The KIT Translation Systems for IWSLT 2014

Isabel Slawik, Mohammed Mediani, Jan Niehues, Yuqi Zhang,
Eunah Cho, Teresa Herrmann, Thanh-Le Ha and Alex Waibel

Institute for Anthropomatics and Robotics
KIT - Karlsruhe Institute of Technology, Germany
firstname.lastname@kit.edu

Abstract

In this paper, we present the KIT systems participating in the TED translation tasks of the IWSLT 2014 machine translation evaluation. We submitted phrase-based translation systems for all three official directions, namely English→German, German→English, and English→French, as well as for the optional directions English→Chinese and English→Arabic. For the official directions we built systems both for the machine translation as well as the spoken language translation track.

This year we improved our systems' performance over last year through n-best list rescoring using neural network-based translation and language models and novel preordering rules based on tree information of multiple syntactic levels. Furthermore, we could successfully apply a novel phrase extraction algorithm and transliteration of unknown words for Arabic. We also submitted a contrastive system for German→English built with stemmed German adjectives.

For the SLT tracks, we used a monolingual translation system to translate the lowercased ASR hypotheses with all punctuation stripped to truecased, punctuated output as a preprocessing step to our usual translation system.

1. Introduction

The Karlsruhe Institute of Technology participated in the IWSLT 2014 Evaluation Campaign with systems for English→German, German→English and English→French, covering all official directions, as well as English→Chinese and English→Arabic. All systems were submitted for the machine translation (MT) track, with additional systems for the spoken language translation (SLT) track in the official directions. This year we also submitted three contrastive systems in order to directly compare the impact of some of our new models. For English→German we focused on the impact of rescoring on our system, for German→English we submitted a contrastive system that was built with stemmed adjectives on the German source side, and for English→Arabic we compared our alternative phrase table pruning method to the standard approach.

We focused our efforts on five components this year. The handling of ASR input was further refined (Section 3), and we newly implemented Restricted Boltzmann Machine (RBM)-based translation and language models for rescoring (Section 4), an alternative method to prune the phrase table (Section 5), a method to transliterate unknown words into Arabic (Section 6) and multiple level tree-based (MLT) reordering rules (Section 7).

The following section briefly describes our baseline system, while Sections 3 through 7 present the different components and extensions used by our phrase-based translation systems. After that, the results of the different experiments for the five language pairs we participated in are presented in Section 8 before we summarize our findings in Section 9.

2. Baseline system

All our systems are phrase-based systems. With the exception of the English→Chinese system, they are trained on the provided EPPS, NC and TED corpora. We also used the provided Common Crawl corpus for English↔German and Giga for English→French. For the monolingual training data we used the target side of all bilingual corpora as well as the News Shuffle corpus. Additionally, we included the Gigaword corpus for English→French and German→English. The English→Chinese system setup is described in Section 8.5.

Before training and translation, the data is preprocessed. During this phase, exceedingly long sentences and sentence pairs with a large length difference are discarded from the training data. We normalize special dates, numbers and symbols and smart-case the first letter of every sentence. For German→English, we split up compounds [1] on the source side of the corpus. Since the Common Crawl and Giga English→French corpus are very noisy, we trained an SVM classifier to filter them as described in [2].

After preprocessing, the parallel corpora are word-aligned using the GIZA++ Toolkit [3] in both directions. The resulting alignments are then combined using the grow-diag-final-and heuristic. The phrases are extracted using the Moses toolkit [4] and then scored by our in-house parallel phrase scorer [5]. Phrase table adaptation combining an in-domain and out-of-domain phrase table is performed as described in [6]. All translations are generated using our in-house phrase-based decoder [7].

Unless stated otherwise, we used 4-gram language models with modified Kneser-Ney smoothing, trained with the SRILM toolkit [8] and scored in the decoding process with KenLM [9]. In addition to common word-based language models, we used two token-based language models. The bilingual language model is used to increase the bilingual context during translation beyond phrase boundaries as described in [10]. A token consists of a target word and all its aligned source words. As a second token language model, we use a cluster language model based on word classes. This helps alleviate the sparsity problem for surface words by replacing every word in the training corpus with its cluster ID calculated by the MKCLS algorithm [11].

We use two main reordering models in our systems. The first consists of automatically learned reordering rules based on POS sequences [12] and syntactic parse tree constituents [13, 14] and performs source sentence reordering according to target language word order [15, 16, 17]. The resulting reordering possibilities for each source sentence are then encoded in a lattice. The second model is a lexicalized reordering model [18] which stores reordering probabilities for each phrase pair.

As an additional model, we use a Discriminative Word Lexicon (DWL) using source context features as described in [19].

We tune our systems using Minimum Error Rate Training (MERT) against the BLEU score as described in [20].

3. Preprocessing for speech translation

A conventional automatic speech recognition (ASR) system generates a stream of recognized words without punctuation marks or reliable case information. Therefore, when we use the ASR output as input for our MT system, it does not fit the style and format of the training data. In order to perform special preprocessing on the SLT test data, we use a monolingual translation system as presented in [21]. The system inserts punctuation marks and corrects case information, so that there is less divergence between the MT training data and the SLT input data. As sentence boundaries are already given in the test sets, we leave them as they are but predict other punctuation marks within the segment. This preprocessing will be denoted as Monolingual Comma and Case Insertion (MCCI).

For building the systems, we took the preprocessed source side of the parallel training data. We remove all punctuation marks from the data and insert a final period at the end of each line. In addition to this, all words are lowercased. This data is used as the source side of our monolingual translation systems. For the target side of the monolingual translation system, we keep the punctuation marks as well as case information, so that the "translation" of our MCCI system consists of inserting punctuation marks and correcting case information.

We built an MCCI system for English and German and applied it to all three official SLT track directions English→German, German→English and English→French.

4. n-best list rescoring

We perform additional experiments to use a neural network language and translation model in n-best list rescoring.

We train an 8-gram Restricted Boltzmann Machine (RBM)-based language model [22] on the in-domain TED corpus. The language model uses 32 hidden units and a shared word representation with 512 dimensionsUnigram sampling is applied as described in [23].

In addition, we use an RBM-based translation model inspired by the work of Devlin et al. [24]. The RBM models the joined probability of 8 target words and a set of attached source words. The set of attached source words is calculated as follows: We first use the the source word aligned to the last target word in the 8-gram. If this does not exist, we take the source word aligned to the nearest target word. The set of source words consists then of this source word, its previous 5 source words and its following 5 source words.

We create this set of 8 target and 11 source words for every target 8-gram in the parallel in-domain TED corpus. In rescoring, we then calculate the free energy of the RBM given the 8-gram and its source set as input. The sum of all free energies in the sentence is used as an additional feature for rescoring.

The 300-best list of the test set is then rescored using the additional features. In order to train the weights for the original features as well as the RBM-based models, we use the ListNet algorithm [25]. We use stochastic gradient descent to find the best weights and use batched updates with a batch size of 10.

5. Alternative phrase table pruning

For efficiency reasons, we always perform a phrase table pruning before decoding. Basically, we use a log-linear model with some a-priori fixed weights in order to rank the different phrase table entries associated with a given source n-gram. The n-best entries are then selected (n being a fixed integer). In the Arabic system, we experimented with a slightly different model in order to rank the entries. The first difference to our standard is that the different features are pre-normalized. Based on other experiments (not reported in this paper), the ℓ^3-normalization is the best suited for this task. That is, each feature value is divided by the cubic root of the sum of all the values raised to the power of 3.

Another difference resides in the fact that the ranking is based on the distance between the phrase table entries and a reference entry. The latter is obtained by combining the maximum scores of the different features in one entry. Based on the same aforementioned experiments, we selected the Jensen-Shannon distance measure for this task [26].

6. Arabic transliteration

In most cases, untranslated words break the harmony of the translation into a language which uses a different scripting (e.g. English into Arabic.) Therefore, it is more conve-

س → s
ن → n
ب → b

Figure 1: Examples of trivial correspondences

nient to transliterate those untranslated words, as they are unlikely to hurt the system performance further. Our transliteration is mostly similar to the character-based translation in its transliteration part [27]. It is consequently a statistical phrase-based translation based on unigram characters.

The corresponding training data of this system is mainly a subset of the word pairs obtained from the aligned corpora (TED and UN). First, the Arabic word of each aligned pair is roughly transliterated into English, using only trivial correspondences (see Fig. 1 for an example). The Levenshtein distance ratio is then computed between the resulting rough transliteration and the English word. Finally, we retain only pairs with ratios higher than a certain threshold (our threshold was empirically set to 0.5).

7. Multi-level tree reordering rules

For our English-Chinese translation we applied a novel rule-based preordering approach [28], which uses the tree information of multiple syntactic levels. This approach extends the tree-based reordering [17] from one level into multiple levels, which has the capability to process complex reordering cases.

Reordering patterns are based on multiple levels of the syntax tree. Figure 2 illustrates how the reordering patterns are detected. The detection starts from the root node of the syntax tree, goes downwards multiple levels and uses the nodes in these levels to detect the reordering pattern. In this example, the nodes that are used for detecting the reordering pattern are colored gray and have an italic font. The leaf nodes in the syntax tree are the words in the sentence. According to the alignment information, the node labeled with *NP* should be moved to the first place in the translation and the node labeled with *IN of* needs to be moved to the second place in the translated sentence. So from the root node with a search depth of 3, the following reordering pattern can be found:

`NP ( `**`CD`**$_0$` NP ( NP ( `**`JJ`**$_1$` `**`NNS`**$_2$` ) PP ( `**`IN`**$_3$
`NP`$_4$` ) ) ) -> NP IN CD JJ NNS`

`-> 4 3 0 1 2 (alternative with index)`

The algorithm for rule extraction detects the reordering patterns from all nodes in the syntax tree and it goes downwards for any number of levels, until it reaches the lowest level in the subtrees. The probability of the reordering patterns are calculated based on the frequency of their occurrences in the training corpus. In addition, reordering patterns that appear less often than a threshold are ignored in order to prevent too concrete rules lacking generalization capability and overfitting.

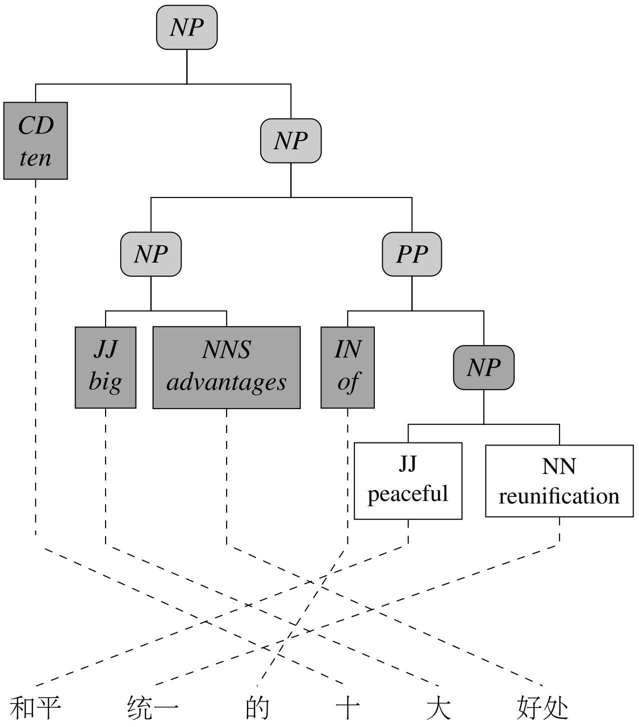

Figure 2: Detection of reordering pattern from multiple syntactic levels

When applying the rules prior to translation, the syntax tree is traversed by depth first search from the root of each subtree to its leaves. If a rule can be applied for a subtree at a given level, a new path for this reordering will be added to the word lattice for decoding. As long as rules can be applied on a subtree for a certain depth, the rules are applied and the search for rule application on this subtree stops.The search continues on the next subtree.

This multiple level tree-based (MLT) reordering rules can be combined with other types of reordering rules. This is done by combining the generated paths from different rules into one word lattice.

8. Results

In this section we present a summary of our experiments for both the MT and SLT tracks in the IWSLT 2014 evaluation. All the reported results are case-sensitive BLEU scores calculated on the provided development and test sets.

8.1. English→German

Table 1 shows the development stages of the English→German system. The baseline translation system uses two reordering models. First, in preprocessing, different possible source reorderings are encoded in a lattice. We used short-range and long-range POS-based reordering rules as well as tree-based rules. Secondly, a lexicalized reordering model on the phrase level is used. The phrase table is adapted by combining two phrase tables, one trained on all training data and one trained only on the TED in-domain corpus. Furthermore, the translation process is modeled using a bilingual language model trained on all parallel data and a discriminative word lexicon trained on the TED corpus. The DWL uses source context features. Finally, five language models are used. Three are word-based models, the first of which is trained on all available German data. The second one is trained only on the TED corpus. Finally, we use a word-based model trained on 5M sentences chosen through data selection [29]. In addition, a 9-gram POS-based language model and a 9-gram cluster language model using 1000 MKCLS classes are used. Afterwards, we rescored the system using the weights trained using the ListNet algorithm described in Section 4. The rescoring was trained on the test2010 and test2011 data and dev2010 was used as a cross-validation set. This results in an improvement of 0.3 BLEU points. Then we added an RBM-based language model and an RBM-based translation model. We could improve by using the RBM-based translation model by 0.4 BLEU points, reaching the best BLEU score on test2012 with 24.31 BLEU points. This system was submitted as our primary system for English→German. The baseline system without rescoring was submitted as a contrastive system.

System	Dev	Test
Baseline	27.3	23.67
Rescoring	-	23.97
RBMLM	-	23.94
RBMTM	-	**24.31**

Table 1: Experiments for English→German (MT)

8.1.1. *SLT track*

Table 2 shows the translation quality of the individual system components. First we used the MT system and tested it on the SLT test set dev2010. After adding inter-sentence punctuation marks to the ASR hypothesis using the MCCI approach, we could improve by 1.3 BLEU points. Afterwards, we also used the ListNet-based rescoring for this task. This time we used only test2010 as a training set and test2011 as our cross-validation set. This improved the translation quality by 0.1 BLEU points. Finally, we added the RBM-based language model and translation model. This gave additional improvements of 0.1 BLEU points. We submitted the MCCI system as a contrastive system and the system using RBMLM and RBMTM in rescoring as our primary one.

System	Dev	Test
Baseline	27.3	17.57
MCCI	-	18.83
Rescoring	-	18.91
RBMLM	-	19.02
RBMTM	-	18.96
RBMLM+TM	-	**19.01**

Table 2: Experiments for English→German (SLT)

8.2. German→English

Table 3 presents the results of our experiments for German→English. Our baseline system already incorporates a number of advanced models. Reordering is done using both POS-based preordering rules as well as a lexicalized reordering model. We adapted the in-domain and background phrase tables using the union candidate selection method. The system also includes a DWL trained on the in-domain data and five language models. In addition to the large background language model trained on all available English data, our baseline uses an in-domain language model, a background and in-domain bilingual language model, as well as a 9-gram in-domain cluster language model trained with 100 word classes. If we extend the preordering rules to include rules derived from parse trees, we can achieve a slight gain in BLEU. While the development score stays almost the same, we accomplish an improvement of nearly 0.3 BLEU points on the test data by extending the DWL to include source context. Training the DWL on n-best list data results in a similar gain in BLEU points yet again. We can further improve the score by applying the preordering rules learned from parse trees recursively. As our final model, we included a language model trained on on data automatically selected using cross-entropy differences [29]. We selected the top 10M sentences to train the language model. This leads to our final score of 31.98 BLEU points, almost 1 BLEU point over our baseline.

System	Dev	Test
Baseline	38.57	31.01
+ Tree Rules	38.79	31.04
+ DWL Source Context	38.78	31.32
+ DWL n-best List	38.86	31.63
+ Recursive Rules	38.92	31.71
+ Data Selection	39.03	**31.98**

Table 3: Experiments for German→English (MT)

8.2.1. *Adjective stemming*

Based on the system performing best in the previous experiment, we also submitted a contrastive system for German→English that employs stemming of adjectives.

Since German is a morphologically rich language, we are dealing with many surface forms. This creates data sparsity problems, as every surface form is treated as a distinct word in German. When translating into English, some of

System	Dev	Test
Primary	39.03	**31.98**
Stemmed	39.22	31.68

Table 4: Contrastive system for German→English (MT)

the information encoded in inflections such as gender or case may be discarded. However, stemming the whole German corpus hurts translation since too much information is lost. We therefore experimented with only stemming adjectives, which in German can have five different suffixes depending on the gender and case. The stemming was performed on the preprocessed files before compound splitting. The files were tagged with the TreeTagger [12] and the RFTagger [30]. We based our decision when and how to stem on the fine-grained tags output by the RFTagger. We only stemmed words tagged as an attributive adjective, since they are inflected in German. If the word as tagged as a comparative or superlative, we manually removed the inflected suffix in order to maintain the comparative nature of the adjective. For all other adjectives, we used the stem output by the TreeTagger. After stemming, compound splitting was applied as described in Section 2.

We then trained a new alignment and phrasetable on the stemmed corpora. Previous experiments had shown that using the stemmed phrasetable in conjunction with the unstemmed one gave better results than forcing the system to use the stemmed variant alone. However, our best system includes a DWL, biLM and cluster LM, which cannot be applied to the stemmed phrases in a straightforward manner. We therefore decided to unstem our phrasetable given the stems seen in the dev and test data. We looked at all the stem mappings from the development and test data and compiled a stem lexicon, mapping the surface forms observed in the Dev/Test data to their corresponding stems. We then applied this lexicon in reverse on our phrase table, in effect duplicating every entry containing a stemmed adjective with the inflected form replacing the stem. For translation we concatenated the default phrase table and the stemmed phrase table and combined the features log-linearly. This way our system was able to learn a weighing of the phrase scores during MERT. The resulting scores are reported in Table 4. While the stemmed system performs worse on the test data according to BLEU score, it does outperform our primary system on the development data. Using the stemmed system, we are able to translate seven adjectives we were not able to translate with our primary system. We therefore decided to submit our stemmed system as a contrastive system to fully evaluate our system's performance.

8.2.2. *SLT track*

Table 5 gives an overview of our systems for German→English SLT. As a baseline for the spoken language translation task, we used our best-performing system from the MT task. Applied to the ASR transcripts with only standard preprocessing, this gives us a baseline of 16.86 BLEU points. We can increase this score by nearly two BLEU points simply by adding a final period to every ASR segment. This shows that punctuation greatly influences the performance of our system. When we apply the more sophisticated MCCI system for punctuation and true casing of the test data, we achieve a similar improvement over the previous system. The last 0.2 BLEU points are gained by re-optimizing the system on development data that has been run through the MCCI system, resulting in our final system.

ASR Adaptation	Dev	Test
Baseline	39.03	16.86
+ period	-	18.79
MCCI	-	20.59
+ dev MCCI	35.79	**20.79**

Table 5: Experiments for German→English (SLT)

8.3. English→French

Table 6 summarizes the experiments performed for this direction.

The translation model of the baseline was built from TED, EPPS, NC, and Common-crawl corpora. It uses short-range POS-based reordering rules trained on TED, EPPS, and NC. It is also adapted to an in-domain translation model, exclusively trained on the TED corpus, using the union candidate selection method. In addition, 5 language models are used, 3 of which are conventional word-based LMs. One of the remaining LMs is a bilingual LM and the other is a cluster-based LM. The word-based LMs are trained on the French part of the parallel data, the monolingual data, and the union of all the French data respectively. The cluster-based LM is 4-gram trained on TED using 500 classes.

After that, we experimented with two different DWL models. The first small DWL was trained on the TED corpus only. It improves the score on Test by 0.15 BLEU points while its effect on Dev is negligible. The second model is larger. It was trained on EPPS and NC in addition to TED. With the large DWL, the gain is much more important: 0.2 BLEU points on Dev and 0.4 BLEU points on Test. For our submission we used this last configuration.

System	Dev	Test
Baseline	40.17	34.12
With small DWL	40.19	34.27
With large DWL	40.40	**34.66**

Table 6: Experiments for English→French (MT)

8.3.1. SLT track

As a baseline for the SLT track, we used our best performing English→French MT system on the automatically punctuated and cased version of the SLT input. We experimented with different ways of tuning the SLT system. These experiments are shown in Table 7.

The baseline uses all the models mentioned in the previous section (Section 8.3) except the cluster-based LM and DWL. In this configuration, both Dev (Dev2010) and Test (Test2010) sets were automatically punctuated and cased with MCCI. We then translated the test set with a comparable MT system without retuning on the punctuated Dev. This MT system was also tuned on the Dev2010 (on its text version though) and to our surprise this outperforms the baseline by almost 0.7 BLEU points. We could even get an additional gain (more than 0.3 BLEU), by tuning on the same MT tuning set (Test2011). By translating the test set with our final MT system (adding the cluster-based LM and the DWL to the baseline), the performance of the system was boosted by an additional 0.7 BLEU points. This final system was used in our submission.

System	Dev	Test
Baseline	22.53	23.35
MT tuned	-	24.03
MT tuned (2011)	-	24.38
+ DWL + clusterLM	-	**25.05**

Table 7: Experiments for English→French (SLT)

8.4. English→Arabic

The raw data provided for this pair was processed similarly to our English→Arabic system last year [31]. We show the effect of the two main extensions for this year's submission in Table 8. The baseline's translation model is built by performing adaptation on two models. The first is trained on all parallel data (UN and TED) and the other is trained on TED only. It integrates a bilingual LM and a cluster-based LM (with 500 classes), and 4 more word-based LMs. Three of the word-based LMs were respectively trained on the provided corpora (TED, UN, and Giga), and the last one incorporates all Arabic data. We used the alternative pruning without retuning, which gave us a gain of 0.2 BLEU points. The transliteration of the untranslated words however has an unnoticeable effect (0.01). We decided to include it in our system since it is unlikely to hurt the system as it is applied only to untranslated words. The primary system we submitted applied the alternative pruning and the transliteration, while the contrastive one used our standard pruning and transliteration.

System	Dev	Test
Baseline	15.98	7.71
+ Pruning	-	7.91
+ Transliteration	-	**7.92**

Table 8: Experiments for English→Arabic (MT)

8.5. English→Chinese

This year we also participated in the text translation task of English→Chinese. There are four novel methods applied in this year's system. First we have applied the new MLT reordering model as described in Section 7. Secondly, we added the ECI corpus (LDC94T5) to train the language model. Thirdly we tuned the system with the data set Test2011 and tested it with Test2012. Last but not least we built the system based on Chinese words instead of on Chinese characters.

The system is trained on the bilingual TED and filtered UN corpora. Since the UN corpus is document-aligned, we performed sentence alignment using the Kuhn–Munkres (KM) algorithm [32]. For each sentence pair, we used the number of aligned word pairs which occur in a dictionary (corpus LDC2002L27) as the weight for the KM algorithm. We then set a threshold and selected the 30k best-matching sentences for training.

The language models are trained on the monolingual TED, ECI, Google n-grams and the target side of the whole UN data. The Chinese target side is segmented with the Stanford word segmenter[1].

Table 9 shows the improvements step by step. We report not only the BLEU score on the words ($Test_w$), but also the score on the Chinese characters ($Test_{char}$). Briefly, the reordering models and adaptation have given the main contribution to the improvement of translation quality. The baseline is a monotone translation with 6-gram language model. We have used the POS-based long-range reordering and the MLT reordering model in combination. The MLT reordering model yields a consistent improvement of about 0.3 BLEU points over the long-range reordering model. We use the TED corpus as the in-domain data to adapt the phrase table and language model. This adaptation on the TED corpus improves the results up to about 0.7 BLEU points. We have added three more language models besides the basic 6-gram one. *google1980LM* is a 5-gram language model trained on the Google n-grams of the 1980s. We have also tried to use all the Google n-grams. However, it does not help to use more data. *BiLM* is a 4-gram bilingual language model and *clusterLM* is a 4-gram cluster-based language model.

[1] http://nlp.stanford.edu/software/segmenter.shtml

System	Dev_w	$Test_w$	$Test_{char}$
Baseline	13.73	12.07	19.18
+ POS Reordering (long)	14.08	12.24	19.34
+ MLT Reordering	14.34	12.57	19.68
+ Adaptation	14.93	13.34	20.65
+ google1980LM	15.13	12.67	20.02
+ BiLM	15.20	12.95	20.32
+ clusterLM	15.18	**13.58**	**20.88**

Table 9: Experiments for English→Chinese (MT)

9. Conclusions

In this paper, we presented the systems with which we participated in the TED tasks of the IWSLT 2014 Evaluation Campaign. In total we submitted twelve systems for five language pairs, consisting of five primary MT systems, three contrastive ones, three primary SLT systems and one contrastive SLT system.

For all languages we used strong baseline systems, including various word and token-based language models, adaptation techniques and combinations of preordering and lexicalized reordering models. Careful data selection and inclusion of individual models trained on different data proved successful in many of the systems.

A new model this year is a reordering model that operates on multiple tree levels, which was applied successfully for English→Chinese.

Further improvements could be achieved for English→German by n-best list rescoring with language and translation models trained with Restricted Boltzmann Machines.

For translation into Arabic, a special phrase table pruning technique gave an improvement over the baseline. Even though the merits of a transliteration approach did hardly reflect in BLEU, they did not harm and helped to unify translation appearancein the Arabic target output.

We submitted contrastive systems in order to show the impact of our novel n-best list rescoring, adjective stemming and phrase extraction approaches for English→German, German→English and English→Arabic respectively.

A monolingual translation system for comma insertion and case correction played a vital role in adjusting the ASR output for speech translation and was successfully applied in all three SLT systems.

10. Acknowledgements

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 287658.

11. References

[1] P. Koehn and K. Knight, "Empirical Methods for Compound Splitting," in *Proceedings of the 10th Conference of the European Chapter of the Association for Computational Linguistics*, Budapest, Hungary, 2003.

[2] M. Mediani, E. Cho, J. Niehues, T. Herrmann, and A. Waibel, "The KIT English-French Translation systems for IWSLT 2011," in *Proceedings of the 8th International Workshop on Spoken Language Translation*, San Francisco, CA, USA.

[3] F. J. Och and H. Ney, "A Systematic Comparison of Various Statistical Alignment Models," *Computational Linguistics*, vol. 29, no. 1, 2003.

[4] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses: Open Source Toolkit for Statistical Machine Translation," in *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, Prague, Czech Republic, 2007.

[5] M. Mediani, J. Niehues, and A. Waibel, "Parallel Phrase Scoring for Extra-large Corpora," in *The Prague Bulletin of Mathematical Linguistics*, no. 98, 2012.

[6] J. Niehues and A. Waibel, "Detailed Analysis of Different Strategies for Phrase Table Adaptation in SMT," in *Proceedings of the 10th Conference of the Association for Machine Translation in the Americas*, San Diego, CA, USA, 2012.

[7] S. Vogel, "SMT Decoder Dissected: Word Reordering." in *Proceedings of the International Conference on Natural Language Processing and Knowledge Engineering*, Beijing, China, 2003.

[8] A. Stolcke, "SRILM – An Extensible Language Modeling Toolkit." in *Proceedings of the International Conference on Spoken Language Processing*, Denver, CO, USA, 2002.

[9] K. Heafield, "KenLM: faster and smaller language model queries," in *Proceedings of the 6th Workshop on Statistical Machine Translation*, Edinburgh, United Kingdom, 2011.

[10] J. Niehues, T. Herrmann, S. Vogel, and A. Waibel, "Wider Context by Using Bilingual Language Models in Machine Translation," in *Proceedings of the 6th Workshop on Statistical Machine Translation*, Edinburgh, United Kingdom, 2011.

[11] F. J. Och, "An Efficient Method for Determining Bilingual Word Classes," in *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics*, Bergen, Norway, 1999.

[12] H. Schmid, "Probabilistic Part-of-Speech Tagging Using Decision Trees," in *Proceedings of the International Conference on New Methods in Language Processing*, Manchester, United Kingdom, 1994.

[13] A. N. Rafferty and C. D. Manning, “Parsing Three German Treebanks: Lexicalized and Unlexicalized Baselines,” in *Proceedings of the Workshop on Parsing German*, 2008.

[14] D. Klein and C. D. Manning, “Accurate Unlexicalized Parsing,” in *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, Sapporo, Japan, 2003.

[15] K. Rottmann and S. Vogel, “Word Reordering in Statistical Machine Translation with a POS-Based Distortion Model,” in *Proceedings of the 11th International Conference on Theoretical and Methodological Issues in Machine Translation*, Skövde, Sweden, 2007.

[16] J. Niehues and M. Kolss, “A POS-Based Model for Long-Range Reorderings in SMT,” in *Proceedings of the 4th Workshop on Statistical Machine Translation*, Athens, Greece, 2009.

[17] T. Herrmann, J. Niehues, and A. Waibel, “Combining Word Reordering Methods on different Linguistic Abstraction Levels for Statistical Machine Translation,” in *Proceedings of the 7th Workshop on Syntax, Semantics and Structure in Statistical Translation*, Altanta, GA, USA, 2013.

[18] P. Koehn, A. Axelrod, A. B. Mayne, C. Callison-Burch, M. Osborne, and D. Talbot, “Edinburgh System Description for the 2005 IWSLT Speech Translation Evaluation,” in *Proceedings of the 2nd International Workshop on Spoken Language Translation*, Pittsburgh, PA, USA, 2005.

[19] J. Niehues and A. Waibel, “An MT Error-Driven Discriminative Word Lexicon using Sentence Structure Features,” in *Proceedings of the 8th Workshop on Statistical Machine Translation*, Sofia, Bulgaria, 2013.

[20] A. Venugopal, A. Zollman, and A. Waibel, “Training and Evaluation Error Minimization Rules for Statistical Machine Translation,” in *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, Ann Arbor, MI, USA, 2005.

[21] E. Cho, J. Niehues, and A. Waibel, “Segmentation and Punctuation Prediction in Speech Language Translation using a Monolingual Translation System,” in *Proceedings of the 9th International Workshop on Spoken Language Translation*, Hong Kong, 2012.

[22] J. Niehues and A. Waibel, “Continuous Space Language Models using Restricted Boltzmann Machines,” in *Proceedings of the 9th International Workshop on Spoken Language Translation*, Hong Kong, 2012.

[23] J. Niehues, A. Allauzen, F. Yvon, and A. Waibel, “Combining Techniques from Different NN-based Language Models for Machine Translation,” in *Proceedings of the 11th Conference of the Association for Machine Translation in the Americas*, Vancouver, BC, Canada, 2014.

[24] J. Devlin, R. Zbib, Z. Huang, T. Lamar, R. Schwartz, and J. Makhoul, “Fast and Robust Neural Network Joint Models for Statistical Machine Translation,” in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, Baltimore, MD, USA, 2014.

[25] Z. Cao, T. Qin, T. yan Liu, M.-F. Tsai, and H. Li, “Learning to Rank: From Pairwise Approach to Listwise Approach,” in *Proceedings of the 24th International Conference on Machine Learning*, Corvallis, OR, USA, 2007.

[26] E. Deza and M. Deza, *Dictionary of Distances*. North-Holland, 2006.

[27] P. Nakov and J. Tiedemann, “Combining Word-level and Character-level Models for Machine Translation Between Closely-related Languages,” in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, Jeju Island, Korea, 2012.

[28] G. Wu, Y. Zhang, and A. Waibel, “Rule-Based Preordering on Multiple Syntactic Levels in Statistical Machine Translation,” in *Proceedings of the 11th International Workshop on Spoken Language Translation*, Lake Tahoe, USA, 2014.

[29] R. Moore and W. Lewis, “Intelligent Selection of Language Model Training Data,” in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, Uppsala, Sweden, 2010.

[30] H. Schmid and F. Laws, “Estimation of Conditional Probabilities with Decision Trees and an Application to Fine-Grained POS Tagging,” in *Proceedings of the 22nd International Conference on Computational Linguistics*, Manchester, United Kingdom, 2008.

[31] T.-L. Ha, T. Herrmann, J. Niehues, M. Mediani, E. Cho, Y. Zhang, I. Slawik, and A. Waibel, “The KIT Systems for IWSLT 2013,” in *Proceedings of the 10th International Workshop on Spoken Language Translation*, Heidelberg, Germany, 2013.

[32] H. W. Kuhn, “The Hungarian Method for the Assignment Problem,” *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, 1955.