Combination of Machine Translation Systems via Hypothesis Selection from Combined N-Best Lists

Almut Silja Hildebrand and **Stephan Vogel**
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA 15213, USA
silja, vogel+@cs.cmu.edu

Abstract

Different approaches in machine translation achieve similar translation quality with a variety of translations in the output. Recently it has been shown, that it is possible to leverage the individual strengths of various systems and improve the overall translation quality by combining translation outputs. In this paper we present a method of hypothesis selection which is relatively simple compared to system combination methods which construct a synthesis of the input hypotheses. Our method uses information from n-best lists from several MT systems and features on the sentence level which are independent from the MT systems involved to improve the translation quality.

1 Introduction

In the field of machine translation, systems based on different principles for the generation of automatic translations such as phrase based, hierarchical, syntax based or example based translation have advanced to achieve similar translation quality. The different methods of machine translation lead to a variety of translation hypotheses for each of the source sentences.

Extensive work has been done on the topic of system combination for a number of years, for example the ROVER system (Fiscus, 1997), which combines the output of several speech recognition systems using a voting method on the word level. In machine translation, aligning the translation hypotheses to each other poses an additional problem because of the word reordering between the two respective languages. For example the MEMT system (Jayaraman and Lavie, 2005) and Sim et al. (2007) propose solutions to this problem.

Recently there have been a number of publications in this area, for example (Rosti et al., 2007) and (Huang and Papineni, 2007). Both of these approaches combine the system output on several levels: word, phrase and sentence level. Rosti et al. (2007) use several information sources such as the internal scores of the input systems, n-best lists and source-to-target phrase alignments to build three independent combination methods on all three levels. The best single combination method is the one on the word level, only outperformed by the combination of all three methods. Huang and Papineni (2007) also use information on all three levels from the input translation systems. They recalculate word lexicon costs, combine phrase tables, boost phrase pairs and reorderings used by the input systems. Then they re-decode a lattice constructed from source-target phrase pairs used by all the input systems during the first pass. Finally they apply an independent hypothesis selection step, which uses all original systems as well as the combined system as input.

Our method is very straight forward compared to the elaborate methods mentioned above, while still achieving comparable results. We simply combine n-best lists from all input systems and then select the best hypothesis according to several feature scores on a sentence to sentence basis. Our method is independent of internal translation system scores because those are usually not comparable. Besides the n-best list, no further information from the input systems is needed, which makes it possible to also include non-statistical translation systems in the combination.

In section 2 we describe the three types of features used in our combination: language model features, lexical features and n-best list based features. We optimize the feature weights for linear combination using MERT. We report our results on the large scale Chinese-English translation task combining six MT systems in section 3.

2 Features

All features described in this section are calculated based on the translation hypotheses only. We do not use any feature scores assigned to the hypotheses by the individual translation systems, but recalculate all feature scores in a consistent manner. In preliminary experiments we added the system scores to our features in the combination. This did not improve the combination result and in some cases even hurt the performance, probably because the scores and costs used by the individual systems are generally not comparable.

Using only system independent features enables our method to use the output of any translation system, no matter what method of generation was used there.

Because the strengths of individual systems might vary on the level of different genres as well as on a sentence by sentence basis, we also did not want to assign global weights to the individual systems.

The system independence of our features also leads to a robustness regarding varying performance of the individual systems on the different test sets used for tuning the feature weights and the unseen test data. For example the NIST test set from 2003 contains only newswire data, while the one from 2006 also contains weblog data, hence global system weights trained on MT03 might not perform well on MT06. It is also robust to incremental changes in an individual system between translation of the tuning and the testing data sets.

2.1 Language Models

To calculate language model scores, we use traditional n-gram language models with n-gram lengths of four and five. We calculate the score for each sentence in the n-best list by summing the log-probability for each word, given its history. We then normalize the sentence log-probability with the target sentence length to get an average word log-probability, which is comparable for translation hypotheses of different length.

2.2 Statistical Word Lexica

Brown et al. (1990) describe five statistical models for machine translation, the so-called IBM model 1 - IBM model 5. We use word lexica from either model 1 or model 4, which contain translation probabilities for source-target word pairs.

The statistical word to word translation lexicon allows to calculate the translation probability $P_{lex}(e)$ of each word e of the target sentence. $P_{lex}(e)$ is the sum of all translation probabilities of e for each word f_j from the source sentence f_1^J. This feature does not take word order or word alignment into account.

$$P_{lex}(e|f_1^J) = \frac{1}{J+1}\sum_{j=0}^{J} p(e|f_j) \qquad (1)$$

where f_1^J is the source sentence, J is the source sentence length, f_0 is the empty source word and $p(e|f_j)$ is the lexicon probability of the target word e, given one source word f_j.

Because the sum in equation 1 is dominated by the maximum lexicon probability as described in (Ueffing and Ney, 2007), we also use it as an additional feature:

$$P_{lex-max}(e|f_1^J) = max_{j=0,\ldots,J}\, p(e|f_j) \qquad (2)$$

For both lexicon score variants we calculate an average word translation probability as the sentence score, we sum over all words e_i in the target sentence and normalize with the target sentence length I.

From the word lexicon we also calculate the percentage of words, whose lexicon probability falls under a threshold. In one language direction it represents the fraction of source words that could not be translated and in the other direction it gives the fraction of target words that were generated from the empty word or were translated unreliably. This word deletion model was described and successfully applied in (Bender et al., 2004) and (Zens et al., 2005).

All three lexicon scores are calculated in both language directions. The source and the target sentence switch roles and the lexicon from the reverse language direction is used.

This results in six separate features per pair of statistical word lexica.

2.3 Position Dependent N-best List Word Agreement

The agreement score of a word e occurring in position i of the target sentence is calculated as the relative frequency of the N_k translation hypotheses in the n-best list for source sentence k containing word e at position i. It is the percentage of entries in the n-best list, which ”agrees” on a translation with e in position i. As described in (Ueffing and Ney, 2007) the relative frequency of e occurring in target position i in the n-best list is computed as:

$$h_k(e_i) = \frac{1}{N_k} \sum_{n=1}^{N_k} \delta(e_{n,i}, e) \qquad (3)$$

Here N_k is the number of entries in the n-best list for the corresponding source sentence k and $\delta(w_1, w_2) = 1$ if $w_1 = w_2$.

This feature tries to capture how many entries in the n-best list agree on not only the same word choice in the translation, but also the same word order. Since corresponding word positions might be shifted due to variations earlier in the sentence, we also use a word agreement score based on a window of size $i \pm t$ around position i. The agreement score is calculated accordingly:

$$h_k(e_i) = \frac{1}{N_k} \sum_{n=1}^{N_k} \delta(e_{n,i-t} \cdots e_{n,i+t}, e) \qquad (4)$$

where $\delta(w_{n,i-t} \cdots w_{n,i+t}, w) = 1$ if word w occurs in the window $w_{i-t} \cdots w_{i+t}$ of n-best list entry n.

The score for the whole hypothesis is the sum over all word agreement scores normalized by the sentence length.

We use window sizes for $t = 0$ to $t = 2$ as three separate features.

2.4 Position independent N-best List N-gram Agreement

The n-gram agreement score of each n-gram in the target sentence is the relative frequency of target sentences in the n-best list for one source sentence, that contain the n-gram $e_{i-(n-1)}...e_i$, independent from the position of the n-gram in the sentence.

$$h_k(e^i_{i-(n-1)}) = \frac{1}{N_k} \sum_{j=1}^{N_k} \delta(e^i_{i-(n-1)}, e^I_{1,j}) \qquad (5)$$

where $\delta(e^i_{i-(n-1)}, e^I_{1,j}) = 1$ if n-gram $e^i_{i-(n-1)}$ occurs in n-best list entry $e^I_{1,j}$.

This feature represents the percentage of the translation hypotheses, which contain the respective n-gram. If a hypothesis contains an n-gram more than once, it is only counted once, hence the maximum for h is 1.0 (100%). The score for the whole hypothesis is the sum over the word scores normalized by the sentence length.

We use n-gram lengths $n = 1..6$ as six separate features.

2.5 N-best List N-gram Probability

The n-best list n-gram probability is a traditional n-gram language model probability. The counts for the n-grams are collected on the n-best list entries for one source sentence only. No smoothing is applied, as the model is applied to the same n-best list it was trained on, hence the n-gram counts will never be zero. The n-gram probability for a target word e_i given its history $e^{i-1}_{i-(n-1)}$ is defined as

$$p(e_i|e^{i-1}_{i-(n-1)}) = \frac{C(e^i_{i-(n-1)})}{C(e^{i-1}_{i-(n-1)})} \qquad (6)$$

where $C(e^i_{i-(n-1)})$ is the count of the n-gram $e_{i-(n-1)}...e_i$ in all n-best list entries for the respective source sentence.

This feature set is derived from (Zens and Ney, 2006) with the difference that we use simple counts instead of fractional counts. This is because we want to be able to use this feature in cases where no posterior probabilities from the translation system are available.

The probability for the whole hypothesis is normalized by the hypothesis length to get an average word probability. We use n-gram lengths $n = 1..6$ as six separate features.

3 Experiments

3.1 Evaluation

In this paper we report results using BLEU (Papineni et al., 2002) and TER (Snover et al., 2005) metrics. In the MER training we optimize for maximum BLEU.

The Chinese to English test sets from the NIST MT evaluations in 2003 and 2006 were used as development and unseen test data. The MT03 test set contains 919 sentences of newswire data with four reference translations. From the 2006 NIST evaluation we used the text translation portion of the NIST part of the test data which consists of 1099 sentences with four references of newswire and weblog data. For each result reported in the following sections we used MER training to optimize the feature weights on a n-best list for MT03. These always contained the same set of systems and were combined under the same conditions as for the unseen data.

3.2 Models

In all of the following experiments we used two language models, six features from a pair of statistical word lexica, three features from the position dependent n-best list word agreement and six features each from the n-best list n-gram agreement as well as the n-best list n-gram probability, 23 features in total.

The language models were trained on the data from all sources in the English Gigaword Corpus V3, which contains several newspapers of the years between 1994 to 2006, observing the blackout dates for all NIST test sets. We also included the English side of the bilingual training data, resulting in a total of 2.7 billion running words after tokenization.

From these corpora we trained two language models. A 500 million word 4-gram LM from the bilingual data plus the data from the Chinese Xinhua News Agency and an interpolated 5-gram LM from the complete 2.7 giga word corpus.

We trained separate open vocabulary language models for each source and interpolated them using the SRI Language Modeling Toolkit (Stolcke, 2002). Held out data for the interpolation weights was comprised of one reference translation each from the Chinese MT03, MT04 and MT05 test sets. Table 1 shows the interpolation weights for the different sources. Apart from the English part of the bilingual data, the newswire data from the Chinese Xinhua News Agency and the Agence France Press have the largest weights. This reflects the makeup of the test data, which comes in large parts from these sources. Other sources, for example the UN parlamentary speeches or the New York Times, differ significantly in style and vocabulary from the test data and, therefore, get small weights.

xin 0.30	cna 0.06	nyt 0.03
bil 0.26	un 0.07	ltw 0.01
afp 0.21	apw 0.05	

Table 1: LM interpolation weights per source

The statistical word lexica were trained on the Chinese-English bilingual corpora relevant to GALE available through the LDC[1]. After sentence alignment and data cleaning these sources add up to 10.7 million sentences with 260 million running words on the English side. The lexica were trained with the GIZA++ toolkit (Och and Ney, 2003).

The research groups who provided the system outputs all had access to the same training data.

3.3 Systems

We used the output from six different Chinese-English machine translation systems trained on large data for the GALE and NIST evaluations in the beginning of 2008. They are based on phrase based, hierarchical and example based translation principles, trained on data with different Chinese word segmentations, built by three translation research groups, running four MT decoders. The systems A to F are ordered by their performance in BLEU on the unseen Chinese MT06 test set (see Table 2).

system	MT03	MT06 BLEU	MT06 TER
A	34.68	31.45	59.43
B	35.16	31.28	57.92
C	34.98	31.25	57.55
D	34.70	31.04	57.20
E	33.50	30.36	59.32
F	28.95	26.00	62.43

Table 2: Individual systems sorted by BLEU on unseen data

[1] http://projects.ldc.upenn.edu/gale/data/catalog.html

3.4 Feature Impact

For comparison to our set of 23 features we ran our setup with the two language models only as a simple baseline. In (Och et al., 2004) word lexicon features were described as the most useful features for n-best list re-scoring. Thus, we added those to the language model probabilities as a second baseline (LM+Lex). The results in Table 3 show that a system combination which uses these models alone can not improve over the BLEU score of 31.45 of system A. This probably is the case, because the statistical systems among the input systems are already using this type of information, and in fact share the training data which was used to build those models.

To explore the question which feature group contributes the most to the improvement in translation quality and to avoid testing all possible combinations of features, we removed one feature group at a time from the complete set. Table 3 shows that although adding the word lexicon features to the language models did not improve the result for the LM+Lex baseline, the overall result still drops slightly from 33.72 BLEU for all features to 33.61 BLEU for noLex. The combination result decreases insignificantly but consistently when removing any feature group.

features	MT03	MT06 BLEU / TER
LM only	35.13	31.17 / 59.34
LM+Lex	36.96	30.97 / 59.41
no LM	39.10	32.83 / 56.23
no Lex	39.62	33.61 / 56.88
no WordAgr	39.59	33.67 / 57.25
no NgrAgr	39.45	33.47 / 56.58
no NgrProb	39.69	33.65 / 57.40
LM+NgrAgr	39.45	33.58 / 57.15
all	39.76	33.72 / 56.79

Table 3: Impact of feature groups on the combination result

The biggest drops are caused by removing the language model (-0.89 for no LM) and the n-gram agreement (-0.25 for no NgrAgr) feature groups. Using only those feature groups which have the biggest impact brings the combination result up to 33.58 BLEU which is close to the best, but using all features still remains the best choice.

3.5 N-Best List Size

To find the optimal size for the n-best list combination, we compared the results of using list sizes from 1-best up to 1000-best for each individual system. Hasan et al. (2007) investigated the impact of n-best list size on the rescoring performance. They tested n-best list sizes up to 100 000 hypotheses. They found, that using more than 10,000 hypotheses does not help to improve the translation quality and that the difference between using 1000 and 10,000 hypotheses was very small. Based on their results we decided not to go beyond 1000-best.

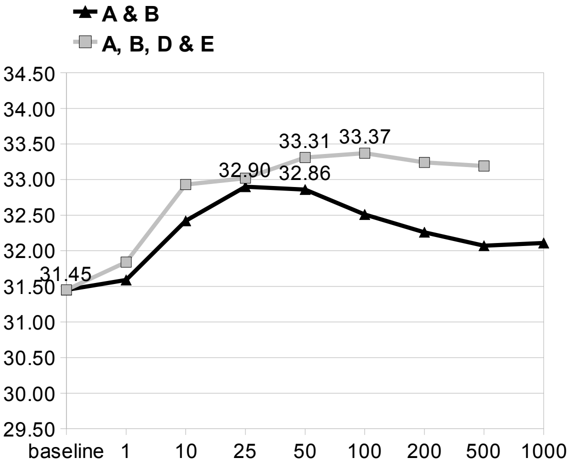

Figure 1: Combination results for different n-best sizes for two and four systems for MT06 in BLEU

Because unique 1000-best lists were only available from the systems A, B, D and E, we ran two series of experiments using the top two systems as well as these four systems. In the combination of two systems the n-best list sizes of 25 and 50 hypotheses achive virtually the same score (See Figure 1). The same is true for sizes 50 and 100 in the combination of four systems.

Because the experiments agree on 50 as the optimal size of the n-best list from each input system, and also because we only had unique 50-best lists available for one of the six systems, we chose the n-best list size of 50 hypotheses for all our following experiments.

The reason why the optimal n-best list size is rather small could be due to the fact that the input lists are ordered by the producing systems. Including hypotheses lower in the list introduces more and more bad hypothesis along with some good candidates. The restriction of the n-best list to the small size of 50, could be interpreted as indirectly using the knowledge of the decoder about the quality of the hypotheses, which is represented in the rank information.

3.6 Combination of all Systems

Starting with rescoring the n-best list of system A by itself, we progressively added all systems to the combination. The results in Table 4 show, that adding systems one by one improves the result with smaller impact for each additional system.

system	baseline	combined
A	31.45	31.76 / 58.95
+ B	31.28	32.86 / 57.90
+ C	31.25	33.32 / 56.87
+ D	31.04	33.51 / 56.77
+ E	30.36	**33.72** / 56.79
+ F	26.00	33.63 / **56.45**

Table 4: Combination results for adding in all systems progressively for MT06 in BLEU/TER

We achieved the best result of 33.72 BLEU by combining five systems, which is a gain of 2.27 points over the best system. The BLEU score on the tuning set MT03 for this combination was 38.63, which is 3.2 points higher than the best score on MT03 of 35.16 by system B. Adding the weakest system, with more than 5 BLEU points distance to the best system, does not improve the combination in the BLEU metric. However, even though system F also has the highest TER, the combination including all six systems could reduce TER by -0.75 over the best baseline TER of 57.20 by system D (see Table 2). TER is consistently reduced with adding more systems, but this result is not very meaningful since all individual systems as well as the combination were optimized to maximize BLEU.

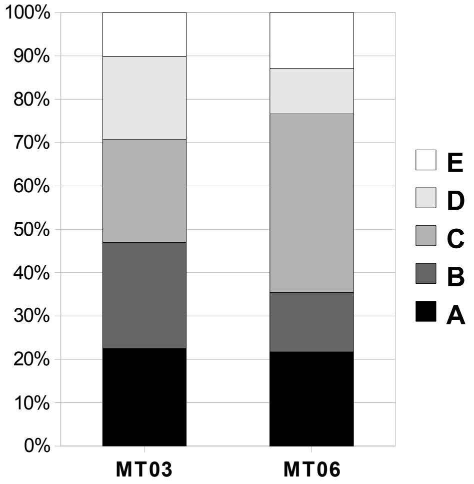

Figure 2: Contribution of each system to the new first best of the five system combination for MT03 and MT06

Figures 2 and 3 show the analysis of how many hypotheses were contributed by the different systems for the training set as well as for the unseen data. In some cases more than one system generated the highest ranking hypothesis for a source sentence, then it was awarded to all systems which generated it. We did not remove duplicate hypotheses generated by different input systems, because the boosting effect in the n-best list based features is desired. In the combination of all six systems, for example, 72 of the chosen hypotheses were generated by two systems, 4 by all six systems. These are typically very short sentences, for example by-lines.

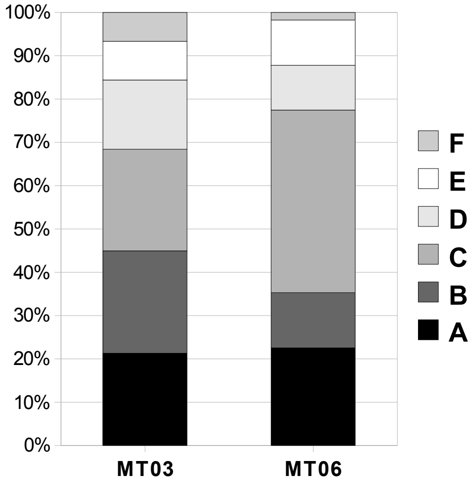

Figure 3: Contribution of each system to the new first best of the combination with all six systems for MT03 and MT06

The fact that we do not assign global system weights makes the combination more robust to varying performance of the individual systems on different translation data. E.g. system C contributed 24% to the combination of the tuning set, but 41% for the unseen data. This could indicate system C's more robust performance on the more diverse MT06 test set.

4 Conclusions and Future work

We introduce a relatively simple system combination method for machine translation systems. We select hypotheses from a joint n-best list of all input systems, using sentence level features calculated independently from the individual systems internal features. We combine n-best lists from up to six machine translation systems for the large scale Chinese to English translation task of the GALE and NIST evaluations in 2008 and achieve an improvement of +2.3 BLEU over the best individual system. An improvement of around two BLEU points is statistically significant but is difficult to detect for a human reader. The examples in Table 5 show cases, where the combination choose a hypothesis from a lower quality system over one from system A.

The language model and n-best list n-gram agreement feature groups have the biggest impact on the performance of the system combination. Removing any feature group decreases the BLEU score, if only insignificantly in most cases.

In our experiments the optimal n-best list size lies between 25-best and 100-best. Adding more and more systems to the combination improves the result progressively as long as the original system performance is not too much below the best system.

Our method can easily be extended to use more feature scores, for example sentence length penalty or source-target punctuation match or additional models like phrase tables, class based language models and distortion models. We will also try to leverage the rank information from the individual n-best lists and train global system weights.

Acknowledgments

I would like to thank the machine translation groups at UMD and CMU for providing the various system outputs. This work was partly funded by DARPA under the project GALE (Grant number #HR0011-06-2-0001).

References

Oliver Bender, Richard Zens, Evgeny Matusov, and Hermann Ney. 2004. Alignment templates: the RWTH SMT system. In *Proc. of the IWSLT*, pages 79–84, Kyoto, Japan.

Peter F. Brown, John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. 1990. A statistical approach to machine translation. In *Computational Linguistics*, volume 16(2).

Jonathan G. Fiscus. 1997. A post-processing system to yield reduced word error rates: Recogniser output voting error reduction (ROVER).

Saša Hasan, Richard Zens, and Hermann Ney. 2007. Are very large N-best lists useful for SMT? In *Human Language Technologies 2007: The Conference of the NAACL; Companion Volume, Short Papers*, pages 57–60, Rochester, New York, April. Association for Computational Linguistics.

Fei Huang and Kishore Papineni. 2007. Hierarchical system combination for machine translation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 277–286.

Shyamsundar Jayaraman and Alon Lavie. 2005. Multiengine machine translation guided by explicit word matching. In *Proceedings of EAMT 2005 (10th Annual Conference of the European Association for Machine Translation)*, Budapest, Hungary, May.

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

Franz Joseph Och, Daniel Gildea, Sanjeev Khudanpur, Anoop Sarkar, Kenji Yamada, Alex Fraser, Shankar Kumar, Libin Shen, David Smith, Katherine Eng, Viren Jain, Zhen Jin, and Dragomir Radev. 2004. A smorgasbord of features for statistical machine translation. In *the Human Language Technology Conference and the 5th Meeting of the NAACL: HLT-NAACL*, Boston, USA, May.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the ACL 2002, Philadelphia, USA*.

Antti-Veikko Rosti, Necip Fazil Ayan, Bing Xiang, Spyros Matsoukas, Richard Schwartz, and Bonnie Dorr. 2007. Combining outputs from multiple machine translation systems. In *Human Language Technologies 2007: The Conference of the NAACL; Proceedings of the Main Conference*, pages 228–235, Rochester, New York, April. Association for Computational Linguistics.

system A	the outlook for 2006 just sword :
system F	the sword of justice : the outlook for 2006
reference	sword of justice: outlook for 2006
system A	after the results were announced in early 2007 , will be held in the macao special administrative region (sar) held a presentation ceremony .
system E	the results will be announced at the beginning of 2007 , after the prize presentation ceremony held in the macao special administrative region (sar) .
reference	following announcement of the evaluation results, an award ceremony will be held in the macao sar in early 2007.
system A	you act of barbarism has crossed the line of civil society . in early october 2005 triad means you have used the hooligans beat me and lawyers
system E	you act of barbarism has crossed the bottom line of the civil society . in early october 2005 you used hooligans triad means beatings and li fangping lawyers
reference	you people's barbarism has exceeded the bottom line of a civilized society. at the beginning of october, 2005, you employed hooligan gangland tactics to beat the lawyer li fangping and myself

Table 5: Examples

K.C. Sim, W. Byrne, M. Gales, H. Sahbi, and P. Woodland. 2007. Consensus network decoding for statistical machine translation system combination. In *the 32nd International Conference on Acoustics, Speech, and Signal Processing*, Hawai, Apr.

Mathew Snover, Bonnie Dorr, Richard Schwartz, John Makhoul, Linnea Micciula, and Ralph Weischedel. 2005. A Study of Translation Error Rate with Targeted Human Annotation. Technical Report LAMP-TR-126,CS-TR-4755,UMIACS-TR-2005-58, University of Maryland, College Park and BBN Technologies, July.

Andreas Stolcke. 2002. Srilm - an extensible language modeling toolkit. In *Proceedings International Conference for Spoken Language Processing*, Denver, Colorado, September.

Nicola Ueffing and Hermann Ney. 2007. Word-level confidence estimation for machine translation. *Computational Linguistics*, 33(1):9–40.

Richard Zens and Hermann Ney. 2006. N-gram posterior probabilities for statistical machine translation. In *Proceedings on the Workshop on Statistical Machine Translation*, pages 72–77, New York City, June. Association for Computational Linguistics.

Richard Zens, Oliver Bender, Saša Hasan, Shahram Khadivi, Evgeny Matusov, Jia Xu, Yuqi Zhang, and Hermann Ney. 2005. The RWTH phrase-based statistical machine translation system. In *Proceedings of the IWSLT*, pages 155–162, Pittsburgh, PA, USA, October.