Quaero Speech-to-Text and Text Translation Evaluation Systems

Sebastian Stüker[1], Kevin Kilgour[1], and Jan Niehues[2]

[1] Research Group 3-01 'Multilingual Speech Recognition', Karlsruhe Institute of Technology, Karlsruhe, Germany, sebastian.stueker@kit.edu, kevin.kilgour@kit.edu

[2] Interactive Systems Laboratories, Karlsruhe Institute of Technology, Karlsruhe, Germany, jan.niehues@kit.edu

1 Introduction

Our laboratory has used the HP XC4000, the high performance computer of the federal state Baden-Württemberg, in order to participate in the second Quaero evaluation for automatic speech recognition (ASR) and Machine Translation (MT).

State-of-the-art ASR research systems usually employ techniques which require the parallel execution of several recognition systems for the purpose of system combination. The use of unsupervised adaptation techniques further requires the execution of several stages or passes of ASR systems. This leads to the fact that modern research systems process speech only with a run-time of many times realtime, under certain circumstances up to 100 times realtime. The process of speech recognition in this form can be easily parallelized at a speaker level in independent processes without the need for inter-process communication. Therefore the scheduling system of the XC4000 in combination with its global, high performing file space, is an ideal environment for executing such an evaluation.

The training of machine translation systems also requires the processing of large amounts of data in order to train statistical models.

2 Quaero

Quaero (http://www.quaero.org) is a French research and development program with German participation. It targets to develop multimedia and multilingual indexing and management tools for professional and general public applications such as the automatic analysis, classification, extraction, and exploitation of information. The projects within Quaero address five main application areas:

W.E. Nagel et al. (eds.), *High Performance Computing in Science and Engineering '10*, DOI 10.1007/978-3-642-15748-6_38,

- Multimedia Internet search
- Enhanced access services to audiovisual content on portals
- Personalized video selection and distribution
- Professional audiovisual asset management
- Digitalization and enrichment of library content, audiovisual cultural heritage, and scientific information.

Also included in Quaero is basic research in the technologies underlying these application areas, including automatic speech recognition, machine translation, and speech-to-speech translation. The vision of Quaero is to give the general public as well as professional user the technical means to access various information types and sources in digital form, that are available to everyone via personal computers, television, and handheld terminals, across languages.

Quaero is organized as a program consisting of seven projects. Five projects are concerned with applications. In addition, one project, the *Core Technology Cluster* (CTC), conducts basic research in the technologies underlying the application projects, and one project is concerned with providing the data resources necessary for the research within CTC.

Our laboratory is mainly involved in the CTC project. Two of the technologies under investigation are *Automatic Speech Recognition* (ASR), i.e. the automatic transcription of human speech into written records and *Machine Translation* (MT). Within Quaero research is driven by competitive evaluation and sharing of results and technologies employed. This process is called *coopetition*. Evaluations are conducted once a year on a predefined domain and a set of languages. As the project continues the number of languages to address will grow. Also the performance of the recognition systems developed within the project is expected to improve.

The second evaluation conducted in August 2009 was intended as a first, real evaluation after the baseline evaluation in 2008. Five languages were addressed: English, French, German, Russian, and Spanish. We participated in the languages English, German, Russian, and Spanish. The test data for the evaluation consisted of various audio files collected from the World Wide Web, including broadcast news, lectures, and video blogs.

3 English Evaluation Recognition Systems

For the English evaluation within Quaero we participated with a recognition system that is a further development of the ISL 2007 English Speech Transcription System for European Parliament Speeches from the European Integrated Project Technology and Corpora for Speech-to-Speech Translation (TC-STAR) [21]. The system has been trained and tested with the help of the Janus Recognition Toolkit that features the IBIS single pass decoder [22]. In general all recognition systems employ left-to-right Hidden Markov Models (HMMs), modeling phoneme sequences with 3 HMM states per phoneme.

3.1 Front-End

We trained systems for two different kinds of acoustic front-ends. One is based on the traditional Mel-frequency Cepstral Coefficients (MFCC) obtained from a fast Fourier Transform and the other on the warped minimum variance distortionless response (MVDR). The second front-end replaces the Fourier transformation by a warped MVDR spectral envelope [25], which is a time domain technique to estimate an all-pole model using a warped short time frequency axis such as the Mel scale. The use of the MVDR eliminates the overemphasis of harmonic peaks typically seen in medium and high pitched voiced speech when spectral estimation is based on linear prediction.

For training, both front-ends have provided features every 10 ms. During decoding this was changed to 8ms after the first stage. In training and decoding, the features were obtained either by the Fourier transformation followed by a Mel-filterbank or the warped MVDR spectral envelope.

For the MVDR front-end we used a model order of 22 without any filterbank since the warped MVDR already provides the properties of the Mel-filterbank, namely warping to the Mel-frequency and smoothing. The advantage of this approach over the use of a higher model order and a linear-filterbank for dimensionality reduction is an increase in resolution in low frequency regions which cannot be attained with traditionally used Mel-filterbanks. Furthermore, with the MVDR we apply an unequal modeling of spectral peaks and valleys that improves noise robustness, due to the fact that noise is mainly present in low energy regions.

Both frond ends apply vocal tract length normalization (VTLN) [26]. For MFCC this is done in the linear domain, for MVDR in the warped frequency domain. The MFCC front-end uses 13 cepstral coefficients, the MVDR front-end uses 15. The mean and variance of the cepstral coefficients were normalized on a per-utterance basis. For both front-ends seven adjacent frames were combined into one single feature vector. The resulting feature vectors were then reduced to 42 dimensions using linear discriminant analysis (LDA).

3.2 Acoustic Model Training

We trained acoustic models for two different kinds of phoneme sets *P1* and *P2*. P1 is a version of the Pronlex phoneme set which consists of 44 phonemes and allophones while P2 is a version of the phoneme set used by the CMU dictionary that consists of 45 phonemes and allophones. We trained models for all four combinations of the two phoneme sets and the two acoustic front-ends described above.

We used two different kinds of set of training data two train different acoustic models.

The first training set (*train1*) contains approximately 80h of English EPPS data provided by RWTH Aachen within the TC-STAR project [7], 9.8h of TED data [13], and 167h of unsupervised EPPS training material that had

been collected within TC-STAR by RWTH Aachen but had not been manually transcribed. Transcriptions for the unsupervised training material were obtained by adapting an acoustic model of last year's system on automatic transcriptions provided by RWTH Aachen on that data. We then decoded the data, using the segmentation provided by RWTH Aachen.

The second set of training data (*train2*) contains in addition to the data from the first training data set 140h of BroadCast News data from the HUB-4 corpus. Exploratory experiments showed that the use of broadcast news data leads to improvements on the web data in the Quaero development data, while the use of meeting data from the CHIL project did not lead to improvements.

All models are semi-continuous quinphone systems that use 16000 distributions over 4000 codebooks. They were trained using incremental splitting of Gaussians training, followed by 2 iterations of Viterbi training. For all models we used one global semi-tied covariance (STC) matrix after LDA [6] as well as Vocal Tract Length Normalization. In addition to that feature space constraint MLLR (cMLLR) speaker adaptive training [5] was applied on top.

We improved the acoustic models further with the help of Maximum Mutual Information Estimation (MMIE) training [19]. We applied MMIE training firstly to the models after the 2 viterbi iterations, and secondly to the models after the FSA-SAT training, taking the adaptation matrices from the last iteration of the maximum likelihood FSA-training and keeping them unchanged during the MMIE training.

This all resulted in fourteen different acoustic models: for each combination of front-end, MVDR and MFCC, and phoneme set, P1 and P2, one set of models trained with VTLN plus MMIE, and one with FSA-SAT plus MMIE. From now on we refer to these models as *P1-train1-MFCC-VTLN*, *P2-train1-MFCC-VTLN*, *P1-train1-MVDR-VTLN*, *P2-train1-MVDR-VTLN*, *P2-train2-MFCC-VTLN*, *P2-train2-MVDR-VTLN*, *P1-train1-MFCC-SAT*, *P2-train-MFCC-SAT*, *P1-train1-MVDR-SAT*, *P2-train1-MVDR-SAT*, *P1-train2-MFCC-SAT*, *P2-train2-MFCC-SAT*, *P1-train2-MVDR-SAT*, and *P2-train2-MVDR-SAT*.

Segmentation and Clustering

Segmenting the input data into smaller, sentence-like chunks used for recognition was performed with the help of a fast decoding pass on the unsegmented input data in order to determine speech and non-speech regions. Segmentation was then done by consecutively splitting segments at the longest non-speech region that was at least 0.3 seconds long. The resulting segments had to contain at least eight speech words and had to have a minimum duration of six seconds.

In order to group the resulting segments into several clusters, with each cluster, in the ideal case, corresponding to one individual speaker we used the same hierarchical, agglomerative clustering technique as last year which

Table 1. English text sources

Corpus	Wordcount
UK parliament debares (Hansard)	49,681,456
EPPS acoustic training data	749,668
EPPS text data	33043959
UN Parallel Text (English)	40,991,279
Gigaword 4rth Edition without 2008 texts	3,618,495,574
Hub4 Broadcast News data	832,068
Web dump (pre 2008)	144,062,839
total	3,887,856,843

is based on TGMM-GLR distance measurement and the Bayesian Information Criterion (BIC) stopping criterion [11]. The resulting speaker labels were used to perform acoustic model adaptation in the multipass decoding strategy described below.

Language Model and Test Dictionary

To select the vocabulary the development data text was split into a tuning set and a test set with each containing approximately half the text of every *show*. For each of our English text sources (see Table 1) we built a Witten-Bell smoothed unigram language model using the union of the text sources' vocabulary as the language models' vocabulary (global vocabulary). With the help of the maximum likelihood count estimation method described in [24] we found the best mixture weights for representing the tuning set's vocabulary as a weighted mixture of the sources' word counts thereby giving us a ranking of all the words in global vocabulary by their relevance to the tuning set. While the baseline 64k vocabulary had an OOV rate of 3.9% when measured on the validation set, the OOV rate of the vocabulary containing only the top ranked 64k words was 2.9%. This vocabulary was slowly increased until the OOV rate was under 1%. The final 130k vocabulary had a case sensitive OOV rate of 0.73%.

Pronunciations missing from the initial dictionary were created either manually or automatically with the help of Bill Fisher's tool [3] for P1 and Festival [2] for P2 respectively.

For each of the text sources in Table 1 we built, using the SRI Language Modeling Toolkit [23], a modified Kneser-Ney smoothed 4 gram language model. These were then interpolated using interpolation weights estimated on the aforementioned tuning set which was extendend with some general data to avoid overfitting problems. Because the resulting language model contained over 3×10^8 3 grams and 4 grams we decided to prune it to reduce its memory footprint and speed up decoding time. The final language model contained only slightly more than 6×10^7 3 grams and 4 grams.

Decoding Strategy and Results

Decoding within our recognition system was performed in two stages. The acoustic models of the second stage were adapted on the output(s) from the previous stage using Maximum Likelihood Linear Regression (MLLR) [14], Vocal Tract Length Normalization (VTLN) [26], and feature-space constrained MLLR (cMLLR) [5]. For the second and third stage the frame shift during recognition was changed to 8ms.

In the first stage we used the acoustic models P2-train1-MFCC-VTLN, P2-train-1-MVDR-VTLN, P2-train2-MFCC-VTLN, P2-train-2-MVDR-VTLN. The models trained on train1 were run on the EPPS portion of the test data only, the other models on the rest of the test data. The resulting word lattices were then combined via confusion network combination to the output *o1*. In this first stage we adapted the acoustic models using incremental VTLN and incremental fMLLR on a per speaker basis.

For the second stage all SAT systems were adapted on o1. The result of the different models were combined via confusion network combination to the final output.

On the official 2009 development set the system achieved a word error rate of 31.7%.

4 Translation System

In the Quaero text translation evaluation 2009 we participated in the following tasks: German-English, English-German, French-English, English-French, German-French, French-German. The submitted translations were generated by the following systems.

4.1 Data

Our translation systems were trained on the parallel corpora from the following sources: Europarl, Project Syndicate, Voltaire, and Who corpora. Table 2 shows few numbers about the training data.

The development sets for our systems were created by joining the nc-dev and nc-devtest sets from the WMT. The News Commentary 2007 test sets form the WMT were utilized as test sets for our systems.

Multiple monolingual corpora were used to train the language models. For every language at least two language models were trained: the target part of the parallel data and the News corpus. In addition to that, for English and French a language model was also trained on the Gigaword corpus.

For Discriminative Word Alignment training, hand-aligned data was used. For instance, DE-EN and EN-DE systems used a set of around 500 sentences from the Europarl domain annotated with “sure” and “possible” links. Whereas EN-FR and FR-EN used the data from the 2003 NAACL shared task and which consists of 443 sentences.

Table 2. Brief parallel corpora statistics

		Sydicate		Voltaire	
		Sentences	Words	Sentences	Words
EN-DE,	DE	41.2K	966.8K	189	4.4K
DE-EN	EN	41.2K	919.9K	189	4.4K
FR-DE,	FR	24.8K	659.4K	94	2.0K
DE-FR	DE	24.8K	607.5K	94	1.6K
EN-FR,	FR	28.5K	753.3K	17.9K	335.4k
FR-EN	EN	28.5K	666.7K	17.9K	340.1k
		Who		**Europarl**	
		Sentences	Words	Sentences	Words
EN-DE,	DE	–	–	1.42M	33.10M
DE-EN	EN	–	–	1.42M	35.69M
FR-DE,	FR	–	–	1.28M	33.73M
DE-FR	DE	–	–	1.28M	30.01M
EN-FR,	FR	122.3K	2.5M	1.43M	37.65M
FR-EN	EN	122.3K	2.1M	1.43M	36.20M

4.2 Preprocessing

The training data was preprocessed before used for training. In this step different normalizations like mapping different types of quotes were done. In the end the first word of every sentence was smartcased.

For the German text additional preprocessing steps were applied. First, the old German data uses the "Alte Deutsche Rechtschreibung" whereas the newer parts of the corpus use the "Neue Deutsche Rechtschreibung". We tried to normalize the text by converting the whole text to the "Neue Deutsche Rechtschreibung". In a first step, we search for words that are only correct according to the old writing rules. Therefore, we selected all words in the corpus, that are correct according to the hunspell lexicon using the old rules, but not correct according to the hunspell lexicon using the new rules. In a second step we tried to find the correct spelling according to the new rules. Therefore, we first applied some rules describing how words changed from one spelling system to the other like replacing 'ß' by 'ss'. If the new word is a correct word according to the hunspell lexicon using the new spelling rules, we map the words.

If translating from German to French or English, we apply compound splitting as described in [12] to the German corpus.

As a last preprocessing step we remove sentences that are too long and empty lines to get the final corpus.

4.3 Language Model

For all tasks and languages we use 4-gram language models generated using the SRILM Toolkit [23]. Different types and amounts of monolingual data

were used to train the language models. We experimented with the EPPS data, the News corpus, the News-Commentary corpus as in-domain data and the Gigaword corpus for those languages for which it is available (i.e. English and French).

Three different language model setups were applied: (1) Using a language model trained on one of the above mentioned data types. The factor for the language model is optimized during Minimum Error Rate Training. (2) Accessing two different language models at the same time from within the decoder. Here, two factors are optimized, one for each language model. (3) Combining two language models into one through linear interpolation. We choose the interpolation weights such that the perplexity of the resulting language model on the development set is minimized. In this case one language model factor was optimized with MERT.

4.4 Translation Model

In order to generate the translation model, i.e. the SMT phrase table, we first used the GIZA++ Toolkit [17] to calculate a word alignment for the training corpus. This was done in both directions and then the alignments were combined using the "grow-diag-final-and" heuristic. Afterwards the Moses Toolkit was used to build the phrase table.

The relative frequencies of the phrase pairs are a very important feature of the translation model, but they often overestimate rare phrase pairs. Therefore, the raw relative frequency estimates found in the phrase translation tables are smoothed by applying modified Kneser-Ney discounting as described in [4].

In addition to this baseline approach, some of the systems used a discriminative word alignment approach as described in [16] instead of using the word alignment generated by the "grow-diag-final-and" heuristic. For German-English and English-German we trained the discriminative word alignment on hand-aligned data and used the lexical probabilities as well as the fertilities generated by the GIZA++ Toolkit and POS information generated by the TreeTagger. We used all local features, the GIZA and indicator fertility features as well as the first order features for 6 directions. The model was trained in three steps first using the maximum likelihood optimization and afterwards it was optimized towards the alignment error rate. For more details see [10].

For French-German and German-French no hand-aligned data was available. To be able to use the discriminative word alignment for those language pairs as well, we reused a model trained on the English-German or German-English data. In this case we used only the lexical probabilities as well as the fertilities generated by the GIZA++ Toolkit as knowledge sources. To generate the alignment for French-German we then used the model trained on the English-German hand-aligned data and replace the English-German knowledge sources by the French-German ones. We did the same for German-French using the German-English model.

Since the test data was from the same source as part of the training data, we adapted the translation model towards this domain. Therefore, we trained a second phrase table only on the data from the project syndicate. Afterwards we combined both phrase tables in a log-linear way.

In addition we tried to trigger the relative frequencies in the phrase table by other words in the source sentence. Therefore, we calculated for every source word that cooccurs with a phrase pair, the relative probabilities for that phrase pair given that this source word occurs. In a next step, we then selected the trigger words, that led to a significant change in the relative frequencies. We then sorted the triggers by the information gain from this trigger and used the top N triggers in the phrase table. This was defined as

$$log(\frac{cooc(Trigger, PP)}{\#Trigger}) * \frac{cooc(Trigger, PP)}{\#Trigger} \quad (1)$$

where $cooc(Trigger, PP)$ is the number of cooccurrences of the trigger word and the phrase pair and $\#Trigger$ is the number of times the trigger occurs in the corpus. In the experiments the parameter N was set to 100.

During decoding we then checked if a trigger word of the phrase pair occurs in the source sentence. If this is the case, we used the triggered relative frequencies instead of the original ones.

4.5 Reordering Model

In addition to the built-in reordering in the decoder with a reordering window of two words, we also applied the POS-based reordering model presented in [20] to account for the different word orders in the languages. This model learns rules from a parallel text on how to reorder the source side such that it matches the word order of the target side. The aim is to generate a reordered source side that can be translated in a more monotone way.

In this framework, first, POS information is added to the source side and reordering rules are extracted from an aligned parallel corpus. These rules are of the form *VVIMP VMFIN PPER* → *PPER VMFIN VVIMP* and describe how the source side has to be reordered to match the target side. Then the rules are assigned scores according to their relative frequencies.

In a preprocessing step to the actual decoding different reorderings of the source sentences are encoded in a word lattice. Therefore, for all reordering rules that can be applied to a sentence the resulting reorderings are added to the lattice if the score is better than a given threshold. The decoding is then performed on the resulting word lattice.

This approach does model the reordering well if only short-range reorderings occur. But especially when translating from and to German, there are also long-range reorderings that require the verb to be shifted nearly across the whole sentence. During this shift of the verb, the rest of the sentence remains mainly unchanged. It does not matter which words are in between, since

they are moved as a whole. Furthermore, rules including an explicit sequence of POS-tags spanning the whole sentence would be too specific. A lot more rules would be needed to cover long-range reorderings with each rule being applicable only very sparsely. Therefore, we model long-range reordering by generalizing over the unaffected sequences and introduce rules with gaps. (For more details see [15]). These are learned in a way similar to the other type of reordering rules described above, but contain a gap representing one or several arbitrary words. It is, for example, possible to have the following rule *VAFIN * VVPP* → *VAFIN VVPP **, which puts both parts of the German verb next to each other.

4.6 Decoder

All our systems are currently based on the STTK toolkit for statistical machine translation, which is being developed at our institute (Interactive Systems Laboratories). The current UKA translation system is a phrase-based statistical MT system that combines several models in the log-linear domain. The standard models combine multiple lexical scores, word and phrase translation probabilities, relative phrase frequencies, word and phrase count features, and one or several n-gram language models.

The scaling factors for all system configurations are determined automatically and optimized with respect to BLEU score via Minimum Error Rate Training [1].

In some configurations, a lattice is used as input for the decoder. This lattice encodes different possible reorderings of the source sentence according to the POS-based reordering model described above.

4.7 Re-Ranking

For n-best list re-ranking we combined two unique 500-best lists, which may have less than 500 entries for some sentences. The two lists were generated by the same system using two different word re-ordering strategies.

We used several features computed from different information sources such as additional language models trained on the full word form, lemmatized text and POS tags, IBM-4 word lexica and the n-best list itself.

- Language models
- POS and Lemma LMs
- Statistical Word Lexica
- Position Dependent Word Agreement
- N-best List N-gram Agreement
- N-best List N-gram Probability
- Sentence Length Features

In each case we trained two language models, one from the target side of the bilingual data and on from all available target language data. For the translation into German we also used a language model from POS tags and one from lemmatized text. We calculated 4 features from the word lexica: the word probability sum as well as the maximum word probability in both language directions. The A position-dependent word agreement, the position-independent n-gram agreement, the n-best list n-gram probability and the sentence length features are calculated solely from information in the n-best list itself. The features are described in more detail in [9].

We used MER training [18] to optimize the feature weights on a n-best list for a tuning set.

In the evaluation we only used the n-best list re-ranking for the translation into German, because preliminary experiments had not shown stable improvements for other language directions. This might be due to the additional information from the POS and lemma LMs for German.

5 Parallelization Utilized

For the two different tasks described above, Automatic Speech Recognition and Machine Translation, the XC4000 was utilized at different stages in the development and application of the system. For ASR the XC4000 was used during decoding, while for MT the XC4000 was used during the training stage of the systems.

5.1 Automatic Speech Recognition

During the period described in this paper, only the actual decoding and adaptation algorithms for the ASR systems have been run on the cluster, but not the training algorithms. Decoding can run in parallel on a per-speaker basis without any need for inter-process communication with other parallel jobs.

The SLURM scheduler was thus used to simply start several processes in parallel that in theory can run independently of each other. The English test set contains 303 speakers, so that up to 303 parallel processes can be started for the decoding of the test set. In reality usually an amount of ca. 100 jobs was run in parallel. The runtime of the different jobs per speaker can vary greatly, depending on how much speech is associated with one speaker. Due to the accounting system of the queue on the cluster—one process still running on one CPU on one node, while all the other processes belonging to one scheduled job have already finished, will lead to being charged the same amount of CPU time as if all processes were still running—as little processes per scheduled job as possible were sought. However, due to the fact that only 10 jobs in the production environment per user are allowed—no matter how many nodes or CPUs are actually employed—this would mean that only 40 parallel jobs could be started, if 4 CPUs per node in the cluster are assumed.

In order to make use of more parallelization, therefore up to 8 or 16 processes per scheduled job were committed to the queue, even though that potentially means that one is being charged for runtime of nodes on which all processes have already finished.

5.2 Machine Translation

For the Machine Translation systems we used the XC4000 during the training phase of the systems. As described above, one of the main components of our training is the use of the GIZA++ training toolkit. The original training toolkit is not programmed to be executed in parallel but needs to run in a single process on all the training data.

For our experiments we used an adaptation of the GIZA++ training toolkit that has added the possibility of parallelization to the toolkit [8] and that can run a computing cluster such as the XC4000. Using the slurm scheduler deployed on the XC4000 we were then able to execute the training in parallel. Data sharing and result combination is done at file level using the shared file space of the XC4000 so that no inter-process combination is necessary.

Acknowledgements. This work was realized as part of the Quaero Programme, funded by OSEO, French State agency for innovation. The Research group '3-01 Multilingual Automatic Speech Recognition' received financial support by the 'Concept of the Future' of Karlsruhe Institute of Technology within the framework of the German Excellence Initiative.

References

1. Andreas Zollman, Ashish Venugopal, and Alex Waibel. Training and Evaluation Error Minimization Rules for Statistical Machine Translation. In *Proc. of ACL 2005, Workshop on Data-drive Machine Translation and Beyond (WPT-05)*, Ann Arbor, MI, 2005.
2. A.W. Black and P.A. Taylor. The festival speech synthesis system: System documentation. Technical report, Human Communication Research Centre, University of Edinburgh, Edinburgh, Scotland, United Kingdom, 1997.
3. W.M. Fisher. A statistical text-to-phone function using ngrams and rules. In *Proceedings the 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Phoenix, AZ, USA, December 1999. IEEE.
4. George Foster, Roland Kuhn, and Howard Johnson. Phrasetable Smoothing for Statistical Machine Translation. In *Proc. of Empirical Methods in Natural Language Processing*, Sydney, Australia, 2006.
5. M.J.F. Gales. Maximum likelihood linear transformations for hmm-based speech recognition. Technical report, Cambridge University, Engineering Department, May 1997.
6. M.J.F. Gales. Semi-tied covariance matrices for hidden Markov models. Technical report, Cambridge University, Engineering Department, February 1998.

7. Christian Gollan, Maximilian Bisani, Stephan Kanthak, Ralf Schlüter, and Hermann Ney. Cross domain automatic transcription on the tc-star epps corpus. In *Proceedings of the 2005 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'05)*, Philadelphia, PA, USA, March 2005.
8. Qin Gao and Stephan Vogel. Parallel implementation of word alignment tool. In *Proceedings of the ACL Workshops Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pages 49–57, Columbus, Ohio, June 2008. ACL.
9. Almut Silja Hildebrand and Stephan Vogel. Combination of machine translation systems via hypothesis selection from combined n-best lists. In *MT at work: Proceedings of the 8th Conference of the AMTA*, pages 254–261, Waikiki, Hawaii, October 2008.
10. Muntsin Kolss, Jan Niehues, Teresa Herrmann, and Alex Waibel. The Universität Karlsruhe Translation System for the EACL-WMT 2009. In *Fourth Workshop on Statistical Machine Translation (WMT 2009)*, Athens, Greece, 2009.
11. Qin Jin and Tanja Schultz. Speaker segmentation and clustering in meetings. In *Proceedings of the 8th International Conference on Spoken Language Processing (Interspeech 2004 — ICSLP)*, Jeju Island, Korea, October 2004. ISCA.
12. Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical Phrase-Based Translation. In *HLT/NAACL 2003*, 2003.
13. E. Leeuwis, M. Federico, and M. Cettolo. Language modeling and transcription of the ted corpus lectures. In *International Conference on Acoustics, Speech, and Signal Processing, Hong Kong*, China, March 2003.
14. C.J. Leggetter and P.C. Woodland. Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models. *Computer Speech and Language*, 9:171–185, 1995.
15. Jan Niehues and Muntsin Kolss. A POS-Based Model for Long-Range Reorderings in SMT. In *Fourth Workshop on Statistical Machine Translation (WMT 2009)*, Athens, Greece, 2009.
16. Jan Niehues and Stephan Vogel. Discriminative Word Alignment via Alignment Matrix Modeling. In *Proc. of Third ACL Workshop on Statistical Machine Translation*, Columbus, USA, 2008.
17. Franz J. Och. GIZA++: Training of statistical translation models. http://www.fjoch.com/GIZA++.html, 2000.
18. Franz Josef Och. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan, July 2003. Association for Computational Linguistics.
19. D. Povey and P.C. Woodland. Improved discriminative training techniques for large vocabulary continuous speech recognition. In *International Conference on Acoustics, Speech, and Signal Processing*, Salt Lake City, UT, USA, May 2001.
20. Kay Rottman and Stephan Vogel. Word reordering in statistical machine translation with a pos-based distortion model. In *TMI '07*, 2007.
21. Sebastian Stüker, Christian Fügen, Florian Kraft, and Matthias Wölfel. The isl 2007 English speech transcription system for European parliament speeches. In *Proceedings of the 10th European Conference on Speech Communication and Technology (INTERSPEECH 2007)*, pages 2609–2612, Antwerp, Belgium, August 2007.

22. H. Soltau, F. Metze, C. Fügen, and A. Waibel. A one pass-decoder based on polymorphic linguistic context assignment. In *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU '01)*, pages 214–217, Madonna di Campiglio Trento, Italy, December 2001.
23. A. Stolcke. SRILM – An Extensible Language Modeling Toolkit. In *Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP 2002)*, pages 901–904, Denver, CO, USA, 2002. ISCA.
24. A. Venkataraman and W. Wang. Techniques for effective vocabulary selection. *Arxiv preprint cs/0306022*, 2003.
25. M.C. Wölfel and J.W. McDonough. Minimum variance distortionless response spectralestimation, review and refinements. *IEEE Signal Processing Magazine*, 22(5):117–126, September 2005.
26. Puming Zhan and Martin Westphal. Speaker normalization based on frequency warping. In *Proceedings of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Munich, Germany, April 1997.