The MITLL-AFRL IWSLT 2016 Systems[†]

Michaeel Kazi[1], *Elizabeth Salesky*[1], *Brian Thompson*[1], *Jonathan Taylor*[1], *Jeremy Gwinnup*[2]‡, *Timothy Anderson*[2], *Grant Erdmann*[2], *Eric Hansen*[2], *Brian Ore*[2], *Katherine Young*[2], *Michael Hutt*[2]

[1]MIT Lincoln Laboratory
Human Language Technology Group
244 Wood Street
Lexington, MA 02420, USA

[2]Air Force Research Laboratory
Airman Systems Directorate
2255 H Street
Wright-Patterson AFB, OH 45433, USA

Abstract

This report summarizes the MITLL-AFRL MT and ASR systems and the experiments run during the 2016 IWSLT evaluation campaign. Building on lessons learned from previous years' results, we refine our ASR systems and examine the explosion of neural machine translation systems and techniques developed in the past year. We experiment with a variety of phrase-based, hierarchical and neural-network approaches in machine translation and utilize system combination to create a composite system with the best characteristics of all MT approaches.

Preliminary results are denoted by * in this draft.

1. Introduction

During the evaluation campaign for the 2016 International Workshop on Spoken Language Translation (IWSLT16) [1] our experimental efforts in machine translation (MT) focused on the extension of our efforts from WMT16[2] and IWSLT15[3] and the exploration of many new neural machine translation (NMT) techniques including the refinement and improvement of our in-house NMT techniques, advanced selection techniques for parallel training data and the combination of this myriad of systems and techniques via system combination.

Our Automatic Speech Recognition (ASR) systems largely remain the same as last year, with the exception of training with this year's additional data.

2. Machine Translation

For our efforts in the machine translation task this year we acknowledge the recent explosion of neural machine translation (NMT) techniques and leverage our previous experience with phrase-based and hierarchical machine translation systems to create a best-of-breed machine translation system via the technique of system combination.

[†]Distribution A: approved for public release; unlimited distribution. This material is based upon work supported by the Air Force Research Laboratory in part under Air Force Contracts No. (FA8721-05-C-0002 and/or FA8702-15-D-0001). Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Air Force Research Laboratory.

‡Currently affiliated with the Air Force Research Laboratory.

2.1. Data Used

As in previous years, we made use of all data sources available to train various aspects of our MT systems: the WIT[3] Corpus [4], the new QED (formerly AMARA) Corpus [5], the new Parallel UN Corpus [6], and the existing Multi-UN Corpus [7]. We also make use of all the data sources available to WMT16[8].

2.2. Preprocessing

As in previous years, we use processes documented in [9] to clean the training data. The QED corpus required more processing as detailed in §2.2.3.

2.2.1. Farasa Processing

We applied Farasa[10] Arabic morphological processing to most of our MT systems. We experimented with variations of this processing, including the conversion of the resulting morphological fragments to individual tokens. An additional variation was to break each fragment into individual tokens instead of grouping them into either agglomerations of prefixes and suffixes, or prefix + root + suffix words.

2.2.2. Subword Units

As shown in [11] we use the benefits of byte-pair encoding on some of our neural MT systems to address the vocabulary-size problem. By breaking apart less-frequently seen words, we collapse the size of the vocabulary necessary for neural MT systems, allowing greater coverage of unseen words. This approach also has benefits for addressing transliteration of unknown words.

2.2.3. QED Corpus Processing for MT

We preprocessed the Arabic-English dataset from the QED corpus to correct sentence alignment errors and run-together words. These errors probably derive from the origin of the QED files as video transcriptions, assembled from short segments of video [9]. Sentences in the corpus are often split across lines, sometimes leaving the matching English and

Arabic words on different lines. We used line-final punctuation as a guide to assemble English lines into full sentences, while simultaneously concatenating their Arabic counterparts. Some Arabic files contain lines with just a period, corresponding to a blank line in the English file; we removed over 800 of these placeholders during concatenation. Our concatenation process failed when the sentences lacked punctuation or when the sentence-final punctuation fell in the middle of the line. This type of data led to very long concatenated sections. We therefore excluded files which exhibited excessive concatenation, measured as either a series of 5 or more concatenations with more than 500 total characters, or as an overall average of 30 or more words per line. These restrictions excluded 330 of our concatenated files. Three files were blank, and one other file was excluded on the basis of extremely bad spelling (including lowercase letter l for the personal pronoun, I). We retained 889 out of 1223 files, for a total of 72,475 concatenated lines.

The assembly of QED transcripts from short video segments may also lead to a chunking error, in which words are run together in the middle of each line in the file. We observed this type of error in a small number of English files. We used the Aspell[1] spell checker to identify run-together words such as *andthe* where the whole word is not in the Aspell dictionary, but the component words can be found. We had to manually review the list of suggested corrections to prevent the splitting of unknown names and technical terms; these were added to a supplemental Aspell dictionary. We also had to correct our automatic splits in some cases where there were multiple ways to split a run-together word (e.g., *breadcrumbscan* → *breadcrumb scan* OR *breadcrumbs can*). We implemented our corrections via table entries for the Varcon[2] variant conversion program.

Spelling was particularly bad in some English talks in the corpus. For example, *completly, enviorment, actualy, regardelss, satilites, correspinding, pricipal,* and so on. Many misspelled words showed up during our manual review of chunking errors. We identified files with excessive spelling problems and created additional spelling correction entries for the Varcon tables.

2.3. Training Data Subselection

Using definitions below, we select as a parallel training set a subset S from a large, general set C to maximize its similarity to a target set T, using a coverage metric $g(S,T)$. Defining $c_i(X)$ as the count of feature i's occurrence in corpus X,

$$g(S,T) = \frac{\sum_{i\in\mathcal{I}} f(\min(c_i(S), c_i(T)))}{\sum_{i\in\mathcal{I}} f(c_i(T)) + p_i(S,T)}$$

where the oversaturation penalty $p_i(S,T)$ is

$$\max(0, c_i(S) - c_i(T))\left[f(c_i(T)+1) - f(c_i(T))\right].$$

[1] `http://aspell.net`

[2] `http://wordlist.aspell.net/varcon`

The coverage maximization problem, $\max_{S\subset C} g(S,T)$, is solved via greedy optimization, iteratively adding the segment to S that provides the largest increase in g. The set S is reviewed after each addition, removing any older segment in S that decreases g.

We use $f(x) = \log(1+x)$ as a submodular function to weight counts. For Arabic the feature set $\mathcal{I}$ is composed of all unigrams and bigrams, based on testing over n-gram lengths. For English trigrams are added, again based on empirical testing. In our usage the set C is the Parallel UN corpus, and the target set T is composed of the TED dev and test sets from 2010–2013.

2.4. Neural Probabilistic Language Model Experiments

We trained several Neural Probabalistic Language Models (NPLM), partly with the goal of seeing whether the gain from hybrid neural MT systems was clearly better than augmenting a phrase-based system with feedforward networks. We also intended to try a character-level version of the Devlin [12] Neural Network Joint Model (NNJM). The character-input version replaces the input word vector layer with the convolutional approach described in [13]. To this end, we trained our own Tensorflow [14] implementation, and output the network in the NPLM format as required by Moses. We trained the model using the standard source context of 11, target of 3, and one to two hidden layers of size 512. The model was trained on in-domain TED data and validated on `tst2012`. The NNJM results are indicated in Table 1.

NNJM Description	Cased BLEU
Baseline, mosestoken	27.42
NNJM 2 HL, Rescoring	27.83
NNJM 2 HL x4 (s2t,t2s,l2r,r2l)	28.12
NNJM 1 HL, Decoding	27.61
Character 2 HL, Rescoring	27.75*
Character 1 HL, Decoding	27.75*

Table 1: Effects of NNJM integration into a baseline system without special Arabic processing, showing the benefit of character-level features on unstemmed data. Results are shown in BLEU decoding `tst2014`.

2.5. Moses MT Systems

Our baseline phrase-based system used the standard Moses [15] toolkit and only the provided in-domain training data. All Moses systems were tuned with Drem[16]. This baseline system was utilized as System 3 in system combination, shown in Table 5. Utilizing the parallel data selected in §2.3, we trained an additional system for combination listed as System 5.

While training additional Moses systems, for the purposes of obtaining an aligned development set (for Devlin models), we ran GIZA[17] on the in-domain TED data as

well as tst2012, but only using the former to build phrase-tables and models. The resulting improvement in GIZA alignment quality does make a small difference in translation quality. Our in-domain system additionally employed truecasing (trained on TED), hierarchical MSLR reordering [18], order 5 operational sequence model [19], an order-7 word class in-domain language model, and a 6-gram in-domain language model.

In addition to in-domain TED data, we used our language model from WMT16 consisting of all of the newscrawl data from 2007-2014, plus the news discussions and Europarl corpora. For extra parallel data, we experimented with domain adaptation from Multi-UN, Parallel UN, QED, and OpenSubtitles corpora. For the selection process, we used bilingual cross-entropy data selection [20], specifically the latest method from Axelrod et al [21], where we replace words outside of the top 10K most frequent words by a tag that includes the part-of-speech and the relative frequency of the word in the in-domain versus out-of-domain datasets. For the English data, we used the Stanford Part-of-Speech tagger [22], and for the Arabic, we induced word classes with ClusterCat[23]. The frequency bins used were powers of 10, as in [21]. We achieved a significant gain using Multi-UN data, as can be seen in Table 2.

Dataset + Num selected	Separate PT	Combined PT
Baseline	-	27.42
multi UN 500K	27.58	27.81
multi UN 1M	27.76	27.71
UN v1.0 1M	27.61	27.72
QED all	27.23	27.63
OpenSubtitles 1M	27.23	26.88
OpenSubtitles 2M	27.27	23.68

Table 2: Effects of additional parallel training data for phrase-based MT as scored against tst2014.

2.6. Nematus Systems

We were able to successfully train multiple Nematus [24] systems to achieve results on-par or slightly better than our phrase-based systems. Using the WMT16 scripts[3] provided by the author, we trained a system using all of the Multi-UN data. The system used byte-pair encoding trained on the union of Arabic and English text, with 160,000 split operations. The resulting vocabularies were approximately 120K source tokens and 80K target tokens. We validated this model during training on IWSLT tst2012 until the scores stabilized at approximately 29 uncased BLEU. Then we fine-tuned the model using the in-domain TED dataset, for a small number of epochs. This achieved 34.21* BLEU on tst2012 and 28.1 cased-BLEU on tst2014 - this system is utilized as System 8 in system combination as shown in Table 5. This same model was used to rerank the scores from our best phrase-based system, boosting the scores from 27.90 to 28.90 on tst2014.

Using this same technique for the QED task, we fine-tuned the same model on the QED training set described in Section 2.2.3, validating on a dev set (comprised of talks 0cvHoFWiJxVO, eODkKYQZcmjf, fbpZ98nxEgnj, SFFR5jvxTZHl, T4hMt9Ft5CKP, WVL2qxNoFdCC, Z6SoWjI2G6Em, ZPAQGyVEAUsV), boosting those scores by a factor of nearly double (See Table 7).

Since our phrase-based system did not use byte-pair encoding, the rescoring of the n-best list had a preprocessing step. This also made it difficult to decode with the model (see §2.8), so for the hybrid system, we also trained a Nematus system on truecase but not byte-pair encoded data, using an input vocabulary size of 160K and an output vocabulary of 80K. Decoding tst2014, this system achieved approximately 27 BLEU in the initial pass, and fine-tuned to 27.5* BLEU.

As a contrast, we also trained a Nematus system that used Farasa [10] to create subword units on the Arabic side, and byte-pair encoding on the English side. This system utilized a vocabulary size of 120,000 combined source/target. This system was motivated by the fact that Arabic words have prefixes and suffixes added to a root word to denote morphological information. By breaking apart the root word and the morphological suffixes and prefixes, we can reduce the size of the vocabulary used by a NMT system to the root words and a common set of prefixes and suffixes. While not performing as well as our straight byte-pair encoding Nematus system, it does add diversity to the systems used in system combination as described in §2.9. The result of decoding with the single-best model is listed as System 1 and the result of ensemble decoding with the 8-best models is listed as System 2.

2.7. Lamtram Systems

Following the success of the Nematus models, we trained additional neural machine translation systems using Lamtram [25]. We used 2x200 dimensional hidden layers. Our best system utilized the MultiUN corpus with byte-pair encoding and post-trained using the TED training data. Lamtram can incorporate probabilites from an external lexicon to boost translation probabilities. We used fast_align [26] to generate an IBM Model 1 lexicon. We did not have success running with minimum-risk. Results stabilized after 4 epochs with the UN data. BLEU scores improved by increasing beam size up to width of 10 which was the maximum possible under our GPU device contraints. Results are shown in Table 3. The best system was used in as System 4 in system combination (Table 5).

[3] https://github.com/rsennrich/wmt16-scripts

ID	System	train ppl	BLEU	cased BLEU
1	TED 4ep	16.6*	20.14*	20.52*
2	+ lexicon	16.6	21.22*	21.53*
3	MultiUN 3ep	4.1	19.83	20.96
4	MultiUN 4ep	4.0	20.78	21.86
5	+ TED post, 4ep	4.2*	23.43*	24.01*
6	+ lexicon	4.2	23.69	24.23

Table 3: Results on `tst2014` as measured in BLEU.

2.8. Hybrid MT Systems

An exciting development was the integration of neural MT models directly into the decoder – which is made possible by running on GPUs, using the AMUNMT[27] Moses variant. We used this tool to decode with our Nematus trained systems, yielding the improvements indicated in Table 4. We received more benefit from rescoring with the byte-pair encoded model than decoding with the non-byte-pair encoded model, and saw no gain from using both of them simultaneously.

System	Cased BLEU
Phrase-based BPE	26.92
+ NMT Decoding	27.89
Phrase-based no BPE	27.42
+ NMT Decoding	28.04
-/+ NMT Rescoring BPE	29.10

Table 4: Hybrid PB/NMT results decoding `tst2014` reported in cased BLEU

2.9. System Combination

With the wide variety of systems and techniques tested this year, system combination becomes important. We examined methods to combine the disparate translation outputs. Inspired by the success of the combination of multiple systems in the QT21/HimL submission[28] to WMT16[8], we utilized RWTH's Jane system combination technique [29] to combine outputs from each system to produce a unified, better translation result. Individual system inputs and combination results for decoding `tst2014` are listed in Table 5.

To determine the relative similarity of different system outputs for the purpose of system combination, we used automatic evaluation metrics. Outputs were scored against each other, using one of the outputs as the "reference". Figure 1 shows a comparison between systems, based on output for `tst2014`. Many metrics, such as BLEU, are asymmetric. The row of the table identifies the corpus treated as the hypothesis, and the column is the corpus treated as the reference. We see that the NMT systems are similar to each other, as are non-NMT systems. From experience we expect the greatest gain from system combination will be from combining somewhat dissimilar systems. Thus, we used at least one NMT and at least one non-NMT system in the final combination set.

ID	System	BLEU	Cased BLEU
1	Nematus Farasa SW	26.44	27.91
2	Nematus Farasa SW Ens 8	27.37	28.36
3	Moses PB Baseline	20.30	21.69
4	Lamtram	23.69	24.67
5	Moses PB Subsel 750 n32	21.77	23.07
6	Moses Hiero Farasa6	26.62	27.94
7	Moses PB Farasa6	26.01	27.50
8	Nematus 160k Vocab	28.10	29.30
9	Moses + Nematus Rescore	29.10	29.75
10	Moses + NemResc. + UNPT	29.41	30.13
Sys	System Combination	29.41	30.49

Table 5: Results for `tst2014` as measured in cased and uncased BLEU. These systems are used as components in system combination.

In fact, each of the 10 systems used in our system combination are very different (with the exception of systems 5 and 6, which are preprocessed and trained the same aside from decoding strategy), even moreso than the various component systems for QT21/HimL[28] submission to WMT16 (with the exceptions of closely-related systems 1-2 and 9-10.)

In Figure 2 we see the results of comparing the system-combination output against the next-best scoring contributing system utiliziing the MTComparEval tool [30].

2.10. MT Results

We ultimately submitted two systems for evaluation: our best individual system-combination effort and our best phrase-based system with neural MT rescoring. These systems are listed in Table 6.

System	BLEU	Cased BLEU
System Combination	30.49	29.41
Moses + Nematus Rescore	30.13	29.41

Table 6: Submission systems scores reported in BLEU decoding `tst2014`.

Table 7 shows the individual contributions of the methods used in our phrase-based + neural rescoring submission system, also noting the contributions of the cleaned QED corpus shown in §2.2.3.

3. ASR

ASR systems were trained and evaluated using the same procedure as in IWSLT 2015 [3], except that this year we used

	0	1	2	3	4	5	6	7	8	9	10
0	–	27.95	28.90	24.75	24.85	25.19	27.96	27.51	29.35	30.27	30.68
1	27.91	–	70.39	35.97	40.15	34.93	39.47	37.06	49.98	44.69	44.38
2	28.85	70.33	–	38.01	42.50	36.88	40.89	38.34	53.24	46.52	46.24
3	24.69	35.93	38.00	–	33.06	54.98	43.50	42.39	38.68	47.45	46.82
4	24.76	40.01	42.37	32.96	–	32.96	33.52	32.29	45.44	39.87	40.15
5	25.12	34.86	36.83	54.92	33.05	–	42.24	41.69	37.45	44.99	45.99
6	27.99	39.56	41.01	43.64	33.63	42.39	–	55.11	40.71	51.79	50.66
7	27.54	37.15	38.45	42.52	32.40	41.83	55.10	–	38.13	49.52	49.23
8	29.30	49.95	53.26	38.71	45.59	37.51	40.60	38.02	–	49.18	49.41
9	30.29	44.77	46.64	47.58	40.01	45.13	51.75	49.49	49.29	–	77.24
10	30.68	44.44	46.34	46.93	40.29	46.13	50.60	49.18	49.50	77.20	–

(a) Relative uncased, tokenized BLEU of MT systems decoding `tst2014`. Row is scored, column is reference.

0: Reference `tst2014`	4: Lamtram	8: Nematus 160k Vocab
1: Nematus Farasa SW	5 : Moses PB Subsel 750 n32	9: Moses + Nematus Rescore
2: Nematus Farasa SW Ens 8	6: Moses Hiero Farasa6	10: Moses + Nematus Rescore + UN PT
3: Moses PB Baseline	7: Moses PB Farasa6	

(b) MT System Legend

Figure 1: Machine Translation System comparison

Source	وماهو مشترك لديهم هو أن التحليلات المختلفة من وجهات النظر المختلفة تصبح جزءا أساسيا من العمل النهائي للهندسة المعمارية .
Reference	and also they have in common that the different analyses from different perspectives becomes an essential part of the final piece of architecture .
9-mos-nem-rescore-un.tst2014.out	and what they have in common is that different analyses of different viewpoints become a key part of the final work of architecture .
syscomb-10-tst2014	and what they have in common is that the different analyses of different perspectives become an essential part of the final work of architecture .

	BREVITY-PENALTY	BLEU	BLEU-cased	PRECISION	RECALL	F-MEASURE
9-mos-nem-rescore-un.tst2014.out	1	27.41	27.41	34.51	34.51	34.51
syscomb-10-tst2014	1	45.14	45.14	50.05	52.23	51.12
Diff	0.0000	-17.7300	-17.7300	-15.5400	-17.7200	-16.6100

Figure 2: MTComparEval example for `tst2014`.

additional acoustic and LM training data. The acoustic data were harvested from 2,050 TED talks using the alignment and closed caption filtering process described in [31], yielding 385 hours of audio. One ASR system was trained with Theano[32] and a version of the Hidden Markov Model Toolkit (HTK) that we modified according to the method of [33]; a second ASR system was built using the Kaldi open source speech recognition toolkit [34]. The LM training data included TED, QED, 1/8 of Gigaword, and 1/8 of News 2007-2015; the subsets of Gigaword and News 2007-2015 were selected using cross-entropy difference scoring with TED and QED as the in-domain text. Interpolated trigram and 4-gram LMs were estimated using the SRILM Toolkit, and a maximum entropy RNN LM was trained with the RNNLM Toolkit.

The test data were decoded using the same process as last year[3]. Table 8 shows the word error rate (WER) of each system on `tst2013` after evaluating the decoder, rescoring with the 4-gram LM, and interpolating the 4-gram and RNN LM scores. The final hypothesis for each utterance was selected by applying N-best Recognizer Output Voting Error Reduction (ROVER) to the output from the HTK adapted system and the Kaldi bottleneck system. The combined system yielded an 8.6% WER on `tst2013` and a 8.9%* WER on `tst2016`.

3.1. QED Corpus for ASR Language Model

The English QED files were processed to correct chunking errors (see Section 2.2). We corrected run-together words in 57 files. In addition, 4 files were excluded on the basis of significant problems with spelling or foreign language sections. One file was also excluded due to problems with duplicated and partially duplicated lines.

System	Cased BLEU
Baseline	25.02
+ DREM	25.55
+ Newscrawl LM	27.42
+ UN data	27.81
+ Rescore Nematus	29.41
Baseline UN Nematus	
+ Finetune TED	28.10
+ Ensemble	28.53*
System	`QED dev` BLEU
Baseline UN Nematus	16.61
Finetune QED training	32.63

Table 7: Additive scores for Moses + Nematus rescore system submission on `tst2014` (unless otherwise noted) as measured in BLEU.

ASR System	Decode	4-gram	4-gram+RNN
HTK first-pass	12.8	12.2	10.8
HTK adapted	10.8	10.4	9.5
Kaldi bottleneck	11.7	11.3	10.4

Table 8: English `tst2013` WER.

4. Conclusion

In closing we see that neural machine translation systems are the new, exciting area of research in the problem-space of machine translation despite growing pains. We see that the "old wisdom" of statistical machine translation systems is still useful and that a thoughtful combination of the two can produce translations greater than the sum of their parts.

5. References

[1] "Overview of the IWSLT 2016 Evaluation Campaign," in *Proceedings of the 11th International Workshop on Spoken Language Translation (IWSLT'16)*, ser. Proceedings of IWSLT, 2016.

[2] J. Gwinnup, T. Anderson, G. Erdmann, K. Young, M. Kazi, E. Salesky, and B. Thompson, "The AFRL-MITLL WMT16 news-translation task systems," in *Proceedings of the First Conference on Machine Translation*. Berlin, Germany: Association for Computational Linguistics, August 2016, pp. 296–302.

[3] M. Kazi, B. Thompson, E. Salesky, T. Anderson, G. Erdmann, E. Hansen, B. Ore, K. Young, J. Gwinnup, M. Hutt, and C. May, "The MITLL-AFRL IWSLT-2015 systems," in *Proc. of the 11th International Workshop on Spoken Language Translation (IWSLT'15)*, Da Nang, Vietnam, December 2015.

[4] 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.

[5] A. Abdelali, F. Guzman, H. Sajjad, and S. Vogel, "The AMARA corpus: Building parallel language resources for the educational domain," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, May 2014.

[6] M. Ziemski, M. Junczys-Dowmunt, and B. Pouliquen, "The united nations parallel corpus v1.0," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, May 2016.

[7] J. Tiedemann, "Parallel data, tools and interfaces in opus," in *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may 2012.

[8] O. Bojar, R. Chatterjee, C. Federmann, Y. Graham, B. Haddow, M. Huck, A. Jimeno Yepes, P. Koehn, V. Logacheva, C. Monz, M. Negri, A. Neveol, M. Neves, M. Popel, M. Post, R. Rubino, C. Scarton, L. Specia, M. Turchi, K. Verspoor, and M. Zampieri, "Findings of the 2016 conference on machine translation," in *Proceedings of the First Conference on Machine Translation*. Berlin, Germany: Association for Computational Linguistics, August 2016, pp. 131–198.

[9] K. Young, J. Gwinnup, and L. Schwartz, "A taxonomy of weeds: A field guide for corpus curators to winnowing the parallel text harvest," in *Proceedings of the 12th Biennial Conference of the Association for Machine Translation in the Americas (AMTA2016)*, 2016.

[10] A. Abdelali, K. Darwish, N. Durrani, and H. Mubarak, "Farasa: A fast and furious segmenter for arabic," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, 2016, pp. 11–16.

[11] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2016, pp. 1715–1725.

[12] J. Devlin, C. Quirk, and A. Menezes, "Pre-computable multi-layer neural network language models," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon,

Cleared for public release on 27 Oct 2016. Originator Reference Number: RH-16-116221. Case Number: 88ABW-2016-5426.

Portugal: Association for Computational Linguistics, September 2015, pp. 256–260. [Online]. Available: http://aclweb.org/anthology/D15-1029

[13] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush, "Character-aware neural language models," *CoRR*, vol. abs/1508.06615, 2015. [Online]. Available: http://arxiv.org/abs/1508.06615

[14] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: http://tensorflow.org/

[15] 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 *Proc. of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ser. ACL '07, 2007, pp. 177–180.

[16] G. Erdmann and J. Gwinnup, "Drem: The AFRL submission to the WMT15 tuning task," in *Proc. of the Tenth Workshop on Statistical Machine Translation*, Lisbon, Portugal, September 2015, pp. 422–427.

[17] F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models," *Computational Linguistics*, vol. 29, no. 1, pp. 19–51, March 2003.

[18] M. Galley and C. D. Manning, "A simple and effective hierarchical phrase reordering model," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, ser. EMNLP '08, 2008, pp. 848–856.

[19] N. Durrani, H. Schmid, and A. Fraser, "A joint sequence translation model with integrated reordering," in *Proc. of the 49th Annual Meeting of the Association for Computational Linguistics (ACL '11)*, Portland, Oregon, June 2011, pp. 1045–1054.

[20] 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*. Association for Computational Linguistics, 2011, pp. 355–362.

[21] A. Axelrod, Y. Vyas, M. Martindale, M. Carpuat, and J. Hopkins, "Class-based n-gram language difference models for data selection," in *IWSLT (International Workshop on Spoken Language Translation)*, Hanoi, Vietnam, December 2015.

[22] K. Toutanova, D. Klein, C. D. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in *In Proceedings of HLT-NAACL 2003*, 2003, pp. 252–259.

[23] J. Dehdari, L. Tan, and J. van Genabith, "BIRA: Improved predictive exchange word clustering," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*. San Diego, CA, USA: Association for Computational Linguistics, June 2016, pp. 1169–1174.

[24] R. Sennrich, B. Haddow, and A. Birch, "Edinburgh neural machine translation systems for wmt 16," in *Proceedings of the First Conference on Machine Translation*. Berlin, Germany: Association for Computational Linguistics, August 2016, pp. 371–376.

[25] G. Neubig, "Lamtram: A toolkit for language and translation modeling using neural networks," http://www.github.com/neubig/lamtram, 2015.

[26] C. Dyer, V. Chahuneau, and N. A. Smith, "A simple, fast, and effective reparameterization of ibm model 2," in *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Atlanta, Georgia, June 2013.

[27] M. Junczys-Dowmunt, T. Dwojak, and R. Sennrich, "The AMU-UEDIN submission to the WMT16 news translation task: Attention-based nmt models as feature functions in phrase-based smt," in *Proceedings of the First Conference on Machine Translation*. Berlin, Germany: Association for Computational Linguistics, August 2016, pp. 319–325.

[28] J.-T. Peter, T. Alkhouli, H. Ney, M. Huck, F. Braune, A. Fraser, A. Tamchyna, O. Bojar, B. Haddow, R. Sennrich, F. Blain, L. Specia, J. Niehues, A. Waibel, A. Allauzen, L. Aufrant, F. Burlot, e. knyazeva, T. Lavergne, F. Yvon, M. Pinnis, and S. Frank, "The qt21/himl combined machine translation system," in *Proceedings of the First Conference on Machine Translation*. Berlin, Germany: Association for Computational Linguistics, August 2016, pp. 344–355. [Online]. Available: http://www.aclweb.org/anthology/W/W16/W16-2320

[29] M. Freitag, M. Huck, and H. Ney, "Jane: Open source machine translation system combination," in *Conference of the European Chapter of the Association for*

Computational Linguistics, Gothenburg, Sweden, April 2014, pp. 29–32.

[30] O. Klejch, E. Avramidis, A. B. Burchart, and M. Popel, "MT-compareval: Graphical evaluation interface for machine translation development," *Prague Bull. Math. Linguistics*, vol. 104, pp. 63–74, 2015.

[31] M. Kazi, M. Coury, E. Salesky, J. Ray, W. Shen, T. Gleason, T. Anderson, G. Erdmann, L. Schwartz, B. Ore, R. Slyh, J. Gwinnup, K. Young, and M. Hutt, "The MIT-LL/AFRL IWSLT-2013 MT system," in *Proceedings of the International Workshop on Spoken Language Translation*, Heidelberg, Germany, 2013.

[32] Theano Development Team, "Theano: A Python framework for fast computation of mathematical expressions," *arXiv e-prints*, vol. abs/1605.02688, May 2016. [Online]. Available: http://arxiv.org/abs/1605.02688

[33] G. Dahl, D. Yu, L. Deng, and A. Acero, "Context-Dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, pp. 30–42, January 2012.

[34] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi Speech Recognition Toolkit," in *Proceedings of the Automatic Speech Recognition and Understanding Workshop*, 2011.