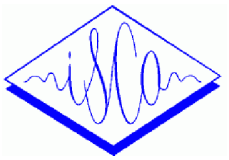


International Workshop on
Spoken Language Translation 2012
Hong Kong, December 6-7, 2012

The UEDIN Systems for the IWSLT 2012 Evaluation

Eva Hasler, Peter Bell, Arnab Ghoshal, Barry Haddow, Philipp Koehn,
Fergus McInnes, Steve Renals, Pawel Swietojanski

School of Informatics, University of Edinburgh, Edinburgh EH8 9AB, UK
{e.hasler,peter.bell,fergus.mcinnes,s.renals}@ed.ac.uk,
{aghoshal,pkoehn,bhaddow}@inf.ed.ac.uk,p.swietojanski@sms.ed.ac.uk

Abstract

This paper describes the University of Edinburgh (UEDIN) systems for the IWSLT 2012 Evaluation. We participated in the ASR (English), MT (English-French, German-English) and SLT (English-French) tracks.

1. Introduction

We report on experiments carried out for the development of automatic speech recognition (ASR), machine translation (MT) and spoken language translation (SLT) systems on the datasets of the International Workshop on Spoken Language Translation (IWSLT) 2012. Details about the evaluation campaign and the different evaluation tracks can be found in [1].

For the ASR track, we focused on the use of adaptive tandem features derived from deep neural networks, trained on both in-domain data from TED talks [2], and out-of-domain data from a corpus of meetings.

Our experiments for the MT track compare approaches to data filtering and phrase table adaptation and focus on adaptation by adding sparse lexicalised features. We explore different tuning setups on in-domain and mixed-domain systems.

For the SLT track, we carried out experiments with a punctuation insertion system as an intermediate step between speech recognition and machine translation, focussing on pre- and post-processing steps and comparing different tuning sets.

2. Automatic Speech Recognition (ASR)

In this section we describe the 2012 UEDIN system for the TED English transcription task. In summary, the system is an HMM-GMM system trained on TED talks available online, using tandem features derived from deep neural networks (DNNs). We were able to obtain benefits by including out-of-domain neural network features trained on a corpus of multi-party meetings. For recognition, a two-pass decoding architecture was used.

2.1. Acoustic modelling

Our core acoustic model training set was derived from 813 TED talks dating prior to the end of 2010. The recordings were automatically segmented, giving a total of 153 hours of speech. Each segment was matched to a portion of the manual transcriptions for the relevant talk using a lightly supervised technique described in [3]. For this purpose, we used existing acoustic models trained on multiparty meetings.

Three-state left-to-right HMMs were trained on features derived from the aligned TED data using a flat start initialisation. During the training process, a further re-alignment of the training segments and transcriptions was carried out, following which around 143 hours of speech remained for the final estimation of state-clustered cross-word triphone models. The resulting models contained approximately 3,000 tied states, with 16 Gaussians per state. Recognition was performed using HTK's HDecode. The first pass recognition transcription was used to estimate a set of CMLLR transforms [4] for each talk, using a regression class tree with 32 leaf-nodes, which were used to adapt the models for a second decoding pass.

The acoustic features used in the baseline system were 13-dimensional PLP features with first, second and third order differential coefficients, projected to 39 dimensions using an HLDA transform. To obtain acoustic features for the final system, we carried out experiments on the use of acoustic features derived from neural networks in the tandem framework [5]. Following our successful experience in [6], we investigated the use of features derived from networks trained on out-of-domain data using the Multi-layer Adaptive Networks (MLAN) architecture. In MLAN, tandem features are generated from in-domain data using neural network weights trained on out-of-domain data, and concatenated with in-domain PLP features and derivatives. A second, adaptive neural network is trained on these features. The final MLAN features used for HMM training and as input to the recogniser are obtained by concatenating posteriors from this second network with the original PLPs, projected with an HLDA transform. Figure 1 contrasts the MLAN process with the more standard use of out-of-domain posterior features. The procedure is described in more detail in [6].

In the experiments presented here, HMMs were trained

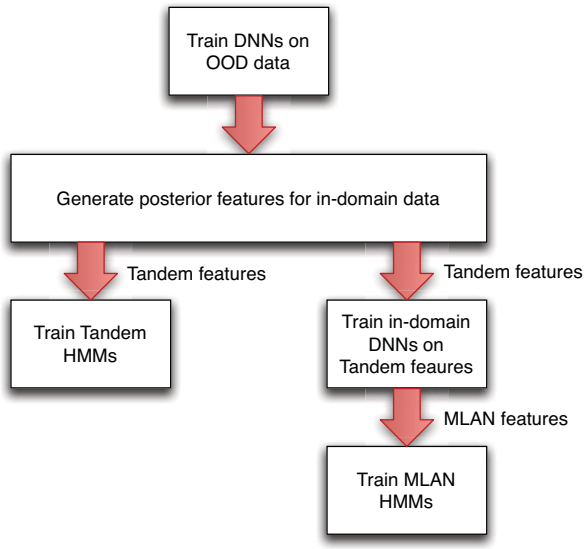


Figure 1: Multi-Level Adaptive Network (MLAN) architecture

on three sets of features:

- *In-domain* tandem features derived from four-layer deep neural networks (DNN) trained on the TED PLP features using monophone targets fixed by forced alignment with the baseline PLP models
- *Out-of-domain* features generated from Stacked Bottleneck networks trained on 120 hours of multi-party meetings from the AMI corpus using the setup described in [7]. Note that in general this domain is not well-matched to the TED domain[1]
- MLAN features obtained from four-layer DNNs trained on the AMI neural network features, concatenated with in-domain PLP features, again using monophone targets

The HMMs were trained using the tandem framework: the various neural network features were projected to 30 dimensions[2] and augmented with in-domain PLP features, projected from 52 to 39 dimensions with an HLDA transform, giving a total feature vector dimension of 69 in all three cases.

In the initial experiments, the HMMs were trained with maximum-likelihood training only. For the final system, we additionally employed speaker-adaptive training (SAT) [4] and MPE discriminative training [8]. When adaptation transforms were applied to the tandem features, the neural network and PLP features were adapted independently, using block diagonal (39x39 and 30x30) transforms.

[1]Standard HMMs trained on the AMI corpus, adapted using CMLLR to the test data, gave WER of 32.0% and 30.7% on the dev2010 and tst2010 sets respectively

[2]Except for the AMI bottleneck features, which were obtained from a 30-dimensional bottleneck with no further projection

Corpus	Word count
IWSLT12.TALK.train.en (in-domain)	2.4M
Europarl v7	54M
News commentary v7	4.4M
News crawl 2007	24.4M
News crawl 2008	23.1M
News crawl 2009	23.4M
News crawl 2010	23.9M
News crawl 2011	47.3M
Total	202.9M

Table 1: LM training data sizes.

2.2. Language modelling

The language models used for the ASR evaluation were obtained by interpolating individual modified Kneser-Ney discounted LMs trained on the small in-domain corpus of TED transcripts and the larger out-of-domain sources. The out-of-domain sources were europarl (v7), news commentary (v7) and news crawl data from 2007 to 2011. A random 1M sentence subset of each of news crawl 2007-2010 was used, instead of the entire available data, for quicker processing. The size of the resulting LM training data is shown in Table 1. The LMs were estimated using the SRILM toolkit [9]. The interpolated LMs had a perplexity of 160 (for 3-gram) and 159 (for 4-gram) on the combined dev2010 and tst2010 data. The optimal interpolation weights for both the 3-gram and 4-gram LMs were roughly 0.64 for the in-domain LM and between 0.02 and 0.06 for the different out-of-domain models. The vocabulary was fixed at 60,000 words.

We also carried out experiments using a language model built for the 2009 NIST Rich Transcription evaluation (RT09). This model was trained on a range of data sources, including corpora of conversational speech and meetings – see [7] for details. The vocabulary for this model was fixed at 50,000.

2.3. Results

We firstly carried out experiments on the dev2010 and tst2010 development data sets, using the NIST scoring toolkit to measure word error rate (WER). Our system models the initials in acronyms such as U.S., U.K. etc as individual words – for internal consistency, the development results here do not apply the automatic contraction of initials, which would result in an approximate 0.3% drop in WER below the figures shown. (Our final evaluation system, however, does include this correction).

Table 2 shows results of a two-pass speaker-adaptive system using the LM built for the IWSLT evaluation. All figures use a trigram LM except for the final row in the table. The results compare the use of different tandem features, and confirm our earlier findings that the MLAN technique is an effective method of domain adaptation, even when the domains are not particularly well matched. The use of SAT and

System	dev2010	tst2010
PLP + HLDA	26.7	24.9
TED tandem	21.3	20.3
AMI tandem	22.8	20.7
MLAN	20.5	18.7
+ SAT + MPE	18.5	16.4
+ 4gram LM	18.3	16.3

Table 2: Development set results (WER/%).

System	WER
MLAN	15.1
+ SAT + MPE	12.8
+ 4gram LM	12.4

Table 3: Results of MLAN systems on the *tst2011* test set

MPE training yields further improvements on the best feature set.

Somewhat unexpectedly, we found the RT09 LM to be more effective than the LM including in-domain data, with the best acoustic models achieving WER of 17.8% and 15.4% on dev2010 and tst2010 respectively. An interpolation of the two language models was found to yield even better performance, however, with WER of 17.1% and 14.7% respectively.

Finally, Table 3 shows results of selected acoustic models on the tst2011 test set, using our IWSLT language model. On the 2012 test data, the final system (MLAN + SAT + MPE + 4gram) achieved a WER of 14.4%.

3. Machine Translation (MT)

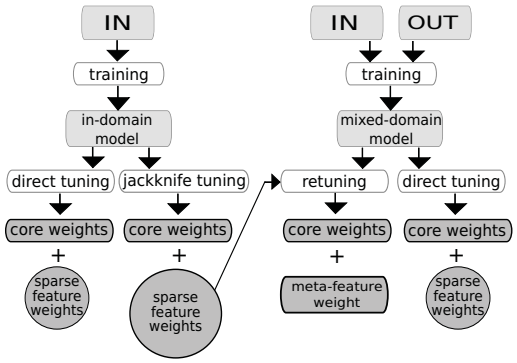
In this section we describe our machine translation systems for two language pairs of the MT track, English-French (en-fr) and German-English (de-en). We compare approaches to data filtering, phrase table adaptation and adaptation by adding sparse lexicalised features tuned on in-domain data, with different tuning setups.

3.1. Baseline SMT systems

Table 4 lists the available parallel and monolingual in-domain and out-of-domain training data. We built baseline systems with the Moses toolkit [10] on in-domain data (TED talks) as shown in tables 5 and 6 (labelled IN-PB and IN-HR) and further on in-domain data plus parallel out-of-domain data as shown in table 7 (labelled IN+OUT-PB). Parallel out-of-domain data consists of the Europarl, News Commentary and MultiUN corpora[3] for both language pairs and for en-fr also the French-English 10^9 corpus from WMT2012. The language models are 5-gram models with modified Kneser-Ney smoothing. Additional experiments were run with monolingual language model data from the Gigaword corpus (French Gigaword Second Edition, English Gigaword Fifth Edition) and News Crawl corpora from WMT2012, as marked in the results tables.

[3]For en-fr, this is the section from the year 2000 only, while for de-en it comprises the sections from 2000-2009.

Figure 2: *In-domain (IN) and mixed-domain (IN+OUT) models with three tuning schemes for tuning sparse feature weights: direct tuning, jackknife tuning and retuning.*



For the German-English systems we applied compound splitting [11] and syntactic pre-ordering [12] on the source side. As optimizers we used MERT as implemented in the current version of Moses and a modified version of the MIRA implementation in Moses as described in [13]. The language models were trained with the SRILM toolkit [9] and Kneser-Ney discounting. They were trained separately for each domain and subdomain (e.g. news data from different years) and linearly interpolated on the in-domain development set. Reported BLEU scores are case-insensitive and were computed using the mteval-v11b.pl script.

Hierarchical systems were only trained on in-domain data and lagged behind phrasebased performance by 0.7 BLEU for en-fr and 0.6 BLEU for de-en. Therefore, for all following systems we limited ourselves to phrasebased systems.

Table 4: *Word counts of in-domain and out-of-domain data.*

Parallel corpus	en-fr	de-en
TED (in-domain)	2.4M/2.5M	2.1M/2.2M
Europarl v7	50M/53M	45M/48M
News Commentary v7	3.0M/3.4M	3.5M/3.4M
MultiUN	316M/354M	5.5M/5.7M
10^9 corpus	576M/672M	n/a
Monolingual corpus	**fr**	**en**
TED (in-domain)	2.5M	2.4M
Europarl v7	55M	54M
News Commentary v7	4.2M	4.5M
News Crawl 2007-2011	512M	2.3G
Gigaword	820.6M	4.1G

3.2. Extensions

We experimented with several adaptation and tuning methods on top of our IN and IN+OUT baselines. One is the data selection method described in [14], using bilingual cross-entropy difference to select sentence pairs that are similar to the in-domain data and dissimilar to the out-of-domain data. We tried different filtering setups, selecting 10%, 20% and 50% of the parallel out-of-domain data. We also used the filtered target sides of the parallel data for building language models. Another approach is described in [15] (labelled *x+yE* there and *in+outE* here) and modifies the IN+OUT phrase tables by replacing all scores of phrase pairs found in the in-domain data by the values estimated on in-domain data only. The idea is to use the out-of-domain data only to provide additional phrases, i.e. to ignore counts from out-of-domain data whenever a phrase pair was seen in the in-domain data.

Table 5: *English-French in-domain (IN) systems trained with MERT (PB=phrasebased, HR=hierarchical), length ratio in brackets.*

System	test2010
IN-PB	**29.58 (0.966)**
IN-HR	28.94 (0.970)

Table 6: *German-English in-domain (IN) systems trained with MERT (PB=phrasebased, HR=hierarchical, PRE=preordering), length ratio in brackets.*

System	test2010
IN-PB (CS)	28.26 (0.999)
IN-PB (PRE)	28.04 (0.996)
IN-PB (CS + PRE)	**28.54 (0.995)**
IN-HR (CS + PRE)	27.88 (0.983)
IN-PB (CS + PRE)	
min=max=5	28.54 (0.995)
+ max=50	**28.57 (0.999)**
+ max=100	**28.60 (0.990)**
+ max=50, min=10	**28.65 (0.991)**

We tried several different approaches in order to specifically adapt the phrase pair choice to the style and vocabulary of TED talks. First, we added sparse word pair and phrase pair features on top of the in-domain translation systems and tuned them discriminatively with the MIRA algorithm. Word pair features are indicators of aligned pairs of a source and a target word, phrase pair features are indicators of a particular phrase pair used in a translation hypothesis and depend on the decoder segmentation of the source sentence. The values of these features in a translation hypothesis are counts of the number of times a word or phrase pair occurs in the current translation hypothesis. These sparse features are meant to capture preferred word and phrase choices in the in-domain data and therefore provide a bias for the translation model towards in-domain style and vocabulary. An example of a phrase pair feature is *pp_a,language∼une,langue=1*.

In the standard setup, sparse features were tuned on a small development set (dev2010), but we also used an alternative setup where they were tuned on the entire in-domain data, using 10 jackknife systems each trained on $\frac{9}{10}$ of the data and leaving out one fold for translation (the jackknife systems were run in parallel just like in normal parallelized discriminative tuning). We refer to the latter setup as *word pairs (JK)* and *phrase pairs (JK)*. For the systems built from in-domain and out-of-domain data (mixed-domain) we trained the sparse features on the development set as before. But since training with the jackknife setup would be rather time-consuming with the larger data sets, we reused the features trained on the in-domain data instead. In order to bring them on the right scale for the larger models, we ran a retuning step where jackknife-tuned features are treated as an additional component in the log-linear translation model. Running MERT on this extended model, we tuned a global meta-feature weight which is applied to all sparse features during decoding. Figure 2 gives an overview of all tuning setups involving sparse features on top of in-domain and mixed-domain models (direct tuning refers to sparse feature tuning on a development set). This is described in more detail in [13].

Table 7: *English-French and German-English mixed-domain (IN + OUT) systems trained with MERT, PB=phrasebased.*

System	test2010	
	en-fr	de-en
IN-PB	29.58	28.54
IN+OUT-PB	31.67	28.39
+ only in-domain LM	30.97	28.61
+ gigaword + newscrawl	31.96	30.26
IN-PB		
+ 10% OUT	32.30	**29.29**
+ 20% OUT	**32.45**	29.11
+ 50% OUT	32.32	28.68
best + gigaword + newscrawl	**32.93**	**31.06**
in+outE	32.19	29.59
+ only in-domain LM	30.89	29.36
+ gigaword + newscrawl	**32.72**	**31.30**

3.3. Results

In this section we compare results of the different data and tuning setups. Unless stated otherwise, the systems were tuned on the dev2010 set and evaluated on the test2010 set.

Table 5 shows the English-French systems and table 6 shows the German-English systems trained on in-domain (IN) data only. In both cases the phrase-based model outperformed the hierarchical model. For German-English, the best baseline system used both compound splitting and syntactic

Table 8: *German-English and English-French extensions of in-domain systems with sparse word pair and phrase pair features.*

System	test2010	
	en-fr	de-en
IN-PB, MERT	29.58	28.54
IN-PB, MIRA	30.28	28.31
+ word pairs	**30.36**	**28.45**
+ phrase pairs	**30.62**	**28.40**
+ word pairs (JK)	**30.80**	**28.78**
+ phrase pairs (JK)	**30.77**	**28.61**

pre-ordering. We tried different settings for the compound splitter, adjusting the minimum and maximum word counts. The min-counts avoids splitting into rare words, the max-count avoids splitting frequent words. The results indicate that changing the default values can yield a slight increase in performance.

Table 7 shows the mixed-domain systems (in-domain (IN) + out-of-domain data (OUT)) for both language pairs. The IN+OUT-PB baselines used the parallel data and the respective language model data. For en-fr, using additional out-of-domain data for the language model is better than using the in-domain LM alone (+0.7), but adding the newscrawl and gigaword data yields only a small further improvement (+0.3). For de-en, the IN+OUT-PB baseline is worse than the IN-PB baseline and improves when using only the in-domain LM. This indicates that the parallel OUT data is very dissimilar to the TED data for this language pair. However, adding newscrawl and gigaword data yields a larger improvement of 1.9 BLEU. The next block shows results of the data filtering approach and confirms the tendency from above. The de-en system profits from using only 10% of the OUT data (+0.9 BLEU) and adding more language model data yields an additional +1.8 BLEU. The en-fr system also benefits from using only part of the OUT data (+0.8 BLEU), in this case 20%, but only improves by 0.5 BLEU with additional LM data. The last block shows results of the *in+outE* approach, which uses the IN+OUT table but with scores from the IN table for all phrase pairs that were seen in the in-domain corpus. The results of this approach are comparable to the data selection method (a bit worse for en-fr and a bit better for de-en), but the advantage is that no data is thrown away and there is no need to tune a threshold for data selection.

Table 8 shows extensions of the in-domain systems for both language pairs. For en-fr, using MIRA to train the baseline system instead of MERT yields a gain of +0.7 BLEU and adding sparse word pair and phrase pair features adds a further 0.2 and 0.3 BLEU. We get the best performance by tuning the sparse features with the jackknife method, i.e. on all in-domain training data, yielding +1.2 over the MERT baseline. For de-en, the MIRA baseline is slightly worse than the MERT baseline, but adding sparse features on top of it has a similar positive effect. One thing to note is that the best weights during MIRA training were selected according to the test2010 set, so the results have to be considered optimistic when evaluating on test2010[4], while for evaluation on test2011 and test2012 we had distinct dev, devtest and test sets.

Table 9 shows combinations of the systems described in tables 7 and 8 for both language pairs. In the first block, we trained sparse features on a development set on top of the IN+OUT systems with data selection (10% for de-en and 20% for en-fr). In the second block, we applied a retuning step to integrate the sparse features trained on jackknife systems into the IN+OUT systems with data selection (see figure 2 for clarification). MERT results for test2010 are averaged over three runs, and the best of these three systems was used to translate test2011. For both language pairs we see improvements over the baselines with both methods of training sparse features (direct tuning and retuning) and we selected the best performing system on test2010 for submission (highlighted in grey). Evaluation on test2011 shows, however, that some of the contrastive systems (other systems from this table) perform better on this test set. The best performing systems on test2010 yield the following scores on test2011: for en-fr, 39.95 BLEU w/o additional LM data and 40.44 BLEU with additional newscrawl and gigaword data, and for de-en, 33.31 BLEU w/o additional LM data and 36.03 BLEU with additional gigaword and newscrawl data.

The systems used for our submissions did not include the additional monolingual data, which add an additional 0.5 BLEU for en-fr and 2.7 BLEU for de-en. As mentioned above, our en-fr system includes only one portion of the multiUN data (from the year 2000) instead of all data from years 2000-2009.

4. Spoken Language Translation (SLT)

Our SLT system takes the output of an ASR system, applies several transformational steps and then translates the output to French, using one of our English→French systems from section 3. We compare different preprocessing and tuning setups and show results on the outputs of four different ASR systems.

The transformations between ASR output and MT input are a pipeline consisting of three steps.

1. preprocessing of ASR output (number conversion)
2. punctuation insertion by translation from English w/o punctuation to English with punctuation
3. postprocessing (punctuation correction)

In the proprocessing step, we convert numbers that are represented in a systematically different way compared to the

[4]Though past experiments have suggested that choosing the weights on the development set instead does no make much difference.

Table 9: *German-English and English-French extensions of mixed-domain systems with sparse features. Grey cells mark systems used for submissions. Results of MERT-tuned systems for test2010 are averages over three runs of which the best was chosen for translating test2011.*

System	en-fr		de-en	
	test2010	test2011	test2010	test2011
IN-PB + 10%/20% OUT, MIRA	33.22	40.02	28.90	**34.03**
+ word pairs	**33.59**	39.95	28.93	33.88
+ phrase pairs	33.44	40.02	29.13	33.99
IN-PB + 10%/20% OUT, MERT	32.32	39.36	29.13	33.29
+ retune(word pairs JK)	32.90	**40.31**	**29.58**	33.31
+ retune(phrase pairs JK)	32.69	39.32	29.38	33.23
Submission system (grey) + gigaword + newscrawl	**33.98**	**40.44**	**31.28**	**36.03**

MT input data (details below). The punctuation insertion system is a standard MT translation system and is similar to the FullPunct-PPMT setup described in [16]. It was trained with the Moses toolkit [10] on 141M parallel sentences from the TED corpus, where the source side consists of transcribed speech and the target side consists of the source side of the parallel MT data. Source and target TED talks were first mapped according to talkids and then sentence-aligned. All speaker information was removed from the data.

Table 10 shows several variants of the punctuation insertion system. The evaluation metric is BLEU with respect to the MT source texts, because the punctuation insertion systems tries to 'translate' ASR outputs into MT inputs. Baseline1 refers to the training data of 141M parallel sentences, baseline2 used this data plus a duplicate of it where all but the sentence-final punctuation was removed. The idea was to avoid excessive insertion of punctuation by providing the system with both alternatives (the same phrases with and without punctuation), but this did not yield better results when combined with the original casing (w/o truecasing). To avoid introducing noise during decoding, we restricted the system to monotone decoding. Truecasing is usually useful to reduce data sparseness, but for punctuation insertion it turned out to be better to keep the original case information in order to avoid inserting sentence-initial punctuation. We also tried removing all quotes from the training data since predicting opening and closing quotes is more difficult than predicting other kinds of punctuation, but this did not yield improvements. In a first step we only converted year numbers with regular expressions, for example

- *nineteen thirty two* $\rightarrow$ *1932*
- *two thousand and nine* $\rightarrow$ *2009*
- *nineteen nineties* $\rightarrow$ *1990s*

Even though there is no strict convention of number representation in MT data, we also tried converting more types of numbers like

- *one hundred seventy four* $\rightarrow$ *174*
- *a hundred and twenty* $\rightarrow$ *120*
- *twenty sixth* $\rightarrow$ *26th*

which yielded some additional improvements. Postprocessing of punctuation insertions removes punctuation from the beginning of the sentence (where it is sometimes erroneously inserted), inserts final periods when there is no sentence-final punctuation and tries to make quotation marks more consistent (by removing single quotation marks or inserting additional ones).

Table 10: *Variants of punctuation insertion systems (evaluation set: test2010).*

Punctuation Insertion System	BLEU(MT source)
baseline 1	83.92
+ monotone decoding	84.01
+ w/o truecasing	84.49
+ w/o quotes	84.02
+ more number conversion	**84.80**
baseline 2	83.99
+ monotone decoding	84.04
+ w/o truecasing	83.76

We experimented with different tuning sets for the punctuation insertion system. The source side is one of devtest2010 ASR transcript, a concatenation of the dev2010 and test2010 ASR transcripts and a concatenation of the dev2010 and test2010 ASR outputs (all number-converted). The target side is the English side of the MT dev2010 set. Table 11 at the top shows the BLEU score with respect to the MT source of the raw ASR 2010 transcript and with number conversion. Next is the performance of the system that was tuned on dev2010 ASR transcripts. The number-converted ASR transcript improves by over 13 BLEU points when running it through the punctuation insertion system. As expected, there is a large gap between the quality of ASR

Table 12: *ASR outputs (English) → French. The punctuation insertion system used for test2010 was trained on ASR transcripts, the system used for test2011/test2012 on ASR outputs.*

SLT pipeline + MT System	BLEU(MT source)	BLEU(MT target)	Oracle
test2010 ASR transcript	85.17	30.54	33.98
test2010 ASR output UEDIN	61.82	22.89	33.98
test2011 ASR output system0	**67.40**	27.37	40.44
test2011 ASR output system1	65.73	**27.47**	40.44
test2011 ASR output system2	65.82	**27.48**	40.44
test2011 ASR output UEDIN	63.35	26.83	40.44
test2012 ASR output system0	70.73	n/a	n/a
test2012 ASR output system1	67.90	n/a	n/a
test2012 ASR output system2	66.82	n/a	n/a
test2012 ASR output UEDIN	63.74	n/a	n/a

Table 11: *Punctuation insertion + postprocessing with varying tuning and evaluation sets.*

Baselines w/o punctuation insertion	BLEU(MT source)
test2010 ASR transcript	70.79
+ number conversion	71.37
Punctuation Insertion System	**BLEU(MT source)**
Tune: dev2010 ASR transcript	
test2010 ASR transcript	84.80
+ postpr.	85.17
test2010 ASR output	61.65
+ postpr.	61.82
test2011 ASR output	62.04
+ postpr.	62.39
Tune: dev2010+tst2010 ASR transcripts	
test2011 ASR output + postpr.	**63.03**
Tune: dev2010+tst2010 ASR outputs	
test2011 ASR output + postpr.	**63.35**

transcripts vs. ASR outputs, but for all data sets the postprocessing step improves the quality. Thus, we can see that each step in the SLT pipeline improves the quality of the final output. The next two blocks show the quality of the test2011 system when the punctuation insertion system is tuned on a combination of the dev2010 and test2010 sets, both ASR transcripts and ASR outputs. Using more tuning data gains another 0.6 BLEU points and using real ASR outputs a further 0.3 BLEU improvement.

Table 12 shows the results of the complete SLT pipeline for test2010 and test2011 (the MT references for test2012 were not available at the time of writing). Before the translation step there is a large gap of more than 23 BLEU points between the ASR transcript and output, which mirrors the recognition errors. This results in a gap of more than 7 BLEU points after translation to French. The translation of the test2010 ASR transcript is 3.5 BLEU points below the translation of the real MT source set which is shown as the oracle (translation with perfect inputs). The MT sytem used for translation of the ASR output was the highlighted en-fr system from table 9, but here we are showing the results of translation systems with additional newscrawl and giga data (the difference was below 0.2 BLEU for the test2011 sets). Translating the test2010 set to English yields a BLEU score of 22.89. This could be improved by using ASR output of the dev2010 for tuning the punctuation system. For the test2011 set, there is gap of 4 BLEU points between the processed ASR outputs of the UEDIN system and the highest-ranking system (system0), measured against the MT source file. The BLEU score difference of the translations is only about 0.5 though, with system0 yielding a translation BLEU score of 27.37. Even though system0 yields the best BLEU score on the MT input file (67.40), system1 and system2 yield the best translation scores of the four systems, with 27.47 and 27.48 BLEU.

5. Conclusion

We presented our results for the ASR, MT and SLT tasks of the IWSLT 2012 Evaluation.

Our best ASR system for the TED task achieved scores of 12.4% on the 2011 test data set and 14.4% on the 2012 set. We found that the MLAN scheme for incorporating out-of-domain information using neural network features was effective in reducing WER compared to our standard tandem system.

Our largest MT systems yield BLEU scores of 40.44 for English-French and 36.03 for German-English on test2011. The data selection and phrase table adaptation methods showed comparable improvements over the mixed-domain baselines and we saw gains by adding sparse lexicalised features tuned on in-domain data. However, the relative results of our primary and constrastive systems varied quite a bit between the test2010 and test2011 data sets, so we cannot yet draw a final conclusion about an optimal setup.

Our SLT system yields BLEU scores between 26.83 and 27.48 on test2011, depending on the quality of the ASR outputs. Pre- and postprocessing of punctuation insertion turned out to be useful and we got slightly better results when tuning

the system on ASR outputs rather than ASR transcripts.

6. References

[1] M. Federico, M. Cettolo, L. Bentivogli, M. Paul, and S. Stüker, "Overview of the IWSLT 2012 evaluation campaign," in *Proc. of the International Workshop on Spoken Language Translation*, Hong Kong, HK, December 2012.

[2] M. Cettolo, C. Girardi, and M. Federico, "Wit^3: Web inventory of transcribed and translated talks," in *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, Trento, Italy, May 2012, pp. 261–268.

[3] A. Stan, P. Bell, and S. King, "A grapheme-based method for automatic alignment of speech and text data," in *Proc. IEEE Workshop on Spoken Language Technology*, Miama, Florida, USA, Dec. 2012.

[4] M. Gales, "Maximum likelihood linear transforms for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, no. 75-98, 1998.

[5] H. Hermanksy, D. Ellis, and S. Sharma, "Tandem connectionist feature extraction for conventional HMM systems," in *Proc. ICASSP*, 2000, pp. 1635–1630.

[6] P. Bell, M. Gales, P. Lanchantin, X. Liu, Y. Long, S. Renals, P. Swietojanski, and P. Woodland, "Transcription of multi-genre media archives using out-of-domain data," in *Proc. IEEE Workshop on Spoken Language Technology*, Miama, Florida, USA, Dec. 2012.

[7] T. Hain, L. Burget, J. Dines, P. Garner, F. Grezl, A. Hannani, M. Huijbregts, M. Karafiat, M. Lincoln, and V. Wan, "Transcribing meetings with the AMIDA systems," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 486–498, 2012.

[8] D. Povey and P. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proc. ICASSP*, vol. I, 2002, pp. 105–108.

[9] A. Stolcke, "SRILM – An Extensible Language Modeling Toolkit," in *Proc. ICSLP*, vol. 2, 2002, pp. 901–904.

[10] 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 *ACL 2007: proceedings of demo and poster sessions*. Prague, Czech Republic: Association for Computational Linguistics, June 2007, pp. 177–180.

[11] P. Koehn and K. Knight, "Empirical methods for compound splitting," in *In Proceedings of EACL*, 2003, pp. 187–193.

[12] M. Collins, P. Koehn, and I. Kučerová, "Clause restructuring for statistical machine translation," in *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ser. ACL '05. Stroudsburg, PA, USA: Association for Computational Linguistics, 2005, pp. 531–540.

[13] E. Hasler, B. Haddow, and P. Koehn, "Sparse lexicalised features and topic adaptation for SMT," in *Proceedings of the International Workshop on Spoken Language Translation*, Hong Kong, HK, December 2012.

[14] A. Axelrod, X. He, and J. Gao, "Domain adaptation via pseudo in-domain data selection," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, ser. EMNLP '11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 355–362.

[15] B. Haddow and P. Koehn, "Analysing the effect of Out-of-Domain data on SMT systems," in *Proceedings of the Seventh Workshop on Statistical Machine Translation*. Montréal, Canada: Association for Computational Linguistics, June 2012, pp. 422–432.

[16] J. Wuebker, M. Huck, S. Mansour, M. Freitag, M. Feng, S. Peitz, C. Schmidt, and H. Ney, "The RWTH Aachen machine translation system for IWSLT 2011," in *International Workshop on Spoken Language Translation*, San Francisco, California, USA, Dec. 2011, pp. 106–113.