Estimating Phrase Pair Relevance for Translation Model Pruning

Matthias Eck, Stephan Vogel, and Alex Waibel

InterACT Research
Carnegie Mellon University
Pittsburgh, PA, 15213, USA
matteck@cs.cmu.edu, vogel+@cs.cmu.edu, waibel@cs.cmu.edu

Abstract

We present pruning strategies for translation models that are based on estimating the relevance of phrase pairs. We apply the overall translation system to a set of data and collect a number of statistics for each phrase pair. Using these statistics in various scoring terms we are able to significantly outperform baseline pruning methods and we can show that the number of phrase pairs can be reduced by up to 80% without significantly affecting the overall system performance.

1 Introduction

In a lot of circumstances a portable automatic speech-to-speech translation system could be very beneficial, for example for tourists, in military or medical relief situations. In these situations it is generally not convenient to carry a standard computer. Smaller devices like PDAs, handheld game consoles or even cell phones could offer many advantages here.

Speech-to-Speech translation on devices like this has been researched and products are starting to be deployed and to become commercially available (Hsiao et al., 2006; Isotani et al., 2003).

Porting a speech-to-speech translation system to a small device poses a unique challenge mainly due to the limited memory and computing power these devices offer.

For this reason most devices limit themselves to a certain domain (e.g. tourism, medical relief) but still tend to lack in performance compared to a standard computer.

This paper will discuss approaches that decrease the amount of memory the actual translation component needs without suffering significant performance losses.

1.1 Statistical Machine Translation

Statistical machine translation systems have recently often outperformed other translation approaches so it seems natural to also apply them in these scenarios.

A main component of every statistical machine translation system is the translation model. The translation model assigns translation probabilities to phrase pairs of source and target phrases extracted from a parallel bilingual text. These phrase pairs are applied during the decoding process and form a translation *lattice* (or word graph, Ueffing et al., 2002). Each possible path through this translation lattice is evaluated according to a number of models and the best path is chosen as the final translation (model-best path). The target sides of the phrase pairs in this path are combined to form the final translation.

Various algorithms for phrase pair extraction have been proposed (e.g. Koehn et al. 2003; Vogel, 2005; Zhao and Waibel, 2005).

Recent developments extract the phrase pairs from the bilingual data as needed depending on the actual test sentence (Callison-Burch et al., 2005; Zhang and Vogel, 2005). These techniques usually improve the performance, but need more computing power and memory compared to pre-extracting the phrase pairs so they will most likely not be used for small devices. For this reason we concentrated on translation models consisting of pre-extracted phrase pairs.

1.2 Phrase Pair Pruning

To limit the memory requirements of a translation system we now try to eliminate some of these phrase pairs (pruning of the translation model). The goal is to reduce the number of phrase pairs and in turn the memory requirement of the whole translation system, while not impacting the translation performance too heavily.

The approaches do not depend on the actual algorithm used to extract the phrase pairs and can be applied to every imaginable method that assigns probabilities to phrase pairs.

2 Previous Work

2.1 Threshold pruning

Related work on statistical machine translation mainly introduced two natural ideas to prune phrase pairs.

Probability threshold

The relevance of very low probability phrase pairs is limited as they will most likely not be chosen for the final translation path. The probability threshold pruning introduces a threshold and removes all phrase pairs with a lower probability.

Translation variety threshold

Some source phrases, especially ones that do not have a clear translation in the target language sometimes have a high number of possible translation candidates. The final translation path will have to choose one out of those. It seems natural to restrict the translation variety, especially if memory space is limited. The translation variety threshold imposes this limit. The pruning is accomplished by sorting the phrase pairs for each source phrase according to their probability and eliminating low probability ones until the threshold is reached.

Both threshold pruning strategies are well known. The Pharaoh decoder (Koehn, 2004) for example has an option to directly apply them to a phrase table.

2.2 Pruning via usage statistics

In Eck et al. (2007) we introduced a pruning strategy that utilizes usage statistics to eliminate phrase pairs. This pruning strategy was inspired by the Optimal Brain Damage algorithm (Le Cun et al., 1990) and collected statistics for phrase pairs by translating the whole training corpus with the originally extracted phrase pairs.
For each phrase pair two statistics were collected during this translation:

- *c*(phrase pair) = Count how often a phrase pair was considered during decoding (i.e. was added to the translation lattice)
- *u*(phrase pair) = Count how often a phrase pair was used in the final translation (i.e. in the chosen path through the lattice).

The reason to use those two statistics was that those occurrences influence the performance as they influence the decoding step by being added to the lattice or by being chosen for the final translation.
After empirical studies the overall score for a phrase pair was calculated as:

$$score(\text{phrase pair}) = [\log(c(\text{phrase pair}) + 1)] * [u(\text{phrase pair}) + 1]$$

The phrase pairs can then be sorted according to this score and the top *n* phrase pairs can be selected for a smaller phrase translation model.

The pruning methods introduced in this paper use a similar approach by also translating the whole training corpus but the scoring is based on a different idea and also uses other statistics.

The previous pruning approach will serve as a baseline but we will also compare the performance to the baseline introduced in Eck et al. (2007) that is based on the threshold pruning approaches in Section 2.1.

3 Translation Model Pruning

3.1 Model-best path Pruning

The fundamental idea of the first approach to translation model pruning is to estimate how likely it is that a phrase pair (pp) will be used in the first best path of an N-best list.
From a pure phrase pair perspective we can view each translation hypothesis in an N-best list as a number of phrase pairs that were applied to the source sentence to generate this hypothesis. This is illustrated in Figure 1.

1-best	pp(1,1)	pp(1,2)	...	pp(1,k_1)
2-best	pp(2,1)	pp(2,2)	...	pp(2,k_2)
3-best	pp(3,1)	pp(3,2)	...	pp(3,k_3)
...	...	...	...	...
i-best	pp(i,1)	pp(i,2)	...	pp(i,k_i)
...	...	...	...	...

Figure 1: Phrase pairs of an N-best list

The i-best hypothesis is generated by the k_i phrase pairs pp(i,1),…pp(i,k_i). Please note that these phrase pairs do not have to be distinct. It is possible that one translation path is generated by duplicate phrase pairs if the original source sentence contains repetitions. Especially neighboring hypotheses in the N-best list also often share a number of phrase pairs and might differ in only one of them.
The path that will finally be chosen by the decoder is the path that gets the overall best score by all applied models (translation model, language model,…) which is the 1-best or *model-best* path. All other paths are disregarded.
To get the same 1-best entry and therefore the same final translation hypothesis for this particular sentence it is only necessary to have the phrase pairs pp(1,1),…pp(1,k_1) in the translation model.
All other phrase pairs that occur in the N-best list could be eliminated without changing the final (*model-best*) translation path for this particular source sentence. These phrase pairs might however be used in the 1-best translation path of other sentences so they cannot simply be removed.
But we can try to estimate the probability that a phrase pair will be used in the 1-best translation path of any sentence by translating a large number of sentences and counting these occurrences.

The approach to estimate these probabilities is similar to the general approach presented in Eck et al. (2007), but instead of focusing on estimating the influence of phrase pairs on the overall decoding process we only focus on the 1-best path.

Based on a large number of translated sentences the probability of a phrase pair occurring in the 1-best path can be estimated as:

$$P(\text{pp in 1-best}) \approx \frac{\#\text{pp in 1-best}}{\#\text{chances}}$$

We do not divide by the number of sentences as a phrase pair might be used multiple times within one sentence, but instead by the number of theoretical chances it has to be used. This number is the same for all phrase pairs so it can be ignored for these purposes.
We use this estimation as a means to assign a score to each phrase pair that should approximate the relative probability that it will be used in a 1-best translation.

$$score(\text{pp}) = \#\text{pp in 1-best}$$

The phrase pairs can then be sorted according to this score and the top n phrase pairs can be selected for a smaller phrase translation model.
We did however notice that this score does not discriminate very well as only a small number of phrase pairs occur in the 1-best translation even if we translated a large number of sentences.

Considering the 1-best to 10-best translation paths
For this reason we not only considered the 1-best translation path but also the 2-best to 10-best translation paths. To limit the influence of the 2-best to 10-best translation paths these counts are divided by their index

($score_{A1}$) and in a second possibility the square of their index ($score_{A2}$). Overall this means we assume that a phrase pair that frequently occurs in the top 10 translation paths in the N-best list generally has a high probability to be in the model-best path.

$$score_{A1}(pp) = \sum_{i=1}^{10} \frac{\#\,pp\ in\ i\text{-}best}{i}$$

$$score_{A2}(pp) = \sum_{i=1}^{10} \frac{\#\,pp\ in\ i\text{-}best}{i^2}$$

Please note that we no longer explicitly use the number of times a phrase pair occurs in the lattice in these and the following scores as we did in Eck et al. (2007). We did however sort all phrase pairs that did not get a score assigned by the number of times they occurred in the lattices.

3.2 Metric-best path Pruning

Each N-best list contains a path (or a number of paths) that is the best path according to a scoring metric. To find this *metric-best* path a reference translation has to be available.
Figure 2 illustrates this situation with the i-best path as the metric-best path. All paths above and below the i-best path have a lower (or possibly equal) score according to this metric.

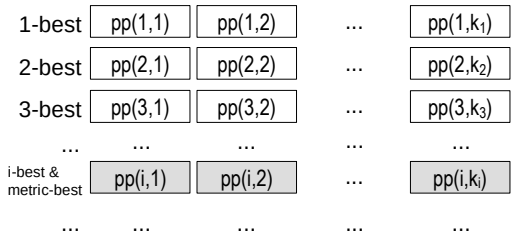

Figure 2: N-best list with metric-best path

A problem of the previous pruning approach is that potentially good phrase pairs in the metric-best path might actually be removed and no longer available for an unseen test sentence just because the models did not value them enough during the collection of the pruning statistics.
For this reason our second pruning approach considers the metric-best path. To avoid pruning of the phrase pairs in the metric-best path we apply the same statistics as in the previous section and define two additional scores for a phrase pair. Here we consider the counts for each phrase pair in the top 10 paths according to a scoring metric. In the experiments we used the edit distance between a translation hypothesis and a reference translation as the scoring metric.
The phrase pair scores $score_{B1}$ and $score_{B2}$ were defined analogously to section 3.1.

$$score_{B1}(pp) = \sum_{i=1}^{10} \frac{\#\,pp\ in\ metric\text{-}i\text{-}best}{i}$$

$$score_{B2}(pp) = \sum_{i=1}^{10} \frac{\#\,pp\ in\ metric\text{-}i\text{-}best}{i^2}$$

3.3 Pruning towards the Metric-best path

The scores defined in the previous section do not actually enforce the metric-best path, but just try to make sure that the phrase pairs within the metric-best path are not removed.
In this section we go one step further and intend to define a score for each phrase pair that actively removes phrase pairs that eliminate the metric-best path from being chosen as the final translation hypothesis (i.e. being the model-best path).

In Figure 2 the paths 1 to i-1 have a higher model score than the metric-best path at index i, while all paths with indices higher than i have a lower model score. If we could eliminate the paths 1 to i-1 the metric-best path would have the highest model score of all paths and become the translation hypothesis.
Removing one phrase pair from each of the 1 to i-1 paths from the translation model would be enough to eliminate these paths from the consideration.

After this pruning step the N-best list will contain new paths replacing these eliminated paths but their model scores will be lower than the model score of the metric-best path at index i. If these new paths would achieve a higher model score, the decoder could have chosen them during the original decoding step[1].
We define the following count to remove these unwanted phrase pairs.

$score_E(pp)$ = Number of times phrase pair pp occurs in a path that has a higher model score than the metric-best path while not occurring in the metric-best path.

In this case a high score would indicate the removal of a phrase pair. This score has to be used in combination with the other scores as it does not consider how often a phrase pair might actually occur in a metric-best path (and should not be pruned).

The potential problem with this score is that it might not be possible to clearly classify phrase pairs into ones that will probably occur in the metric-best path and the ones that will probably not occur in the metric-best path. A high number of phrase pairs could occur in both situations.
Another problem with the approach is that it might only be possible to eliminate some of the paths with better model scores than the metric-best path but not all of them. The remaining paths will still have better model scores but possibly an even lower metric score so the pruning would be counter-productive.

[1] It is possible that early elimination of possible translation hypotheses during the decoding process could change that statement. But this behavior can still be considered very unlikely.

4 Data and Experiments

4.1 Experimental Setup & Baselines

Translation system

All experiments were done with a state-of-the-art statistical machine translation system (Vogel, 2003; Eck et al., 2006). The system uses the phrase extraction method described in (Vogel, 2005) and a 6-gram language model (Zhang and Vogel, 2006).

Training and testing data

The training data for all experiments consisted of the Japanese-English BTEC corpus (Takezawa et al., 2002) with 162,318 lines of parallel text. The test set from the evaluation campaign of IWSLT 2004 (Akiba et al., 2004) was used as testing data. This data consists of 500 lines of tourism data.

The first experiments were done translating Japanese→English as in Eck et al. (2007) but we also validated the results with experiments translating English→Japanese.

16 English reference translations were available while there was only 1 reference on the Japanese side.

The language model was trained on the target side of the bilingual training data.

Extracted phrases

Extracting phrases for n-grams up to length 10 (with low frequency thresholds) resulted in 4,684,044 phrase pairs (273,459 distinct source phrases) for Japanese→English and 4,882,645 phrase pairs (453,201 distinct source phrases) for English→Japanese.

The translation models with all phrase pairs achieved baseline scores of 59.11 BLEU (Papineni et al., 2002) for Japanese→English and 17.04 BLEU for English→Japanese with 95% confidence intervals of [57.13, 61.09] and [16.59, 17.52] respectively.

Baseline pruning

The threshold pruning approaches served as a baseline. 8 different probability thresholds (0, 0.0001, 0.0005, 0.001, 0.005, 0.01, 0.05, 0.1) and 14 variety thresholds were used (1, 2, 3, 4, 5, 6, 8, 10, 15, 20, 50, 100, 200, 500).

It is usual practice to use not only one of these threshold approaches but to combine both of them, e.g. use a probability threshold of 0.001 and simultaneously apply a variety threshold of 10. This usually gives better performance than relying on just one kind of threshold pruning.

For our baseline we used 112(=8*14) combinations of these threshold values.

These combinations give a variety of translation scores at different sizes. We therefore define the baseline score at a given size as the best score that was reached with equal or less phrase pairs.

The *previous best* pruning strategy is the method based on the usage statistics as in section 2.2.

The baselines and previous best pruning results are illustrated in Figure 3 und Figure 4 for the translation directions Japanese→English and English→Japanese respectively.

The baseline scores show a relatively similar behavior for both directions. The English→Japanese threshold pruning baseline has a very good performance for higher numbers of phrase pairs but drops slightly faster for smaller sizes. The previous best approach especially outperforms both baselines for these smaller sizes.

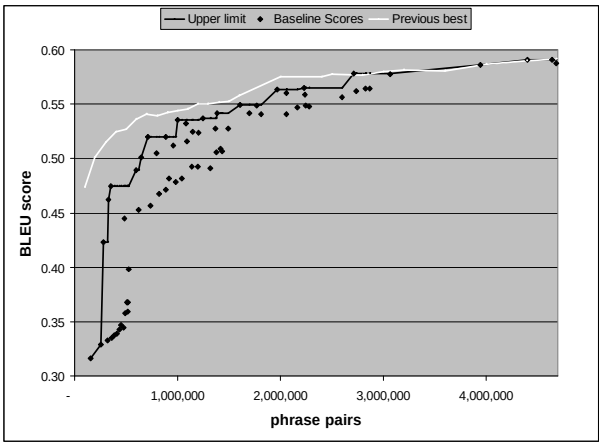

Figure 3: Baseline scores Japanese→English

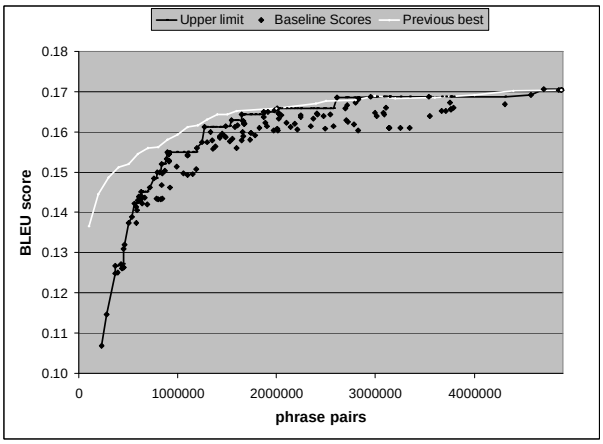

Figure 4: Baseline scores English→Japanese

5 Experimental Results

The presented pruning strategies were first investigated for the Japanese→English translation system to find the best pruning method and then applied to the English→Japanese system to validate the results and do further experiments.

5.1 Investigating Pruning Strategies

To apply the presented pruning strategies, the translation system was used to translate the 162,318 lines of Japanese training data. For each sentence a 1000-best list was generated and the statistics were collected.

Model-best path Pruning

The model-best path pruning counts the number of occurrences of each phrase pair in the 10-best lists and assigns scores to each phrase pair according to the scoring terms A1 and A2 presented in section 3.1. The results in

Table 1 show that both scoring terms clearly outperform the baseline and the previous best pruning approach. The scoring term A2 has a very small advantage over the scoring term A1. In both cases the score at 800,000 phrase pairs is already within the 95% confidence interval.

# phrase pairs	**Baseline**	**Previous best**	***A1***	***A2***
100,000	-	0.4735	0.4792	0.4909
200,000	0.3162	0.5008	0.5306	0.5388
400,000	0.4743	0.5241	0.5596	0.5576
800,000	0.5194	0.5394	0.5747	0.5748
1,200,000	0.5366	0.5498	0.5788	0.5790

Table 1: Results for scoring terms A1 and A2

Metric-best path Pruning

The metric-best path pruning applies the same ideas but uses the number of occurrences in the top 10 metric-best paths to calculate the phrase pair scores (section 3.2). Comparing these results in Table 2 with the scores of the model-best path pruning in Table 1 we see only minor differences. In this case B1 performs slightly better than B2.

# phrase pairs	**Baseline**	**Previous best**	***B1***	***B2***
100,000	-	0.4735	0.4822	0.4759
200,000	0.3162	0.5008	0.5300	0.5266
400,000	0.4743	0.5241	0.5610	0.5572
800,000	0.5194	0.5394	0.5753	0.5654
1,200,000	0.5366	0.5498	0.5787	0.5707

Table 2: Results for scoring terms B1 and B2

Score combination

As both pruning approaches separately performed very well compared to the baseline and the previous best pruning we intended to combine the benefits of both ideas by defining new scores as the sum of these scores.

Table 3 shows the results. Unfortunately none of the combinations shows a significant difference to the single results.

# phrase pairs	***A1+B1***	***A1+B2***	***A2+B1***	***A2+B2***
100,000	0.4786	0.4852	0.4883	0.4909
200,000	0.5306	0.5340	0.5395	0.5386
400,000	0.5596	0.5597	0.5585	0.5562
800,000	0.5747	0.5752	0.5747	0.5751
1,200,000	0.5791	0.5791	0.5791	0.5790

Table 3: Results for Score combination

The main reason for this behavior is that for most sentences in the training data the model-best translation is also the metric-best translation (or they are very close). This would be unusual for regular unseen test data, but in this case the phrases were originally extracted from exactly this data.

This means the model-best statistics will only differ slightly from the metric-best statistics and that resulted in those very similar scores in Table 1 and Table 2. The combination score will then not considerably change the order of phrase pairs, which we also found when analyzing the phrase pair order in these cases.

Overall scoring term A2 seems to be the best choice. It gives consistently good results without using the metric-best path information.

Pruning towards the metric-best path

To test the last approach we suggested in section 3.3 we used the additional statistics $score_E$. As described in section 3.3 this statistics aims to find phrase pairs that occur in a path that gets a higher model score than the metric-best path, hereby preventing the metric-best path from being chosen.

To eliminate these phrase pairs we subtracted this score from the score A2. Just subtracting it gave low scores so we limited its influence by adding factors of 0.1, 0.01 and 0.001. The results in Table 4 show that factor 0.01 seems to have slight advantages for lower number of phrase pairs, but we can generally state that this additional score does not significantly improve the overall performance.

# phrase pairs	***A2 -0.1E***	***A2 -0.01E***	***A2 -0.001E***	***A2***
100,000	0.4841	0.4911	0.4909	0.4909
200,000	0.5371	0.5392	0.5386	0.5388
400,000	0.5536	0.5565	0.5576	0.5576
800,000	0.5711	0.5740	0.5747	0.5748
1,200,000	0.5777	0.5792	0.5792	0.5790

Table 4: Results when incorporating scoring term E

Amount of data used to estimate statistics

It can be relatively tedious to translate all 162,318 lines of data to collect the statistics for the pruning so we wanted to see how the pruning is affected if we used a smaller amount of lines to estimate the statistics. Table 5 compares the results when using 40,000 and 80,000 lines with the translation of the full training data at 162,318 lines. In both cases the results drop very significantly. The additional data in the full training corpus helps tremendously. It will generally provide better statistics for common phrase pairs.

# phrase pairs	***A2 on 40k***	***A2 on 80k***	***A2 on 162k***
100,000	0.4032	0.4159	0.4909
200,000	0.4377	0.4611	0.5386
400,000	0.4488	0.4896	0.5576
800,000	0.4661	0.5072	0.5747
1,200,000	0.4854	0.5204	0.5792

Table 5: Results with different data sizes to estimate the statistics

Conclusion

We can overall say that the new results showed very significant improvements compared to the baseline and the previous best pruning results. Scores using the top 10 model-best or the top 10 metric-best paths showed very similar results and the scores did not improve when combining both statistics.

Also the approach to actively eliminate phrase pairs does not show additional consistent improvements over the earlier results.

It is definitely advisable to use the full bilingual corpus to extract the statistics. The scores dropped significantly if only part of the data was used.
An open question is if it might be beneficial to use additional bilingual in-domain texts to collect the pruning statistics that are not part of the training data.
Preliminary results however showed that it is almost always better to use the bilingual data as additional training data for the phrase pair extraction compared to only using it to collect pruning statistics.

For all practical purposes scoring term A2 seems to be a good choice. Figure 5 compares the results for this term with the baseline and previous best results.

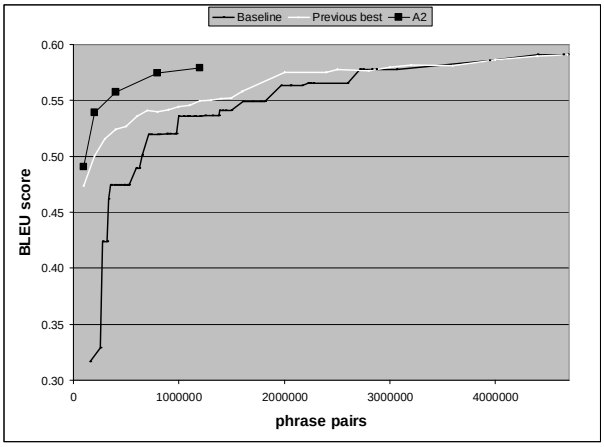

Figure 5: Performance for scoring term A2

5.2 Experiments on English→Japanese

Validation of results
For validation purposes and additional experiments we applied the pruning strategies on translations from English to Japanese. To collect the statistics the English part of the bilingual training data (162,318 lines) was translated to Japanese using the originally extracted phrase pairs.
Scoring term A2 also showed significant improvements over the baseline and previous best pruning strategies as illustrated in Figure 6.

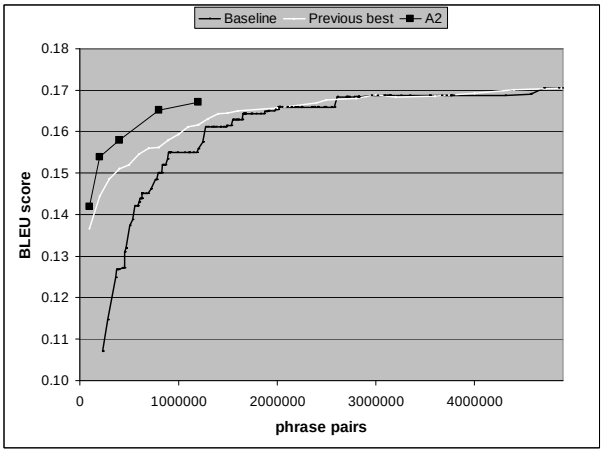

Figure 6: Scoring term A2 on English→Japanese

Table 6 gives some scores for these results. Contrary to the Japanese→English translations the score for 800,000 phrase pairs (pruned according to scoring term A2) just falls short of being within the 95% confidence interval but the score for 1.2 million phrase pairs (25% of the original number) is within the confidence interval.

# phrase pairs	Baseline	Previous best	*A2*
100,000	-	0.1366	0.1421
200,000	0.1011	0.1444	0.1540
400,000	0.1250	0.1512	0.1581
800,000	0.1500	0.1562	0.1652
1,200,000	0.1559	0.1616	0.1672

Table 6: Results for English→Japanese

Influence of additional data
For the last experiment we tested how additional out-of-domain data affected the performance of the pruning.
For this purpose we also used the translation system to translate 40,000 lines of English medical dialog data in addition to the bilingual data to estimate the statistics.
The style of the medical dialog data is not very different from the BTEC data, but the topics are obviously out-of-domain.
The results in Table 7 show that this additional data did not help the pruning performance significantly and some of the numbers are actually slightly lower.

# phrase pairs	*A2*	*A2 +40k medical*
100,000	0.1421	0.1415
200,000	0.1540	0.1535
400,000	0.1581	0.1573
800,000	0.1652	0.1655
1,200,000	0.1672	0.1671

Table 7: Using additional data to estimate statistics

It could be necessary to have a significant amount of additional in-domain data available to further improve these statistics but this was not available to us at this time.

6 Conclusions & Future Work

The proposed pruning approaches show nice improvements over our previous work and a strong baseline. It is possible to remove up to 80% of phrase pairs while not significantly affecting the translation performance.
Using the metric-best path information did not provide additional benefits compared to the model-best paths. It was also not notably valuable to try to enforce the metric-best path by eliminating certain phrase pairs that might prevent this path from being chosen.
These approaches could be easily applied to language models as well, the second major part of a translation system. For a language model the unit that statistics have to be collected for would be the individual n-gram.

7 Acknowledgements

This work was partly supported by the US DARPA under the programs GALE and TRANSTAC.

References

Yasuhiro Akiba, Marcello Federico, Noriko Kando, Hiromi Nakaiwa, Michael Paul, and Jun'ichi Tsujii (2004). Overview of the IWSLT04 Evaluation Campaign. Proceedings of IWSLT 2004, Kyoto, Japan.

Chris Callison-Burch, Colin Bannard, and Josh Schroeder (2005). Scaling Phrase-Based Statistical Machine Translation to Larger Corpora and Longer Phrases. Proceedings of ACL 2005, Ann Arbor, MI, USA.

Yann Le Cun, John S. Denker, and Sara A. Solla (1990). Optimal brain damage. In Advances in Neural Information Processing Systems 2, pages 598-605. Morgan Kaufmann, 1990.

Matthias Eck, Ian Lane, Nguyen Bach, Sanjika Hewavitharana, Muntsin Kolss, Bing Zhao, Almut Silja Hildebrand, Stephan Vogel, and Alex Waibel (2006). The UKA/CMU Statistical Machine Translation System for IWSLT 2006. Proceedings of IWSLT 2006, Kyoto, Japan.

Matthias Eck, Stephan Vogel, and Alex Waibel (2007). Translation Model Pruning via Usage Statistics for Statistical Machine Translation. Proceedings of NAACL-HLT 2007. Rochester, NY, USA.

Roger Hsiao, Ashish Venugopal, Thilo Köhler, Ying Zhang, Paisarn Charoenpornsawat, Andreas Zollmann, Stephan Vogel, Alan W. Black, Tanja Schultz, and Alex Waibel (2006). Optimizing Components for Handheld Two-way Speech Translation for an English-Iraqi Arabic System. Proceedings of ICSLP 2006, Pittsburgh, PA, USA.

Ryosuke Isotani, Kyoshi Yamabana, Shinichi Ando, Ken Hanazawa, Shin-ya Ishikawa and Ken-ichi Iso (2003). Speech-to-speech translation software on PDAs for travel conversation. NEC research & development, Tokyo, Japan.

Philipp Koehn (2004). A Beam Search Decoder for Statistical Machine Translation Models. Proceedings of AMTA 2004, Baltimore, MD, USA.

Philipp Koehn, Franz Josef Och, and Daniel Marcu (2003). Statistical Phrase-Based Translation. Proceedings of HLT/NAACL 2003, Edmonton, Canada.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu (2002). BLEU: a Method for Automatic Evaluation of Machine Translation. Proceedings of ACL 2002, Philadelphia, PA, USA.

Toshiyuki Takezawa, Eiichiro Sumita, Fumiaki Sugaya, Hirofumi Yamamoto, and Seiichi Yamamoto (2002). Toward a Broad-coverage Bilingual Corpus for Speech Translation of Travel Conversation in the Real World. Proceedings of LREC 2002, Las Palmas, Spain.

Nicola Ueffing, Franz J. Och, Hermann Ney (2002). Generation of Word Graphs in Statistical Machine Translation. Proceedings of EMNLP 2002, Philadelphia, PA, USA.

Stephan Vogel (2003). SMT Decoder Dissected: Word Reordering. Proceedings of NLP-KE 2003, Beijing, China.

Stephan Vogel (2005). PESA: Phrase Pair Extraction as Sentence Splitting. Proceedings of MTSummit X, Phuket, Thailand.

Ying Zhang and Stephan Vogel (2005). An Efficient Phrase-to-Phrase Alignment Model for Arbitrarily Long Phrases and Large Corpora. Proceedings of EAMT 2005, Budapest, Hungary.

Ying Zhang and Stephan Vogel (2006). Suffix Array and its Application in Empirical Natural Language Processing. Technical Report CMU-LTI-06-010, Pittsburgh, PA, USA.

Bing Zhao and Alex Waibel (2005). Learning a Log-Linear Model with Bilingual Phrase-Pair Features for Statistical Machine Translation, Proceedings of the SigHan Workshop, Jeju, Korea.