The MITLL-AFRL IWSLT 2014 MT System[†]

Michaeel Kazi, Elizabeth Salesky, Brian Thompson, Jessica Ray, Michael Coury, Wade Shen

MIT Lincoln Laboratory
Human Language Technology Group
244 Wood Street
Lexington, MA 02420, USA
{michaeel.kazi, elizabeth.salesky, brian.thompson, jessica.ray, michael.coury, swade}@ll.mit.edu

Tim Anderson, Grant Erdmann, Jeremy Gwinnup, Katherine Young, Brian Ore, Michael Hutt

Air Force Research Laboratory
Human Effectiveness Directorate
2255 H Street
Wright-Patterson AFB, OH 45433
{timothy.anderson.20, grant.erdmann, jeremy.gwinnup.ctr, katherine.young.1.ctr, brian.ore.ctr, michael.hutt.ctr}@us.af.mil

Abstract

This report summarizes the MITLL-AFRL MT and ASR systems and the experiments run using them during the 2014 IWSLT evaluation campaign. Our MT system is much improved over last year, owing to integration of techniques such as PRO and DREM optimization, factored language models, neural network joint model rescoring, multiple phrase tables, and development set creation. We focused our efforts this year on the tasks of translating from Arabic, Russian, Chinese, and Farsi into English, as well as translating from English to French. ASR performance also improved, partly due to increased efforts with deep neural networks for hybrid and tandem systems. Work focused on both the English and Italian ASR tasks.

1. Introduction

During the evaluation campaign for the 2014 International Workshop on Spoken Language Translation (IWSLT'14) [1] our experimental efforts in machine translation (MT) centered on 1) decoding with factored language models [2], 2) neural network joint model [3] rescoring, 3) multiple phrase tables, and 4) development set creation. Other algorithms in our toolbox included the recurrent neural network language model [4], and the operational sequence models [5].

Experimental efforts for the automatic speech recognition (ASR) task focused on the use of deep neural networks for use in both hybrid and tandem configurations. Updated language models also improved performance compared to our 2013 system.

[†]This work is sponsored by the Air Force Research Laboratory under Air Force contract FA8721-05-C-0002. Opinions, interpretations, conclusions and recommendations are those of the authors and are not necessarily endorsed by the United States Government.

We here describe improvements over our 2013 submission systems. For a more in-depth description of the 2013 system, refer to [6]. This paper is structured as follows. Section 2 presents our work on the MT task, and discusses each of the techniques mentioned above, ending with a discussion of submitted systems. Our work on the ASR task is discussed in Section 3.

2. Machine Translation

2.1. Data usage

Unless otherwise noted, data described in this section originates from the WMT14 website[1]. We used the in-domain data supplied by WIT3 [7] for all language pairs. In English-French, our parallel data included the 10^9 corpus, News Commentary v8, Europarl v7, and the UN corpus. In Russian to English, we used the Yandex corpus[2], Common Crawl, Wiki Headlines, News Crawl, and UN data. In Arabic to English, we used only the UN data, which was sentence-aligned via Champollion [8].

Extra monolingual data (in addition to parallel data) included the News Crawl corpus 2007-2011 (English and French), LDC Gigaword English v5 [9], and LDC French Gigaword v3 [10].

2.2. Data Preprocessing and Cleanup

The TED datasets were examined for repetition errors, in which English sentences or sentence-internal phrases are translated multiple times. These errors derive from the TED website. When repetition errors occur in training data, they cause alignment problems; when they occur in test data, they degrade the machine translation.

[1] http://www.statmt.org/wmt14/translation-task.html
[2] https://translate.yandex.ru/corpus?lang=en

Repeated phrases of more than 10 words were detected and removed. If parallel text was available, phrases were only removed when there was no corresponding repetition in the English sentence. The Farsi test sets contained substantial repetition; lesser amounts were found in the Chinese dev and test data, and in the French dev data. Arabic and Russian dev and test sets were also examined, but did not contain these repetitions. Removing the repetitions from the Farsi `tst2014` file improved BLEU +1.53, based on last year's IWSLT system. We expect to see some improvement for Chinese as well, but due to time constraints defer that comparison to future work. Repeat statistics for the dev and test sets are outlined in Table 1, and for the train sets in Table 2.

Lang.	Set	Repeats	Length
French	`dev2010`	11	887
Chinese	`dev2010`	87	887
	`tst2010`	81	1570
	`tst2014`	13	1068
Farsi	`tst2010`	1	885
	`tst2011`	22	1132
	`tst2012`	343	1375
	`tst2013`	187	923
	`tst2014`	53	1131

Table 1: Repeated sentences per dev/test set

Lang.	Year	Repeats	Length
Arabic	2013	3	155,047
	2014	5	186,467
Chinese	2014	550	177,901
Farsi	2013	5,749	81,872
	2014	8,987	112,704
French	2013	173	162,681
	2014	373	186,510
Russian	2013	109	135,669
	2014	145	185,205

Table 2: Repeated sentences per training set

2.3. Baseline MT System

Our system implements a fairly standard phrase-based SMT [11] architecture. It consists of the following:

- Training corpora filtered for maximum sentence length of 40.
- MADAMIRA Beta 1.0 [12] tokenization for Arabic, Stanford Segmenter [13] + character segmentation for Chinese, Moses tokenizer for Russian and English.
- GIZA++ word alignments, using 100 word classes, Models 2-4 + HMM and optionally Model 5.
- Order 6 TED language model.
- Maximum extracted phrase length of 9.
- Monotone-at-punctuation, drop-unknown.
- Phrasetable with KN smoothing [14].
- Word-based [15] or hierarchical [16] monotone-swap-distort lexical reordering.
- Moses decoder [17], no reordering over punctuation, n-best list size 200.
- Rescore n-best-lists using order-7 class-based TED LM. Default is 80 word classes.
- Pairwise rank optimization [18] or Derivative-Free Robust Error Minimization (DREM) [6] over cumulative n-best lists.
- One-best result (we saw no consistent benefit to using Minimum Bayes Risk).

In addition to the tokenizers listed above, in English-French and the English component of the Arabic task, we used simple in-house tokenizers that separate out punctuation and common language specific constructions (e.g. l' in French). Reported scores are case-sensitive BLEU scores with separated punctuation (via MTEval[3]). To account for variance, unless otherwise stated, scores are averages over 10 optimizations. Baseline systems are tuned on `dev2010`.

2.3.1. Language Modeling

Language models on in-domain or target-side parallel data were trained using either MITLM [19] or SRILM 1.7 [20]. With the Gigaword dataset, we typically used `lmplz` [21]. All LMs were binarized using KenLM [22]. Word classes were trained using `mkcls` [23].

2.4. Additional Phrase Table Training

The use of extra phrase table training data was indispensible in the English to French and Russian to English tasks. For each of these, we used Moore-Lewis [24] cross-entropy filtering cE) and kept 10% of the out-of-domain data. We also experimented with a 2nd phrase table in Arabic to English and Russian to English using the MultiUN and Yandex datasets, respectively. These were tested in addition to a cross-entropy filtered PT.

Lang.	Baseline	cE PT	+ 2nd PT	+Backoff PT
en-fr	38.25†	41.39†	–	–
ar-en	30.94	31.55	31.53	30.66
ru-en	21.13	22.47	22.15	21.25

Table 3: Comparison of mean BLEU on `tst2013` with additional PT training. (†=`tst2012`)

2.5. Neural Network Joint Model

We replicated the architecture described in Devlin et al. Neural Network Joint Model [3], which is similar to a

[3] `http://www.itl.nist.gov/iad/mig/tests/mt/2009/`

continuous space language model, but conditioned on words in the source language as well. Each target side word is considered to be "affiliated" with a source word (via word alignments included in the phrase table). The affiliated word, the 5 words before and after it, and a 3-gram on the target side are input to the neural network; the outputs are posterior probabilities over the entire target language vocabulary.

We implemented this to rescore 200-best lists. Our results were promising; we saw modest gains on a variety of language pairs. Devlin et al. claim gains more than double when this is integrated into the decoder itself. This is future work for us.

We implemented the NNJM within Theano [25], and ran training and rescoring on a Tesla K40 GPU. We trained a vocabulary by taking words seen in TED 4 or more times. Additional words in the phrase table (such as from out-of-domain data) were mapped to word classes using `mkcls`. Training was done on the output of grow-diag-final-and alignments. In the case where out-of-domain data was available and useful, we first trained the network on only the out-of-domain data, then switched to in-domain data only. In building the phrase table, sub-phrase alignments for a given phrase pair were taken from the extracted phrase pair with maximum scoring lexical $p(f|e)$.

Lang.	RNNLM	NNJM-In	Out+In
ar-en	30.59	30.87	30.88
en-fr	40.85	39.75	41.39
ru-en	20.81	21.21	21.27

Table 4: Effects of neural joint model rescoring, mean BLEU over `tst2012`

2.6. Factored Models

Following the success of Edinburgh's Target Sequence Model [2] (and our own rescoring n-best lists via `mkcls`), we enabled factored language models within Moses. In theory this should be better than rescoring, because it will alter the search space the decoder traverses. For class-based LMs, we compared `mkcls` to Percy Liang's `brown-cluster`[4]. We saw that the optimal number of word classes varied, but once tuned, BLEU varied only 0.2% on ru-en. All numbers reported here use `mkcls`.

Lang.	Baseline	50	200	600	1000
ar-en	29.61	29.70	29.58	29.70	29.71
fa-en	16.68	16.26	16.54	16.62	16.87
ru-en	18.75	19.03	19.32	19.45	19.16
zh-en	15.06	14.95	14.64	14.80	15.00

Table 5: nClasses with factored LMs, `tst2013`.

[4] `https://github.com/percyliang/brown-cluster`

We saw further gains of 0.41, 0.37 and 1.2 with additional class-based Gigaword LMs in ar-en, fa-en, and ru-en, respectively. However, the results for zh-en were inconsistently bad. For instance, we saw a gain of 0.29 with 200 classes, a loss of 0.39 with 1000 classes, and all experiments were worse than the baseline score. Additional factored LMs, such as POS tags, were tried in Russian to English, but produced a loss in performance of 0.6 BLEU.

We also experimented with the operational sequence model over word classes. We saw significant gain in English to French using only TED data (+0.69 on `tst2010` using 100 classes), but using the full out-of-domain data, we did not see the same gains (+0.16). Translating into English, we saw limited gains, but OSM with classes reduced std. deviation >1.0 BLEU.

Lang.	Baseline	100	250	500	1000
en-fr	41.39	41.56	40.21	–	–
zh-en	12.85	12.89	12.76	12.83	12.98

Table 6: nClasses with OSM w/WCs, `tst2012` for en-fr, `tst2013` for zh-en.

2.7. Russian Morphological Preprocessing

We used a variation of the Yandex technique for reducing data sparsity [21], stemming nouns and adjectives and inserting a case element as a separate word before each noun. We used `mystem`[5] to identify lemmas and grammatical information; nouns were annotated for number, and adjectives were annotated for degree. Noun forms that could represent singular or plural were annotated as singular. For nouns with ambiguous case, the first possible case element was selected from the continuum of nominative, accusative, genitive, dative, instrumental, ablative. Examples are shown in Table 7.

Noun	Case/Number	Output
дням	dat-pl	DAT день.N+PL
день	nom-sg, acc-sg	NOM день.N+SG

Table 7: Examples of Yandex-style morphological processing

Table 8 shows average BLEU gains over 10 runs by preprocessing the Russian source data in this way. Max scores increased less, on average 0.27, while standard deviation decreased significantly. These trends extended to experiments with extra data, and were exaggerated with the addition of NNJM rescoring.

[5] https://api.yandex.ru/mystem/

System	BLEU	Gain	Δ Stdev
Baseline	21.13	+0.32	-0.2
+outd	23.29	+0.82	-0.3
+RescoreNNJM	23.56	+1.45	-1.14

Table 8: Mean BLEU scores with Yandex-style preprocessing, `tst2013`.

2.8. Farsi-English System

Our system this year was a factored phrase-based system built using supplied in-domain data for the phrase table with 3 language models built using Gigaword, in-domain data, and Google-book n-grams. Gains were obtained by replacing non-printable characters with spaces, utilizing class-factors with 600 classes, using the cleaned test sets as described in Section 2.2, and optimizing with a development set as described in Section 2.9. We selected the number of sentences for these sets based on the maximum Tversky score. Three sets were created, one each to match `tst2013` and `tst2014` and one to match the combination. Non-printing characters were replaced and repeated phrases (Section 2.2) removed before the devset selection occurred. Systems were optimized with PRO using each of these devsets and the best score on `tst2012` of 10 runs was selected as the configuration for submission (see Table 9).

Dev Set	Length	`tst2012`		
		Mean	Stdev	Max
`dev2010`	885	20.52	0.22	20.16
`tst2012`	1375	20.48	0.09	20.60
`tst2013devsel`	931	20.94	0.16	21.23
`tst2014devsel`	888	21.22	0.10	21.34
`tst2014+13devsel`	1245	20.99	0.16	21.23

Table 9: Farsi-English system BLEU scores on regular and Tversky-selected devsets

Based on these results, the system optimized with `tst2014devsel` was used to decode `tst2013` and `tst2014` for submission.

2.9. Development Set Creation

Following the experiments from last year, as well as uncertainty in performance via optimizing `dev2010` or `tst2011`, we implemented a dev set creation mechanism which extracts the most promising segments from the available data. We choose to select the dev set based on maximizing the Tversky similarity measure [26] between the dev set source segments and the test set source segments. We employ Tversky similarity with unit weights, making it equivalent to Jaccard similarity and Tanimoto similarity: our Tversky score is the number of unique words in the intersection of the dev and test sets divided by the number of unique words in the union.

We create the dev set via greedy optimization. Starting with an empty dev set, we iteratively add the segment that provides the largest bang-for-your-buck improvement, i.e., the largest increase in Tversky similarity divided by the number of words in the segment. The result is a dev set with segments ordered by relationship to the test set. We can choose a fixed dev set size based on available resources, a dev set size that maximizes Tversky similarity, or use another heuristic.

In order to test effectiveness of the Tversky metric, baseline systems were trained using only in-domain data for Arabic-English, Russian-English, and Chinese-English language pairs. These systems were then optimized using `dev2010`, `tst2012` and Tversky-selected dev sets of varying length (e.g. `tvdev1188` for Arabic indicating a dev set selected from the first 1,188 lines of the selected data). The pool of possible sentence pairs for the Tversky-selected dev sets is the concatenation of `dev2010`, `tst2010`, `tst2011`, and `tst2012`. The length of these selected sets is set by maximizing the score for the source-side of `tst2014`. (It is worth mentioning that the references play no role in the entire process.) Results are shown in Table 10.

Lang.	dev set	avg BLEU	max BLEU
ar-en	`dev2010`	20.42	20.96
	`tst2012`	20.64	20.94
	`tvdev1188`	21.15	21.52
ru-en	`dev2010`	16.98	17.03
	`tst2012`	16.81	17.03
	`tvdev2500`	17.00	17.13
zh-en	`dev2010`	12.60	12.90
	`tst2012`	12.33	13.03
	`tvdev1500`	11.30	12.92

Table 10: Results of Baseline systems using standard and Tversky-score selected dev sets.

2.10. MT Submission Systems

A brief description and results for all of our MT submission systems can be found in Table 11.

3. ASR

3.1. English ASR

A hybrid Deep Neural Network (DNN)-HMM speech recognition system was developed on 166 hours of TED data, 128 hours from the HUB4 corpus [27, 28], and 96 hours from the Euronews corpus provided by the organizers. This system was trained using the same procedure as our IWSLT 2013 system [6]. The DNNs included 7 hidden layers with 1000 units each and 8000 output units. Compared to our IWSLT 2013 hybrid

System	Description	tst2012	tst2013	tst2014
English-to-French				
primary	cE apw/afp/ted/news LMs, NNJMout+in, OSM o9, opt tvDev1500	42.62		
contrast1	primary – tvDev + opt dev2010	41.80		
Arabic-to-English				
primary	2PTs, hier-msd, nyt+news LM, NNJMin, ted-200 cLM, nyt-600 cLM	30.86	31.80	27.70
contrast1	primary – dev2010 + opt tvDev1200	31.11	31.72	27.39
Chinese-to-English				
primary	nyt LM, dLimit-8, hier-msd LR, max sent len 32	13.83	15.67	12.90
contrast1	primary + ltw LM + 150 classes GIZA	14.20	15.44	13.25
contrast2	primary + tvDev1500	14.09	15.43	12.92
contrast3	primary + hier-mslr LR	13.28	15.59	12.64
Farsi-to-English				
primary	PRO, cleaned source data, 600 cLM o7, hiero LR reordering, nyt LM o7, google book o5, opt tvdev2014	21.13	19.49	18.45
contrast1	primary – tvdev2014 + opt tvdev2013	21.12	19.24	18.56
contrast2	primary – tvdev2014 + opt tvdev2013+2014	21.11	19.14	18.27
Russian-to-English				
primary	PRO, cE PT, ted LM o7, outd LM o7, giga LM o5, ted+outd cLM o7, giga nyt cLM o7, yandex parsing, NNJMout+in, opt on dev2010	21.30	24.42	19.45
contrast1	primary – yandex parsing	21.27	24.10	19.08

Table 11: MT Submission Systems.

DNN-HMM system trained on TED, the additional data yielded a 1.2% Word Error Rate (WER) improvement on dev2012 prior to LM rescoring, and a 0.4% WER improvement after LM rescoring.

A bottleneck [29] DNN system for use with a tandem GMM-HMM [30] was trained using 135 hours of TED data. The Theano library for Python [25] was leveraged during DNN training to enable use of the GPU. The final DNN had 4 hidden layers with 1000 units, plus an additional bottleneck layer with 60 units placed between the last two hidden layers. The DNN was trained with 12 Perceptual Linear Prediction features, along with the zeroth coefficient and first, second, and third order differentials. Features were combined with a frame window of 13 to give a total input size of 676. Outputs corresponded to 6000 shared states. A minibatch size of 256 and initial learning rate of 0.3 was used for training the DNN. The "newbob" learning rate schedule as used in [31] was followed.

A tandem GMM-HMM was trained with the bottleneck features, which were run through PCA. The final tandem model included approximately 7000 shared states with 32 Gaussians per state. This sytem did not perform as well as the hybrid system, but was successful in system combination.

LM data selection was implemented using the same procedure as our IWSLT 2012 system [32]. Interpolated trigram and 4-gram LMs were estimated on TED, 1/8 of Gigaword, and 1/8 of News 2007–2013 using the SRILM Toolkit [20].A Recurrent Neural Network (RNN) maximum entropy LM was estimated on the same set of training texts using the RNNLM Toolkit [4]. The network included 160 hidden units, 300 classes in the output layer, 4-gram features for the direct connections, and a hash size of 10^9. The LM vocabulary included 100000 words.

In addition to the hybrid DNN-HMM and tandem systems described above, we also used our IWSLT 2013 HMM acoustic models (AMs) with the updated LMs. This system was cross adaptated using the initial transcripts from the hybrid DNN-HMM system.

Automatic segmentation of the test data was performed using the same procedure as IWSLT 2013. Recognition lattices were produced for each system and then rescored with the interpolated 4-gram LM. Next, 1000-best lists were extracted from each lattice and rescored with the RNN LM. The final LM scores were obtained by linearly interpolating the log probabilities from the 4-gram and RNN LM. Lastly, system combination was perfomed using N-best ROVER.

Table 12 shows the WER of each system on dev2012 after evaluating the second pass decoder, rescoring with the 4-gram LM, and interpolating the 4-gram and RNN LM scores. Note that the first pass hybrid DNN-HMM and tandem systems yielded a 16.7% and 23.1% WER on dev2012, respectively. N-best ROVER of all three systems yielded a 12.4% WER.

3.2. Italian ASR

An Italian pronunciation dictionary was manually created for the most frequent 28000 words from the Eu-

System	Decode-2	4-gram	4-gram + RNN
DNN-HMM	14.8	14.2	13.3
HMM-2013-AM	15.3	14.6	13.7
Tandem	20.8	20.0	18.3

Table 12: English `dev2012` WER. Results are shown for each system after evaluating the second pass decoder, rescoring with the 4-gram LM, and interpolating the 4-gram and RNN LM scores.

ronews corpus. This was done by a member of our group who speaks Italian as a second language. The 51 phone set included 24 non-geminated consonants, 20 geminated consonants, and 7 vowels. A second pronunciation dictionary with 32 phones was created by ignoring gemination.[6] Lastly, a multilingual (ML) pronunciation dictionary was created from the Italian dictionary that ignored gemination and version 0.7a of the English CMU pronunciation dictionary. Italian and English phones were merged when they shared the same IPA symbol;[7] this dictionary included 48 phones.

HMM and hybrid DNN-HMM systems were trained on the Euronews Italian data set using the same procedure as the English systems. One HMM system was trained using the 51 phone set (denoted as HMM-51), and a second HMM system was trained using the the 32 phone set (denoted as HMM-32). HMM-51 included 6000 shared states with an average of 28 mixtures per state, and HMM-32 included 4000 shared states with an average of 24 mixtures per state. The hybrid DNN-HMM system was developed using HMM-51, and the DNNs included 3 hidden layers with 1000 units each and 6000 output units. A final HMM system (denoted as HMM-ML) was developed on Euronews Italian and TED English using the ML pronunciation dictionary; HMM-ML included 6000 shared states with an average of 28 mixtures per state.

Interpolated trigram and 4-gram LMs were estimated on the provided TED training data, Google Books Ngram corpus, and Web 1T 5-gram corpus. Words from the TED data set were split on apostrophes, and N-grams from Google Books were ignored if the source was published prior to the year 2000. The LM vocabulary included 100000 words. An RNN maximum entropy LM was estimated on TED using the RNNLM Toolkit. The network included 320 hidden units, 300 classes in the output layer, 4-gram features for the direct connections, and a hash size of 10^9.

Initial segments of the test data were created using the English neural network SAD. On the `dev2014` partition, it was discovered that the SAD was misclassifying non-speech sections as speech on several TEDx talks. To alleviate this problem, we reprocessed any speech segment longer than 20 seconds with a second SAD that was trained on English telephone speech from the Fisher corpus [33].

Each system was evaluated using HDecode and LM rescoring was performed using the same procedure described in Section 3.1. Cross adaptation was applied to the HMM systems using the initial transcripts from the hybrid DNN-HMM system. The final hypothesis was selected via N-best ROVER of the DNN-HMM, HMM-32 and HMM-ML systems. This combination yielded a 29.5% WER on `dev2014`. Table 13 shows the WER at each decoding stage; for comparison purposes, we have included the results obtained without cross adaptation of the HMM systems.

3.3. ASR Submission Systems

Final submissions on English `tst2014` and `tst2013` and Italian `tst2014` are shown in Table 14.

4. Acknowledgements

The authors would like to thank Tina May and Wahid Abdul Qudus for their efforts in spot-checking Chinese and Farsi dataset processing, respectively. We would also like to thank Kyle Wilkinson for creating the Italian pronunciation dictionary.

5. References

[1] M. Federico, M. Cettolo, L. Bentivogli, M. Paul, and S. Stüker, "Overview of the IWSLT 2014 Evaluation Campaign," ser. Proceedings of IWSLT, 2014.

[2] A. Birch, N. Durrani, and P. Koehn, "Edinburgh SLT and MT system description for the iwslt 2013 evaluation," *Proc. IWSLT, Heidelberg, Germany*, 2013.

[3] J. Devlin, R. Zbib, Z. Huang, T. Lamar, R. Schwartz, and J. Makhoul, "Fast and robust neural network joint models for statistical machine translation," ser. Proceedings of the ACL, Long Papers, 2014.

[4] T. Mikolov, A. Deoras, D. Povey, L. Burget, and J. Černocký, "Strategies for training large scale neural network language models," ser. Automatic Speech Recognition and Understanding Workshop, 2011.

[6] Palatal nasal consonants were always geminated in our dictionary.

[7] The ARPAbet to IPA mappings used in this work are available at: http://en.wikipedia.org/wiki/Arpabet

Opinions, interpretations, conclusions and recommendations are those of the authors and are not necessarily endorsed by the United States Government. Cleared for public release on 6 Nov 2014. Originator Reference Number: SA-14-113170. Case Number: 88ABW-2014-5156.

	Without Cross Adaptation					With Cross Adaptation		
System	Decode-1	Decode-2	4-gram	4-gram + RNN		Decode-2	4-gram	4-gram + RNN
DNN-HMM	35.0	32.9	32.5	32.5		32.9	32.5	32.5
HMM-32	41.2	34.4	34.1	33.9		32.2	31.8	31.4
HMM-51	41.2	35.1	34.8	34.5		32.2	31.9	31.8
HMM-ML	42.7	35.9	35.7	35.4		32.4	32.3	32.3
N-best ROVER	35.2	31.3	30.8	30.8		30.1	29.7	29.5

Table 13: Italian dev2014 WER. N-best ROVER was applied at each decoding stage using 1000-best lists from the the hybrid DNN-HMM, HMM-32, and HMM-ML systems. Results are shown both with and without cross adaptation of the HMM systems.

System	Description	tst2013	tst2014
English			
primary	N-best ROVER with hybrid DNN-HMM, HMM Sphinx-4, and tandem systems	14.3*	10.0**
contrast1	Hybrid DNN-HMM using Viterbi decoding	15.6*	11.2**
Italian			
primary	N-best ROVER with hybrid DNN-HMM, HMM-32, and HMM-ML systems	–	24.7

Table 14: All submission systems for English and Italian ASR. *Unofficial scores using last year's tst2013 reference files with minor corrections. **Unofficial scores using suggested tst2014 STM and GLM corrections.

[5] N. Durrani, H. Schmid, and A. Fraser, "A joint sequence translation model with integrated reordering," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, ser. HLT '11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 1045–1054.

[6] 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 *The 10th International Workshop on Spoken Language Translation (IWSLT '13)*, Heidelberg, Germany, December 2013, pp. 136–143.

[7] M. Cettolo, C. Girardi, and M. Federico, "WIT3: Web Inventory of Transcribed and Translated Talks," ser. Proceedings of EAMT, 2012, pp. 261–268.

[8] X. Ma, "Champollion: A robust parallel text sentence aligner," ser. Proceedings of LREC, 2006.

[9] R. Parker, D. Graff, J. Kong, K. Chen, and K. Maeda, "English Gigaword Fifth Edition LDC2011T07," *Philadelphia: Linguistic Data Consortium*, 2011.

[10] D. Graff, Ângelo Mendonça, and D. DiPersio, "French Gigaword Third Edition LDC2011T10." *Philadelphia: Linguistic Data Consortium*, 2011.

[11] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, ser. NAACL '03. Stroudsburg, PA, USA: Association for Computational Linguistics, 2003, pp. 48–54.

[12] N. Habash, R. Roth, O. Rambow, R. Eskander, and N. Tomeh, "Morphological analysis and disambiguation for dialectal Arabic." in *HLT-NAACL*. The Association for Computational Linguistics, 2013, pp. 426–432.

[13] P.-C. Chang, M. Galley, and C. D. Manning, "Optimizing Chinese word segmentation for machine translation performance," in *Proceedings of the Third Workshop on Statistical Machine Translation*, ser. StatMT '08. Stroudsburg, PA, USA: Association for Computational Linguistics, 2008, pp. 224–232.

[14] G. Foster, R. Kuhn, and H. Johnson, "Phrasetable smoothing for statistical machine translation," in *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, ser. EMNLP '06. Stroudsburg, PA, USA: Association for Computational Linguistics, 2006, pp. 53–61.

[15] C. Tillmann, "A unigram orientation model for statistical machine translation," in *Proceedings of HLT-NAACL 2004: Short Papers*, ser. HLT-NAACL-Short '04. Stroudsburg, PA, USA: Association for Computational Linguistics, 2004, pp. 101–104.

[16] 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. Stroudsburg, PA, USA: Association for Computational Linguistics, 2008, pp. 848–856.

[17] 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 ACL on Interactive Poster and Demonstration Sessions*, ser. ACL '07. Stroudsburg, PA, USA: Association for Computational Linguistics, 2007, pp. 177–180.

[18] M. Hopkins and J. May, "Tuning as ranking," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, ser. EMNLP '11. Stroudsburg, PA, USA: Association for Computational Linguistics, 2011, pp. 1352–1362.

[19] B.-J. P. Hsu and J. Glass, "Iterative language model estimation: Efficient data structure and algorithms," ser. Interspeech, 2008.

[20] A. Stolcke, "Srilm-an extensible language modeling toolkit," in *Proceedings International Conference on Spoken Language Processing*, November 2002, pp. 257–286.

[21] K. Heafield, I. Pouzyrevsky, J. H. Clark, and P. Koehn, "Scalable modified Kneser-Ney language model estimation," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, Sofia, Bulgaria, August 2013, pp. 690–696. [Online]. Available: http://kheafield.com/professional/edinburgh/estimate_paper.pdf

[22] K. Heafield, "KenLM: faster and smaller language model queries," in *Proceedings of the EMNLP 2011 Sixth Workshop on Statistical Machine Translation*, Edinburgh, Scotland, United Kingdom, July 2011, pp. 187–197.

[23] F. J. Och, "An efficient method for determining bilingual word classes," in *Proceedings of the Ninth Conference on European Chapter of the Association for Computational Linguistics*, ser. EACL '99. Stroudsburg, PA, USA: Association for Computational Linguistics, 1999, pp. 71–76.

[24] R. C. Moore and W. Lewis, "Intelligent selection of language model training data," in *Proceedings of the ACL 2010 Conference Short Papers*, ser. ACLShort '10. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010, pp. 220–224.

[25] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley, and Y. Bengio, "Theano: a CPU and GPU math expression compiler," *Proceedings of the Python for Scientific Computing Conference (SciPy)*, 2010.

[26] A. Tversky, "Features of similarity," *Psychological Review*, vol. 84, pp. 327–352, 1977.

[27] D. Graff, Z. Wu, R. MacIntyre, and M. Liberman, "The 1996 broadcast news speech and language-model corpus," in *Proceedings of the DARPA Workshop on Spoken Language technology*. Citeseer, 1997, pp. 11–14.

[28] J. Fiscus, J. Garofolo, M. Przybocki, W. Fisher, and D. Pallett, "English broadcast news speech (hub4)," *Linguistic Data Consortium, Philadelphia*, 1997.

[29] F. Grezl, M. Karafiát, S. Kontár, and J. Cernockỳ, "Probabilistic and bottle-neck features for LVCSR of meetings." in *ICASSP (4)*, 2007, pp. 757–760.

[30] H. Hermansky, D. P. Ellis, and S. Sharma, "Tandem connectionist feature extraction for conventional HMM systems," in *Acoustics, Speech, and Signal Processing, 2000. ICASSP'00. Proceedings. 2000 IEEE International Conference on*, vol. 3. IEEE, 2000, pp. 1635–1638.

[31] D. Johnson *et al.*, "ICSI quicknet software package," *http://www.icsi.berkeley.edu/Speech/qn.html*, 2004.

[32] J. Drexler, W. Shen, T. Gleason, T. Anderson, R. Slyh, B. Ore, and E. Hansen, "The MIT-LL/AFRL IWSLT-2012 MT system," ser. Proceedings of IWSLT, 2012.

[33] C. Cieri, D. Graff, O. Kimball, D. Miller, and K. Walker, "Fisher English training parts 1 and 2, speech and transcripts," *Linguistic Data Consortium, Philadelphia*, 2005.