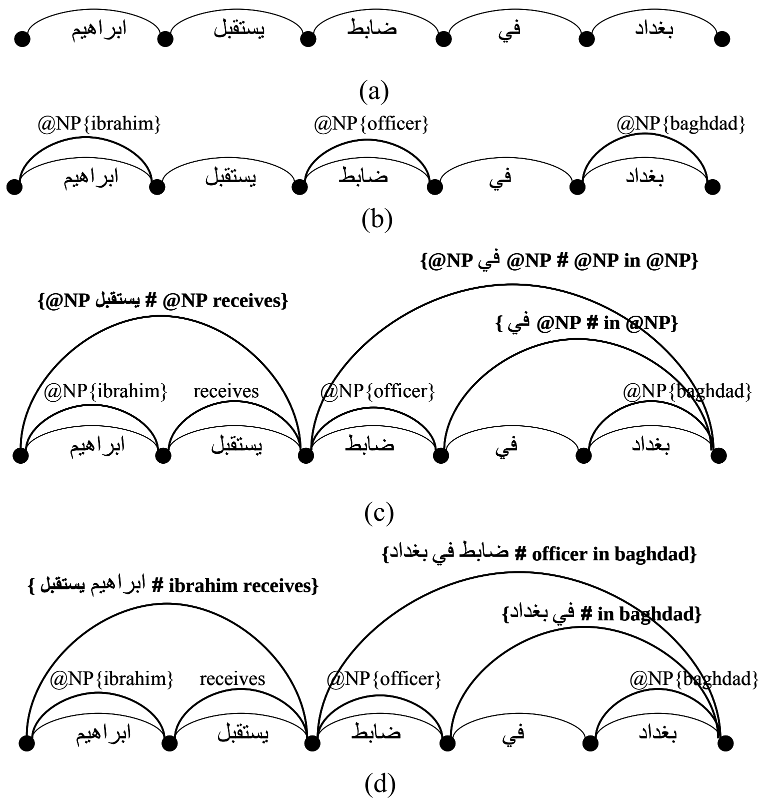

Figure 4: Two level decoding process

The same ideas can be applied if we use the IBM-1 probabilities for the reverse direction thus calculating $p_{j_1 j_2}(t \mid s)$ and we interpolate the two phrase alignment probabilities to get the optimal translation candidate. We not only use the top translation candidate but all candidates up to a certain threshold.

A more detailed description of the method is given in Vogel (2005).

A sample of NP-tagged phrases we extracted using the above method are given in Figure 3. Each line contains an Arabic phrase and its English translation separated by a hash mark. We have removed the scores attached to each phrase for clarity.

Some of the phrases extracted by the above method contain different number of NPs on Arabic and English sides. These phrases are against our assumption that Arabic NPs are translated into English NPs. Therefore we remove them from the phrase table before using in the decoder.

2.4 Two-level Decoding

The decoder combines different knowledge sources including the translation model and the language model, to generate the best translation for a sentence. We use CMU STTK decoder (Vogel et al., 2003). For our experiments, the decoder uses two translation resources in two levels to generate a hierarchy of phrases. NP translation table is used in the first level to identify possible NPs in the test sentence. NP-tagged phrase table is then used in the next level to build a translation lattice. The decoding process is organized into two steps:

1. Build a translation lattice using all available word/phrase translation resources
2. Find the best combination of partial translations by searching through the lattice

In addition, it also performs minimum error-rate training (Och, 2003) to find the best scaling factors for each model used in the decoder.

2.4.1 Building the Translation Lattice

The first step in decoding is building the translation lattice. We illustrate this process by using the following Arabic sentence. Note that the Arabic sentence is written from right-to-left.

Arabic sentence: *ابراهيم يستقبل ضابط في بغداد*
Reference translation: *Ibrahim receives officer in Baghdad*

First the decoder converts the Arabic sentence into a lattice structure where words are attached to the edges (see figure 4a).

Next, for each word sequence starting from the left-most node, it searches the level 1 phrase table (i.e. NP translation table) for matching entries. If an entry is found, the source word sequence is identified as a possible noun phrase. A new edge is then added to the lattice across the corresponding nodes, along with a tag (@NP) and the English translation of the phrase. This process is repeated for every node, except the last one. In figure 4(b), three words have been identified and tagged as possible noun phrases.

The same process of searching for word sequences in the lattice is repeated with level 2 phrase table (i.e. NP-tagged phrases). This time word sequences can consist of actual Arabic words as well as NP tags. Whenever a matching entry is found, it is added to the lattice the same way as before. The process is repeated over the resultant lattice, effectively building a hierarchy of NP-tagged phrase structure.

As illustrated in figure 4(c), this process can add translation edges covering only Arabic words, Arabic words and NP tags, and even previously added NP-tagged phrases. However, in our current experiments we do not use the latter type of edges.

This method does not require the explicit tagging of Arabic test sentences for NPs and allows alternative phrases to compete in the decoder.

Finally the NP-tagged phrases are expanded using actual Arabic/English translation pairs. We make the assumption that the order in which the NPs appear in Arabic side and the English side are the same. i.e. First NP in Arabic side of the phrase corresponds to the first NP in the English side of the phrase, etc. However, this might not always be correct. To rectify this, a more elaborate decoding scheme is required which uses phrase table entries that encode the relationship between NP tags on Arabic and English sides. We plan to address this in the future.

Due to the hierarchical nature of the phrases where NP tags are replaced by all possible alternative translations, and expanded into difference hypotheses, the size of the lattice grows rapidly. As the sentence gets longer, the size of the lattice grows exponentially. To keep the lattice size small and keep the decoding time within reasonable bounds, we currently employ strict pruning strategies that remove non-promising edges from the lattice.

2.4.2 Searching for the Best Path

The second stage in decoding is finding a best path through the translation lattice, now also applying the language model. The search algorithm is extended to allow for word reordering. Essentially, decoding is done from left to right over the Arabic sentence, but words can be skipped within a restricted reordering window (typically 4 words) and translated later. This in effect will reorder the words in the English sentence.

When a hypothesis is expanded, the language model is applied to all English words attached to the edge over which the hypothesis is expanded. In addition, the distortion model is applied, adding a cost depending on the distance of the jump made in the Arabic sentence. Hypotheses are recombined whenever the model cannot change the ranking of alternative hypotheses in the future.

As typically too many hypotheses are generated, pruning is necessary. Pruning is applied at two steps in the search algorithm: First, a hypothesis is stored only when it is within a threshold to the best hypothesis. Second, as the beam shifts whenever a new best hypothesis has been generated, pruning criterion is applied again before a hypothesis is expanded.

A more detailed description of the decoder is given in Vogel (2003).

3. Evaluation Results

We used two test sets from previous NIST evaluations as our test data. MT03 was used as the development set and MT05 was used as an unseen test set. To optimize the parameters of the decoder, we performed minimum error-rate training on MT03 optimizing for Bleu (Papineni et al., 2002) metric. Table 7 gives the details for the two test sets. Both test sets have 4 reference translations per test sentence.

	MT03	MT05
Sentences	663	1056
Tokens	16K	28K
Avg. Sentence Length	24.5	26.8

Table 7: Test set statistics

The baseline system uses the original non-NP-tagged corpus. The phrase extraction uses the same approach described in section 2.3, but no hierarchical phrases are generated while decoding. An n-gram suffix array language model (Zhang and Vogel, 2006) was used in the decoder for all the experiments which was trained using the English side of the training corpus. Baseline scores for the test sets are given in the first row in Table 8. Scores are reported in Bleu and METEOR (Banerjee and Lavie, 2005) metrics.

	MT03		MT05	
	Bleu%	MET	Bleu%	MET
Baseline	35.16	0.624	30.58	0.603
Baseline+ NPs	37.07	0.624	32.50	0.604
NP-tagged+NPs	37.88	0.620	34.62	0.605

Table 8: Translation results
(Both Bleu and MET are case insensitive scores)

First we wanted to see if the extracted NP translations can already help improve the translation quality of our baseline system. To evaluate this, we combined the NP translation table and the regular phrase table used in the baseline system. Here, NP translations are regarded as alternative translations to the baseline system. Since there is no hierarchy in the phrases, in the decoder all the phrases are used at the same level. However, we made the decoder biased towards phrases from the NP translation table, so that whenever there are alternatives, the decoder will favor an NP translation. Results for the combined phrases are given under "Baseline+NPs" in the second row of Table 8.

Both MT03 and MT05 test sets show improvements in Bleu metric over the baseline; 1.91 and 1.92 Bleu points respectively. Both improvements are statistically

1
بكين ترحب بالتعاون الدولي في صناعة السيارات
Beijing Welcomes International Cooperation in Auto Industry
- peking welcomes collaboration in the international auto industry
- peking welcomes international cooperation in auto industry .

2
الرياض تعلن اعتقال احد اكبر مهربي المخدرات في المملكة
Riyad Announces the Arrest of One of the Biggest Drugs Smugglers in the Kingdom
- riyadh announces arrest a key drug-traffickers in kingdom
- riyadh announces the arrest one of the biggest drug traffickers in the kingdom .

3
نائب رئيس مجلس الدولة الصيني يختتم زيارة لكوريا الجنوبية
The Vice-President of the Chinese Council of State Concludes his Visit to South Korea
- the chinese vice-president of the state council , conclude visit south korea
- the deputy president of the chinese state council ends a visit to south korea

4
كانت اثيوبيا واريتريا قد خاضتا حربا علي الحدود من 1998 الي 2000
Ethiopia and Eritrea waged a war over the borders between 1998 and 2000.
- the eritrea and ethiopia had fought war on border from 1998 to 2000 .
- ethiopia and eritrea had fought war on the borders of 1998 to 2000 .

Figure 5: Example translations

significant[1]. We do not see similar improvement for Meteor metric.

Table 9 gives the phrase pair statistics for the baseline phrase table and NP translation table that were used in the experiment. Both phrase tables have been sub-sampled for the specific test set for which it is used. As seen in Table 9, more than half of the phrase pairs in the NP translation table are new phrases that are not found in the baseline phrase table.

	MT03	MT05
Baseline Phrases	149,509	250,103
NP Translations	5399	7343
Only found in NP Translations	3138	4156

Table 9: Phrase table statistics

We also analysed how many of these NP translations were actually picked by the decoder to construct the final translations. 30% of all the phrases used were NP translations.

Next, we evaluated the two-level system by using NP translation table and NP-tagged phrases in the decoder. Further restrictions were introduced to reduce lattice size and decoding time, by removing all NP-tagged phrases that have more than two NP tags from the phrase table. Results of the two-level system are in the last row of Table 8.

The development set, MT03, shows an improvement of 2.72 Bleu points over the baseline. For MT05, the improvement is 4.04 Bleu points. Both improvements are statistically significant. Although we see an improvement in the two-level system over the combined phrases, the difference for MT03 is not statistically significant. For MT05, the corresponding improvement over 2 Bleu points is statistically significant. Here again, we do not see a significant difference in Meteor scores. We suspect this is due to the fact that we optimize our decoder towards the Bleu metric.

A high percentage (20%) of phrases used in the final translation was two-level phrases. A comparison of the average length of phrases (in terms of number of words) used to generate the 1-best translations is given in Table 10. For both test sets, the two-level system generates longer phrases than the baseline. This indicates that NP-tagging has reduced the variation among word sequences, effectively allowing longer phrase matches.

	MT03	MT05
Baseline	1.60	1.57
NP-tagged+NPs	1.82	1.78

Table 10: Avg. phrase length

This is also evident in the increase in average length of the translations generated by the two-level system. Table 11 gives a comparison.

	MT03	MT05
Baseline	28.12	30.57
Baseline+NPs	28.49	30.89
NP-tagged+NPs	29.42	32.20
References	29.85	34.24

Table 11: Avg. translation length

In Figure 5 we give some example translations to illustrate the effects of NP-tagging. Each block contains an Arabic sentence, its English reference translation (in bold face), and translations generated by the baseline and the two-level system, respectively.

[1] 95% confidence interval for Bleu metric for the Baseline system:
MT03: +/- 1.02
MT05: +/- 1.05

In example 1, the two-level system correctly generates the translation "*international cooperation in auto industry*" (using the noun phrases "*international cooperation*" and "*auto industry*" and combining them in NP-tagged phrase "NP in NP"). The baseline translation fails to preserve the meaning, possibly due to reordering. Examples 2 and 3 show the ability of the two-level system to produce proper articles and prepositions. This is especially important in translating Arabic sentences, as the articles are often attached to the nouns. Unless an explicit morphological analysis is performed, word alignments often fail to capture this. In example 4, however, the two-level system fails to generate the correct translation.

4. Conclusions and Future Work

In this paper we presented a noun phrase driven two-level statistical machine translation system. We evaluated the system by translating Arabic text into English. We first identify English noun phrases in the training data using a parser. Then we induced their Arabic translations using word alignment information. Identified noun phrase pairs in both sides were replaced by a noun phrase tag. This noun phrase tagged corpus was used to extract phrase translation pairs. Both NP translations and NP-tagged phrases were then used in a two-level translation decoder to translate new sentences. The system produced significantly better results over the baseline, for both development and test sets. Due to the generalization introduced by tagging NPs, the system was able to produce longer matching phrases.

We plan to extend this work in a number of ways. We currently extract only base NPs from the parse trees. We intend to include other embedded NPs in the future. Currently, we are working on modifying the decoder to prune non promising phrases early, so as to keep the size of the lattice small. This would allow us to relax the constraints on strict pruning of NP-tagged phrases, and thereby improve the results further. When expanding NP-tagged phrases, we currently assume that NPs on both sides are in the same order. We plan to facilitate arbitrary reordering of the NPs within the phrase in the future. We also plan to train the system on a larger training corpus.

Acknowledgements

The authors would like to thank Mohamed Normany and Anthony D'Auria for their support while building the system.

References

Satanjeev Banerjee and Alon Lavie. (2005). METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, Ann Arbor, MI, USA.

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

Eugene Charniak. (2000). A maximum-entropy inspired parser. In *Proceedings of HLT-NAACL 2000*, pages 132-139.

David Chiang. (2005). A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the ACL*, Ann Arbor, MI, USA. pages 263–270

Phillip Koehn, Franz Josef Och and Daniel Marcu. (2003). Statistical phrase-based translation. In *Proceedings of HLT-NAACL 2003*, pages 127-133.

Phillip Koehn. (2003). *Noun Phrase Translation*. Ph.D. thesis, University of Southern California.

Robert C. Moore. (2003). Learning translations of named-entity phrases from parallel corpora. In *Proceedings, 10th Conference of the European Chapter of the Association for Computational Linguistics*, Budapest, Hungary, pp. 259-266.

Franz Josef Och and Hermann Ney. (2003). A systematic comparison of various statistical alignment models, *Computational Linguistics*, 29(1):19-51.

Franz Josef Och. (2003). Minimum error rate training in statistical machine translation, In *Proceedings of the 41st Annual Meeting of the ACL*, Sapporo, Japan, pages 160-167.

Kishore Papineni, Salim. Roukos, Tod Ward, and WeiJing Zhu. (2002). Bleu: a method for automatic evaluation of machine translation. In *Procedings of the 40th Annual Meeting of the ACL*, Philadelphia, PA, USA. pages 311-318.

Stephan Vogel, Ying Zhang, Fei Huang, Alicia Tribble, Ashish Venugopal, Bing Zhao, Alex Waibel. (2003). The CMU statistical translation system. In *Proceedings of Machine Translation Summit IX*, New Orleans, LA, USA.

Stephan Vogel. (2003). SMT decoder dissected: word reordering. In *Proceedings of IEEE International Conference on Natural Language Processing and Knowledge Engineering 2003*, Beijing, China, pages 561-566

Stephan Vogel. (2005). PESA: Phrase pair extraction as sentence splitting. In *Proceedings of the Machine Translation Summit X*, Phuket, Thailand.

Ying Zhang and Stephan Vogel. (2005). An efficient phrase-to-phrase alignment model for arbitrarily long phrase and large corpora. In *Proceedings of the 10th Conference of the European Association for Machine Translation (EAMT-05)*, Budapest, Hungary.

Ying Zhang and Stephan Vogel. (2006). Suffix Array and its Applications in Empirical Natural Language Processing. *Technical Report* CMU-LTI-06-010, Language Technologies Institute, School of Computer Science, Carnegie Mellon University. Pittsburgh PA.