# Edinburgh System Description for the 2005 IWSLT Speech Translation Evaluation

**Philipp Koehn, Amittai Axelrod, Alexandra Birch Mayne,**
**Chris Callison-Burch**, **Miles Osborne**, **David Talbot**
pkoehn@inf.ed.ac.uk, amittai@mit.edu
a.c.birch-mayne@sms.ed.ac.uk, chris@linearb.co.uk
miles@inf.ed.ac.uk, d.r.talbot@sms.ed.ac.uk
School of Informatics
University of Edinburgh
2 Buccleuch Place, Edinburgh EH8 9LW, Scotland, UK

## Abstract

Our participation in the IWSLT 2005 speech translation task is our first effort to work on limited domain speech data. We adapted our statistical machine translation system that performed successfully in previous DARPA competitions on open domain text translations. We participated in the supplied corpora transcription track. We achieved the highest BLEU score in 2 out of 5 language pairs and had competitive results for the other language pairs.

## 1 Introduction

The statistical machine translation group at the University of Edinburgh has been focused on open domain text translation, so we welcomed the challenge to work on the IWSLT 2005 limited domain speech translation task. We participated in the transcription translation tasks for all five language pairs, using only the supplied corpora.

Our MT system was originally developed for translation of European parliament texts from German to English (Koehn et al., 2003). We extended the system while working on the DARPA challenges to translate Chinese and Arabic news texts into English (Koehn, 2004a; Koehn et al., 2005). Now, we were faced with the challenge of speech data in mostly Asian languages.

The translation of transcribed speech differs in many ways from our traditional translation scenario: Much less training data is available, the domain is more limited, and the text style is very different — short questions and statements. In some respect, the task is easier, since smaller training corpora result in faster training times for the system. But it also meant that we had to re-examine various components of our system.

In this paper we present an overview of our current out-of-the-box system in the next section. It includes a more detailed treatment of models added over the last year, especially a novel lexicalised reordering model.

Experimental work went into the adaptation of our system to the IWSLT'05 translation tasks. This is described in Section 3. We used a Linux cluster of about 50 machines, which allowed extensive optimisation of key components of our system, especially word alignment, lexicalised reordering, and reordering limits.

Finally, we report on our results in the competition and some post-evaluation analysis.

## 2 System Description

The system employs a phrase-based statistical machine translation model (Koehn et al., 2003) that uses the Pharaoh decoder (Koehn, 2004b). In this section, we will give an overview of the system.

### 2.1 Phrase-Based Statistical MT

In phrase-based SMT models, the input (foreign) sentence is segmented into so-called phrases, which may be any sequences of adjacent words that do not have to be linguistically motivated. Each phrase is mapped into the target language (English). Phrases may be reordered. See Figure 1 for an illustration.

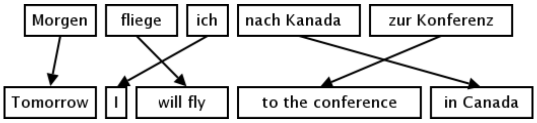

Figure 1: Phrase-based SMT: Input is segmented into phrases, each is mapped into output phrase and may be reordered

## 2.2 Log-linear Model

Mathematically, we employ a log linear approach in our translation system. We search for the most probable English sentence $\mathbf{e}$ given some foreign sentence $\mathbf{f}$ by maximising the sum over a set of feature functions $h_m(\mathbf{e}, \mathbf{f})$:

$$\begin{aligned} \hat{\mathbf{e}} &= \arg\max_e p(\mathbf{e}|\mathbf{f}) && (1) \\ &= \arg\max_e \sum_{m=1}^{M} \lambda_m h_m(\mathbf{e}, \mathbf{f}) && (2) \end{aligned}$$

The log linear model provides a natural framework to integrate many components and to weigh them according to their performance. We are using the following feature functions:

- language model
- phrase translation probability (both directions)
- lexical translation probability (both directions)
- word penalty
- phrase penalty
- linear reordering penalty
- lexicalised reordering

The language model is a smoothed trigram model trained on the target side training data.

The most important component of the system is the phrase translation table. We are extracting phrase pairs from the training corpus by first aligning the words in the corpus, extracting phrase pairs that are consistent with the word alignment, and then assigning probabilities (or scores) to the obtained phrase translations.

## 2.3 Word Alignment

Word alignments are obtained by first using the GIZA++ toolkit in both translation directions and then symmetrising the two alignments. Since the IBM Models implemented in GIZA++ are not able to map one target (English) word to multiple source (foreign) words, the method of symmetrising — called *refined method* (Och and Ney, 2003) — effectively overcomes this deficiency.

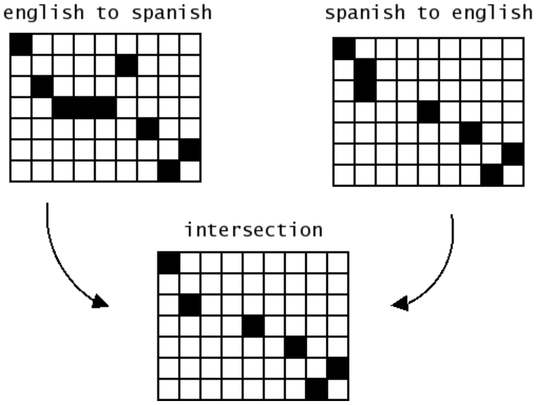

Figure 2: Obtaining a high precision, low recall word alignment by intersecting two GIZA++ alignments

Figure 2 shows the first step in the symmetrisation process: The intersection of the two GIZA++ alignments is taken. Only word alignment points that occur in both alignments are preserved. This is the **intersection** alignment.

In a second step, additional alignment points are added. Only alignment points that are in either of the two GIZA++ alignments (or, in the union of these alignments) are considered. In the growing step, potential alignment points that connect currently unaligned words and that neighbour established alignment points are added. Neighbouring can be either defined as directly to the left, right, top, or bottom (resulting in the **grow** alignment), or also include diagonally neighbourhood (resulting in the **grow-diag** alignment).

In a final step, alignment points that do not neighbour established alignment points are added. In a method called **grow(-diag)-final** this is done for alignment points between words, of which at least one is currently unaligned. In the **grow(-diag)-final-and** method, only alignment points between two unaligned words are added.

See Figure 3 for an illustration. The grey points in the matrix are potential alignment points that occur in the union, but not in the intersection of the two

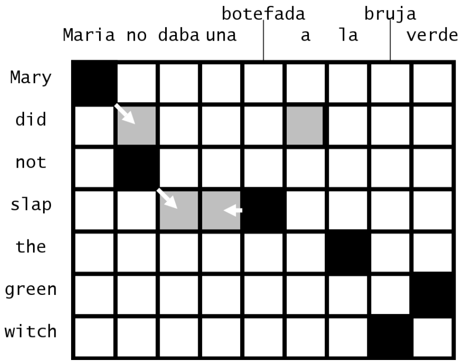

Figure 3: Adding additional alignment points. Potential points are points in the union of the two GIZA++ alignments (grey). In the growing step, neighbouring points are added, when they connect at least one unaligned word. In a final step outlying points may be added (see Section 2.3).

```
GROW-DIAG-FINAL(e2f,f2e):
  neighbouring = ((-1,0),(0,-1),(1,0),(0,1),
                  (-1,-1),(-1,1),(1,-1),(1,1))
  alignment = intersect(e2f,f2e);
  GROW-DIAG(); FINAL(e2f); FINAL(f2e);

GROW-DIAG():
  iterate until no new points added
    for english word e = 0 ... en
      for foreign word f = 0 ... fn
        if ( e aligned with f )
          for each neighbouring point ( e-new, f-new ):
            if ( ( e-new not aligned and f-new not aligned ) and
                 ( e-new, f-new ) in union( e2f, f2e ) )
              add alignment point ( e-new, f-new )
FINAL(a):
  for english word e-new = 0 ... en
    for foreign word f-new = 0 ... fn
      if ( ( e-new not aligned or f-new not aligned ) and
           ( e-new, f-new ) in alignment a )
        add alignment point ( e-new, f-new )
```

Figure 4: Pseudo-code of the **grow-diag-final** method to symmetrise word alignments. See Section 2.3 for variations of this method.

GIZA++ alignments. Three neighbouring points are added. The alignment point between *did* and *a* is added in the grow(-diag)-final method, but not in the grow(-diag)-final-and, since the Spanish word *a* is unaligned, but not the English word *did*. Figure 4 presents the symmetrisation method in pseudo code.

### 2.4 Phrase Extraction

We now extract phrase pairs for the phrase translation table. Any phrase pair that is consistent with the word alignment is collected. We define *consistent* as: The words in the phrase pair have to be aligned to each other and not to any words outside.

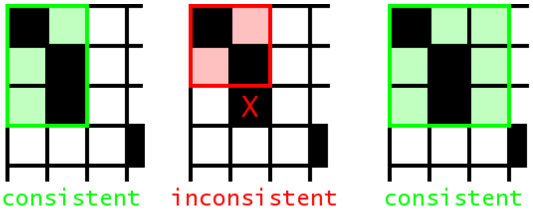

Figure 5: Definition of consistent word alignments: Words of an extracted phrase pair have to be aligned to each other and nothing else

See Figure 5 for an illustration. Note that unaligned words may be included within and at the border of extracted phrase pairs (third example in Figure 5). Each phrase pair, however, must include at least one alignment point.

Using word-level alignments to induce phrase-based translation models is common practise in the statistical machine translation community. It has been adopted by most groups participating in the NIST MT Evaluation (Lee and Przybocki, 2005).

In contrast to this, Marcu and Wong (2002) have defined a method for directly estimating phrasal translation models from parallel corpora, rather than using heuristic methods to induce phrase alignments from word alignments. Their joint probability phrase-based model is computationally demanding, and as such has not been applied to large data sets. Our group has been implementing a scalable version of the joint probability model (Mayne, 2005), and we hope to submit it as a contrastive system in next year's IWSLT.

### 2.5 Phrase Scoring

The phrase translation probability is defined as

$$p(\bar{f}|\bar{e}) = \frac{count(\bar{f},\bar{e})}{\sum_{\bar{f}} count(\bar{f},\bar{e})} \tag{3}$$

where $count(\bar{f},\bar{e})$ gives the total number of times the phrase $\bar{f}$ is aligned with the phrase $\bar{e}$ in the parallel corpus.

Phrase translation probabilities are lexically weighted as in (Koehn et al., 2003):

$$p_{lw}(\bar{f}|\bar{e},\mathbf{a}) = \prod_{i=1}^{n} \frac{1}{|\{i|(i,j) \in \mathbf{a}\}|} \sum_{\forall(i,j)\in\mathbf{a}} p(f_j|e_i) \tag{4}$$

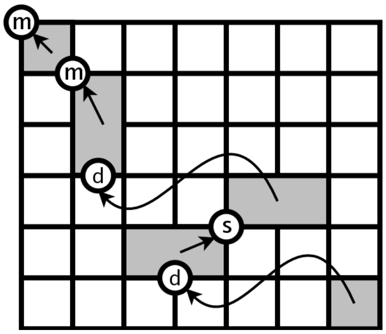

Figure 6: Possible orientations of phrases: monotone (m), swap (s), or discontinuous (d)

where $n$ is the length of $\bar{e}$, and $\mathbf{a}$ is the word-level alignment between phrase $\bar{e}$ and $\bar{f}$. Since a phrase alignment $< \bar{f}, \bar{e} >$ may have multiple possible word-level alignments, we retain a set of alignments and take the most frequent.

Word and phrase penalty add a constant factor ($\omega$ and $\pi$) for each word or phrase generated.

### 2.6 Reordering

Our original reordering model only considers the distance of movements. The reordering penalty adds a factor $\delta^n$ for movements over $n$ words. The movement distance is measured on the foreign side.

Our current system includes a lexicalised reordering model. For each phrase pair, we learn, how likely it directly follows a previous phrase (monotone), is swapped with a previous phrase (swap), or is not connected to the previous phrase at all (discontinuous). See Figure 6 for an illustration.

When collecting phrase pairs, can classify them into these three categories based on:

- monotone: a word alignment point to the top left exists
- swap: an alignment point to the top right exists
- discontinuous: no alignment points to the top left or top right

Given these counts, we can learn probability distributions of the form:

$$p_r(\text{orientation}|\bar{e}, \bar{f}) \tag{5}$$

For the estimation of the probability distribution, we smooth the collected counts.

This lexicalised reordering model is motivated by similar work by Tillmann (2004).

### 2.7 Discriminative Training

Recall that the components of our machine translation system are combined in a log-linear way. The weight of the feature functions, or model components, is set by minimum error rate training. We reimplemented a method suggested by Och (2003).

In short, we optimise the value of the parameter weights $\lambda_m$ by iteratively: (a) running the decoder with a currently best weight setting, (b) extracting an n-best list of possible translations, and (c) finding a better weight setting that re-ranks the n-best-list, so that a better translation score is obtained.

To score translation quality, we employ the BLEU score (Papineni et al., 2002). The search for the best weight setting is a line search for each $\lambda_m$, which is repeated until no improvement can be achieved.

We thank David Chiang of the University of Maryland for providing us with a faster version of our implementation.

## 3 Adaptations to IWSLT'05 Task

In a period of one month, we optimised our system to the IWSLT'05 task. We chose to only participate in the transcription task using the supplied data, since we did not have adequate additional resources or tools for these language pairs, and also had not enough time to investigate these.

The advantage of limiting ourselves to this track, meant that we could quickly train our system. Training the entire system (from corpus preparation over word alignment to building models) took only 15 minutes CPU time instead of about a week for the large-scale Arabic–English DARPA/NIST translation challenge. Hence, we were able to run many experiment to optimise performance.

We decided to use the 2003 test set as tuning set for minimum error rate training, and the 2004 test set as test set for development. All performance numbers reported in this section are %BLEU scores computed with our own evaluation script. This script takes as reference length the closest reference sentence length, as in the official evaluation, but does not eliminate punctuation, as done there.

In our experiments, we tried to find

- the best word alignment method
- the best lexicalised reordering method
- the best reordering distance limit

| | final (default) | final-and | grow-diag | grow | intersect |
|---|---|---|---|---|---|
| English words | 187,843 | 187,843 | 187,843 | 187,843 | 187,843 |
| Alignment points | 282,110 | 234,027 | 220,318 | 185,714 | 79,200 |
| Distinct phrase pairs | 61,168 | 270,654 | 447,550 | 854,680 | 2,561,715 |

Table 1: Different word alignment methods and the effect of the phrase table: Since alignment points restrict possible phrase pairs, fewer alignment points lead to larger phrase tables.

| Language Pair | final (default) | final-and | grow-diag | grow | intersect |
|---|---|---|---|---|---|
| Arabic-English | 48.8 | 48.5 | **49.9** | 39.9 | 47.5 |
| Japanese-English | 40.4 | 39.9 | 39.0 | 39.1 | **45.1** |
| Korean-English | 33.9 | **35.7** | 27.7 | 13.5 | 35.4 |
| Chinese-English | 28.9 | 32.4 | 31.7 | 32.8 | **34.6** |
| English-Chinese | **15.4** | 9.6 | 8.1 | **15.4** | 15.2 |

Table 2: BLEU scores for systems trained using different alignment methods

We also carried out experiments to optimise GIZA++ parameters, but this did not yield any significant improvements. We would like to re-visit these experiments at some future time, since we did not have sufficient time for a thorough treatment at this time.

We also tried to deal with language-specific problems, as previously done for German–English (Collins et al., 2005). We created hand-written rules that move the Japanese verb from the end of the sentence to the beginning. However, we could not consistently achieve improvements using these rules. Since we did not have a part-of-speech tagger for Japanese, we had to rely on the assumption that the last word of a Japanese sentence is the verb. We did not apply these rules in our official submission.

### 3.1 Optimising Word Alignment

Our experience with GIZA++ alignments has been that IBM Model training performs poorly for source words that occur only once in the training corpus. These words are often incorrectly aligned to many target words. This effect creates problems with phrase extraction, since alignment points effectively limit possible phrase pairs. If one word is aligned to many words that are spread throughout the sentence, many reasonable phrase pairs can not be extracted because of the consistency constraints of our phrase extraction algorithm.

Since we deal with much smaller data sets than we are used to, we expected to have more problems with singleton words and their adverse effect on phrase extraction. Hence, we explored a number of alignment methods, ranging from our default method (grow-diag-final), which establishes many word alignment points to the most sparse method of just allowing alignment points that occur in the intersection of the bidirectional alignments (intersect).

The effect of alignment method on the number of alignment points and the number of extracted phrase pairs is exemplified in Table 1 on the case of the Japanese–English training data. Note the differences between the default method and the intersection methods: The intersection only establishes a third of the number of alignment points (79,200 vs. 282,110), causing the number of extracted distinct phrase pairs to explode by a factor of about 40.

However, having a phrase table of 2.6 million distinct phrase pairs is not a computational problem for our system. In fact, for Arabic–English translation, we often work with phrase tables of up to 100 million distinct phrase pairs.

We carried out experiments using five different alignment methods for the different language pairs. For each alignment method and language pair, we trained a system and optimised it using minimum error rate training. Table 2 displays resulting %BLEU scores on the IWSLT'04 test set (using our BLEU scoring script described at the beginning of this section).

| Language Pair | Best Lexicalised Reordering | Word Alignment | Baseline | Improved |
|---|---|---|---|---|
| Arabic-English | orientation-bidirectional-fe | final-and | 49.9 | 50.9 |
| Japanese-English | orientation-fe | intersect | 45.1 | 47.6 |
| Korean-English | orientation-fe | intersect | 35.7 | 42.3 |
| Chinese-English | monotonicity-fe | intersect | 34.6 | 38.6 |
| English-Chinese | monotonicity-bidirectional-fe | grow-diag | 15.2 | 16.6 |

Table 3: Best lexicalised reordering methods, compared against the baseline (using only distance-based reordering penalty): Improvements for all language pairs

The evaluation of the effect of the different alignment methods on translation quality presents a mixed picture: While for all language pairs, the default method does not result in higher performance than the sparser methods, not a single alignment method emerges as the optimal method for all language pairs. For two language pairs, Japanese–English and Chinese–English, the intersection method comes out ahead.

### 3.2 Optimising Lexicalised Reordering

Since we just implemented lexicalised reordering in our system, we used the IWSLT'05 translation task as a testbed to investigate its best configuration. We consider the following choices in the lexicalised reordering model:

- Do we distinguish between monotone, swap, and discontinuous ordering (orientation), or just test for monotone ordering (monotonicity)?
- Do we condition on the identity of the foreign phrase (f), or on both the foreign and English phrase (fe)?
- Do we model reordering in respect to the previous translated phrase, or also in respect to the following translated phrase (bidirectional)?

These three different options lead to eight possible configurations for the lexicalised reordering model. We build translation systems for all possible configurations for all five language pairs. For all the language pairs, no single lexicalised reordering method emerged as significantly better than the others. However, any lexicalised reordering method is better than no lexicalised reordering.

In Table 3, you can see which alignment method scored best for each language pair. Again, a very mixed picture emerged. The only consistent result is that conditioning on the identity of both the foreign and English phrase is superior. Any of the remaining four possible configurations comes out ahead for at least one of the language pairs.

Since we optimised word alignment method and lexicalised reordering method in a integrated fashion, what is the best word alignment method changed for Arabic–English, Korean–English and Chinese–English.

We would like to stress again at this point that the differences are mostly not sufficiently significant to make a strong point here about which word alignment method or which lexicalised reordering method works best. However, we can clearly state that lexicalised reordering is beneficial for all language pairs.

### 3.3 Optimising Reordering Distance Limit

After settling on a word alignment and lexicalised reordering method for each language pair in previous experiments, we concluded our adaptation experiments by optimising the reordering distance limit.

Ideally, we would allow reordering of any distance, since movements over long distance do occur when translating. One example is the movement of the Japanese verb from the end of the sentence to the position at the beginning just after the subject in English.

However, our previous experience has shown that the reordering model is not strong enough to correctly guide long distance movements. In fact, when we completely prohibit movements over more than four words, we achieved better translation results than when allowing more distant reordering.

While our novel lexicalised reordering model has

| | keeping unknown words | | | | | | dropping unknown words | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Reordering Limit | 3 | 4 | 5 | 6 | 7 | 8 | 3 | 4 | 5 | 6 | 7 | 8 |
| Arabic-English | 50.3 | 50.4 | 50.1 | 50.6 | 50.0 | 50.1 | 56.3 | 56.2 | **56.7** | 56.5 | 55.9 | 56.2 |
| Japanese-English | 46.4 | 48.3 | 48.8 | 49.1 | 49.0 | 49.9 | 46.4 | 49.1 | 50.4 | 50.2 | **51.1** | 51.0 |
| Korean-English | 37.8 | 41.8 | 42.0 | 44.1 | 44.1 | **45.2** | 39.0 | 42.2 | 43.2 | 44.9 | 42.5 | 44.1 |
| Chinese-English | 36.8 | 36.7 | 37.2 | 37.5 | 36.9 | 37.2 | 39.3 | 40.0 | **40.2** | 39.8 | 39.6 | 39.8 |
| English-Chinese | 16.6 | 16.8 | 16.0 | 16.4 | 17.2 | 17.1 | 15.8 | 16.9 | 17.3 | 16.7 | **17.8** | 17.4 |

Table 4: Optimising the reordering limit (maximum word distance for phrase movement). The table also shows the effect of dropping unknown words instead of passing them to the output.

shown to be beneficial, it is still a very local model. Decisions are made for a particular phrase based on its empirical reordering behaviour with respect to directly neighbouring phrases. For instance, for a Japanese verb to be translated into English, we will learn that it is typically reordered, but not how far.

Nevertheless, we wanted to carry out experiments with larger reordering limits. Recall that reordering distance is measured in respect to movements of foreign phrases. If we first translate the first foreign word, and then continue with the fifth word, we measure this as a movement over three words (the three foreign words 2, 3, and 4 are skipped).

Table 4 displays the translation performance for systems with different reordering limits. Note that we did not have to retrain the models for these experiments, but we did have to optimise model weights using minimum error rate training.

The results suggest that more permissive reordering limits than a maximum movement distance of 4 words are beneficial. While being aware of the limited statistical significance of these results, we are inclined to cautiously state that for translations involving Asian languages, the maximum reordering limit of 8 (or even higher) seems to be better than the traditional 4.

## 4 Results

For the translation of the test data of the IWSLT'05 translation task, we used the optimised configuration and parameter settings, as obtained by our adaptation experiments.

The results are displayed in Table 5. Compared to the performance of the other participants, we are very satisfied with the results. We scored 1st place in two of the five tracks, and had very respectable showings for the other tracks.

A closer look at the numbers, however, will reveal one striking oddity: For almost all language pairs, we incur a heavy length penalty, which has a devastating effect on the NIST score. Obviously our output is almost always too short.

The culprit for this is our minimum error rate training, which optimises the BLEU score. It uses the *shortest* of the reference sentences as the basis to compute the length penalty. This inherently causes a optimisation to very short output. However, the official evaluation uses the *closest* reference length.

In a post-evaluation experiment, we altered our minimum error rate training to optimise to *average* reference sentence length. The effect on test scores is displayed in Table 6.

Due to the more lenient length penalty, our NIST score improve dramatically. In the case of Japanese–English, it more than doubles from 4.0784 to 8.1209. The effect on the BLEU scores is less pronounced: For four out of five language pairs, we achieved slightly higher BLEU scores, for Chinese–English, the BLEU score drops.

## 5 Conclusions

Our participation at the IWSLT'05 Evaluation Campaign seems to confirm one of the selling points of statistical machine translation: the ability to quickly build machine translation systems for new language pairs. While we had no prior experience with building systems for Korean and Japanese, and only very limited knowledge about any of the non-English languages, we were able to build competitive systems for all the language-pairs.

Our adaptation experiments revealed that translation tasks of speech transcriptions in limited domain,

| Language Pair | BLEU | NIST | WER | PER | METEOR | GTM | Rank |
|---|---|---|---|---|---|---|---|
| Arabic-English | 0.5105 (0.93) | 7.6382 (0.70) | 0.3902 | 0.3462 | 0.6893 | 0.6521 | 5th of 8 |
| Japanese-English | 0.3778 (0.81) | 4.0784 (0.41) | 0.5488 | 0.4861 | 0.5167 | 0.4748 | 4th of 7 |
| Korean-English | 0.3672 (0.88) | 5.6172 (0.60) | 0.5570 | 0.4797 | 0.5585 | 0.4843 | 1st of 4 |
| Chinese-English | 0.4650 (0.90) | 6.4922 (0.62) | 0.4535 | 0.3983 | 0.6320 | 0.5988 | 3rd of 10 |
| English-Chinese | 0.2127 (0.94) | 5.1807 (0.98) | 0.6197 | 0.5286 | 0.0955 | 0.5584 | 1st of 2 |

Table 5: Official Results: The scores for our official submission to the IWSLT'05 Evaluation Campaign (length penalty in parenthesis), and rank among participants according to the BLEU score.

| Language Pair | BLEU | NIST | WER | PER | METEOR | GTM |
|---|---|---|---|---|---|---|
| Arabic-English | 0.5180 (0.98) | 9.7749 (0.94) | 0.3860 | 0.3323 | 0.7270 | 0.6613 |
| Japanese-English | 0.3941 (0.95) | 8.1209 (0.91) | 0.5489 | 0.4599 | 0.5971 | 0.4890 |
| Korean-English | 0.3859 (1.00) | 8.4455 (0.99) | 0.5617 | 0.4559 | 0.6221 | 0.4980 |
| Chinese-English | 0.4364 (1.00) | 9.0834 (0.99) | 0.5043 | 0.4089 | 0.6841 | 0.5914 |
| English-Chinese | 0.2230 (0.91) | 5.2391 (0.97) | 0.6037 | 0.5149 | 0.0955 | 0.5657 |

Table 6: Optimisation to average reference sentence length instead of shortest reference length (length penalty in parenthesis): Note the improved length penalties and vastly improved NIST scores. 4 out of 5 BLEU scores are higher as well (exception is Chinese-English).

using small training corpus sizes, do require different settings of our translation system than we traditionally used for open domain text translation with much larger training corpora.

We also were able to verify the benefits of our novel lexicalised reordering model, which consistently led to significant perform gains.

## References

Collins, M., Koehn, P., and Kucerova, I. (2005). Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL '05)*, pages 531–540, Ann Arbor, Michigan. Association for Computational Linguistics.

Koehn, P. (2004a). The foundation for statistical machine translation at MIT. In *Proceedings of Machine Translation Evaluation Workshop 2004*.

Koehn, P. (2004b). Pharaoh: a beam search decoder for statistical machine translation. In *6th Conference of the Association for Machine Translation in the Americas, AMTA*, Lecture Notes in Computer Science. Springer.

Koehn, P., Axelrod, A., Mayne, A. B., Callison-Burch, C., Osborne, M., Talbot, D., and White, M. (2005). Edinburgh system description for the 2005 NIST MT evaluation. In *Proceedings of Machine Translation Evaluation Workshop 2005*.

Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase based translation. In *Proceedings of the Joint Conference on Human Language Technologies and the Annual Meeting of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL)*.

Lee, A. and Przybocki, M. (2005). NIST 2005 machine translation evaluation official results. Official release of automatic evaluation scores for all submissions.

Marcu, D. and Wong, W. (2002). A phrase-based, joint probability model for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Mayne, A. B. (2005). Scaling the joint probability phrase based statistical translation model. Master's thesis, University of Edinburgh.

Och, F. J. (2003). Minimum error rate training for statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association of Computational Linguistics (ACL)*.

Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–52.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association of Computational Linguistics (ACL)*.

Tillmann, C. (2004). A unigram orientation model for statistical machine translation. In *Proceedings of the Joint Conference on Human Language Technologies and the Annual Meeting of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL)*.