INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Language Model Data Augmentation for Keyword Spotting in Low-Resourced Training Conditions

Arseniy Gorin[1], *Rasa Lileikytė*[1], *Guangpu Huang*[1], *Lori Lamel*[1],
Jean-Luc Gauvain[1], *Antoine Laurent*[2]

[1]LIMSI, CNRS, Université Paris–Saclay, 508 Campus Universitaire F–91405 Orsay, France
[2]Vocapia Research, 28 rue Jean Rostand, 91400 Orsay, France
[1]{gorin,lileikyte,huang,lamel,gauvain}@limsi.fr, [2]laurent@vocapia.com

Abstract

This research extends our earlier work on using machine translation (MT) and word-based recurrent neural networks to augment language model training data for keyword search in conversational Cantonese speech. MT-based data augmentation is applied to two language pairs: English-Lithuanian and English-Amharic. Using filtered N-best MT hypotheses for language modeling is found to perform better than just using the 1-best translation. Target language texts collected from the Web and filtered to select conversational-like data are used in several manners. In addition to using Web data for training the language model of the speech recognizer, we further investigate using this data to improve the language model and phrase table of the MT system to get better translations of the English data. Finally, generating text data with a character-based recurrent neural network is investigated. This approach allows new word forms to be produced, providing a way to reduce the out-of-vocabulary rate and thereby improve keyword spotting performance. We study how these different methods of language model data augmentation impact speech-to-text and keyword spotting performance for the Lithuanian and Amharic languages. The best results are obtained by combining all of the explored methods.

Index Terms: speech recognition, text augmentation, language modeling, machine translation, low-resourced languages

1. Introduction

Language modeling under low-resourced training conditions is generally associated with high out-of-vocabulary (OOV) rates and non-reliable parameter estimation, resulting in poor performance of speech-to-text (STT) and keyword spotting (KWS) systems.

Various text resources (Web, newspapers, etc.) are frequently used for language model (LM) training. However, locating and selecting texts is challenging for conversational telephone speech (CTS) because of its specific syntactic and semantic nature. Several approaches were investigated [1, 2, 3] in the context of IARPA Babel project [4]. Here subtitles are shown to be more useful than, for example, Wikipedia data. Larger improvements are reported when data are collected by querying the Web using conversational transcripts.

Alternatively, text data from well-resourced languages can be exploited to improve the language model of a low-resourced one. In [5] a Chinese language model is improved by applying machine translation (MT) from English, and in [6] this is done by using only document-aligned comparable texts. Authors of [7] and [8] report on improving weather forecast LM with English-Icelandic and French-Romanian MT.

In [9], using translations of Mandarin CTS in LM improved the performance of Cantonese STT and KWS system. Additional improvement was demonstrated by adding texts generated with word-level recurrent neural networks (RNNs) [10].

This work extends [9] in three ways. First, MT based augmentation is investigated for two other languages: Lithuanian and Amharic. For both languages an MT model is trained on parallel data with Moses toolkit [11] and then used to translate a large corpus of English CTS transcripts. Compared to closeness of Mandarin and Cantonese, both Lithuanian and Amharic are quite different from English. As for Amharic, only limited out-of-domain data are available for MT training. Second, N-best translations are extracted to enlarge LM training data, which results in larger improvements of STT and KWS performance over just using the 1-best translation. Third, non-parallel Web texts are used to improve the MT quality either by enriching the language model in the MT system, or by enlarging phrase table in a semi-supervised manner.

An alternative to using Web and MT text resources is to generate texts with a character-level RNN [12]. Several examples of generated poetry and mathematical articles were demonstrated in [13], and in [14, 15] end-to-end lexicon-free decoders using a character RNN without lexicon and LM, achieved comparable results to the conventional pipeline. In this work texts are generated with the character-level RNN to enlarge both lexicon and LM training data.

2. Languages and data

This research is based on two languages from the Babel project: Lithuanian (IARPA-babel304b-v1.0b) and Amharic (IARPA-babel307b-v1.0b).

2.1. Speech data and transcripts

Lithuanian is a language from Baltic subgroup of Indo-European family. It is based on the Latin alphabet with a few additional characters (32 letters in total). Amharic belongs to the transversal sub-branch of South Ethiopian languages (an offshoot of Semitic language family). It uses Ge'ez writing system (283 letters in our charset), none of which are in the Latin alphabet.

The speech corpora for Lithuanian and Amharic consist of spontaneous telephone conversations, each with about 40 hours of manually transcribed training data. Results are reported on the 10 hour development data set. The official lists of development keywords (words or short phrases) provided by NIST were used for KWS. The Lithuanian keyword list has 4079 keywords (412 contain at least one OOV term and considered OOV), and the Amharic list consists of 2348 keywords (368 OOV).

http://dx.doi.org/10.21437/Interspeech.2016-1200

Table 1: Text resources (excluding audio transcripts). For the parallel texts, English is the source language, and Lithuanian/Amharic are the target languages.

Data	Source		Target	
	tokens	vocab	tokens	vocab
Eng-Lit Parallel	109.5M	396k	83.8M	887k
Eng-Amh Parallel	2.2M	27k	1.4M	46k
Eng CTS	37.3M	90k	–	–
Lit Web	–	–	46.1M	2696k
Amh Web	–	–	73.0M	3484k

2.2. Text resources

The OPUS [16] parallel corpus served to build the MT systems. The English-Lithuanian data (1^{st} entry in Table 1) include OpenSubtitles, Europarl, legal documents and books. The English-Amharic data (2^{nd} entry) are composed of a Quran translation from OPUS corpus, a small dictionary and a few newspapers from the Ge'ez Frontier Foundation[1]. The parallel resources for Amharic are about $1/50^{th}$ the volume of those available for Lithuanian, and contain almost no conversational texts. The English CTS transcriptions (3^{rd} entry) are from the LDC Fisher, Switchboard and Callhome corpora [17, 18, 19]. The last two entries summarize the statistics of the raw Web texts for both languages provided by BBN [3].

3. Keyword spotting system

For each segment, the automatic speech recognizer (ASR) [20] generates word lattices using a trigram LM. Each STT hypothesis is generated with consensus decoding [21]. For KWS, search is carried out on the consensus network ignoring word boundaries [22]. Keyword-specific thresholding (KST) is applied for score normalization [23].

Back-off trigram LMs are trained using the transcripts of the 40 hour training data set. Lithuanian pronunciation lexicon is based on graphemes, as it was shown in [24, 25] that using phonemes results in a similar performance for this language. For Amharic phoneme based lexicon is derived from the grapheme-to-phoneme mappings provided by Appen. In total there are 33 units for Lithuanian and 30 for Amharic, plus 3 non-speech units (silence and 2 fillers).

The acoustic models (AMs) are triphone-based and word position-dependent. The Lithuanian AM (see details in [24]) is based on hidden Markov model with Gaussian mixture observation densities and multilingual stacked bottleneck features provided by BUT [26]. The Amharic AM is a combination of state posterior probabilities computed with 2 deep neural networks (DNNs) [27] trained with state-level minimum Bayes risk (sMBR) criterion [28]. Each DNN has 4 hidden layers with about 9M parameters. The input bottleneck features are extracted from 3M parameter DNNs trained using PLP and TRAP acoustic features.

Performance of the baseline Lithuanian and Amharic systems in terms of word error rate (WER) and maximum term-weighted value (MTWV) are given in the lines A of Table 2. MTWV is a commonly used measure for evaluating KWS systems [29]. The higher MTWV score means better performance. For the text augmentation experiments, MTWV is measured on 3 types of keywords: in-vocabulary (INV) words that appear in the initial word list (INV-INV), OOV words that become INV due to vocabulary extension (OOV-INV), and still remaining OOV words (OOV-OOV).

[1] https://github.com/geezorg/data.git

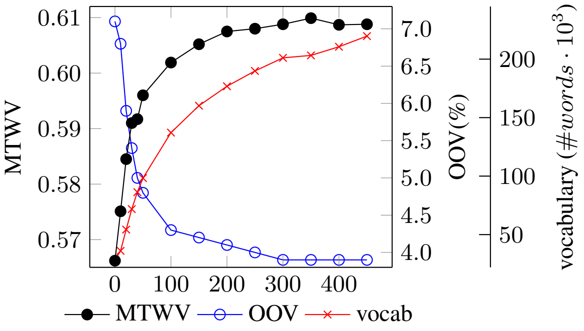

Figure 1: MTWV, OOV rate and vocabulary of Lithuanian system as functions of quantity (millions of tokens) of selected 20-best MT data.

4. MT based data augmentation

Given a large corpus of English CTS data, the goal is to produce conversational texts in Lithuanian and Amharic for ASR LM training[2].

The parallel data are aligned with the FastAlign toolkit [30] and used to train the MT phrase table. The target language data are also used to train a 4-gram LM for the MT decoder. The MT system parameters are optimized using minimum error rate training (MERT) [31] on a held-out set of 3k parallel sentences.

To assess the improvements coming from the use of the translated data, a contrastive experiment is done by adding target language texts (from the MT training data) in the ASR LM (entries B of Table 2). Compared to the baseline systems, the OOV rate is reduced by about 35% for Lithuanian and 9% for Amharic, resulting in 3.1 and 0.8 improvements in MTWV, respectively. The improvements for Lithuanian are larger due to richer and more conversational style MT training texts (see Table 1). Adding the translated data in the ASR LM (Table 2, lines C) gives no significant improvement over simply adding the MT training data.

To get more information from translated texts, the 20-best translations were extracted and filtered based on cross-entropy (CE) using the XenC toolkit [32]. XenC trains two 4-gram LMs on the Babel training transcripts and the translated texts. For each line of MT text, a score is assigned based on the difference in cross-entropy computed with the two language models. The transcripts are ranked and the best sentences are selected. Figure 1 shows the MTWV, the OOV rate and the vocabulary size as functions of the quantity of selected texts. The MTWV is highly correlated with the OOV rate.

Results with the best selections (350M tokens for Lithuanian and 300M for Amharic) are shown in lines D of Table 2. For Lithuanian, the WER is 1% lower than with the 1-best MT, but only a small improvement is seen for Amharic. We noticed that some words from English data (mostly proper names) were appearing in selected Lithuanian translations. This is not possible for Amharic, since all English words are out-of-charset.

5. Using Web data in STT, KWS and MT

Several ways of using Web data provided by our partner BBN were compared. Two forms were available, the raw texts and texts filtered with a unigram document frequency (DF) based approach [33]. The first two lines in Table 3 compare adding the raw and DF filtered texts in the ASR LM. Different vocabulary

[2] Here and later the baseline LM is interpolated with the LM trained on the additional data, and the weights tuned on the development set

Table 2: Results with various texts in ASR language model: transcriptions of 40 hour train set (A); and when adding: target language data from parallel corpus (B); 1-best translations of English CTS (C); and 20-best translations filtered with cross-entropy (D).

ID	Lang	LM			OOV	WER	MTWV			
		transcripts	tokens	vocab	%	%	All	INV-INV	OOV-INV	OOV-OOV
A	Lit	trn (baseline)	284.0k	28k	7.1	43.0	0.566	0.619	—	0.102
B		+ Parallel Lit	59.4M	697k	4.5	42.3	0.597	0.625	0.603	0.167
C		+ MT 1-best	23.5M	185k	5.2	42.5	0.590	0.621	0.630	0.185
D		+ MT 20-best (CE filt)	350.0M	211k	3.9	41.5	0.610	0.629	0.661	0.210
A	Amh	trn (baseline)	248.5k	33k	10.3	44.9	0.499	0.583	–	0.035
B		+ Parallel Amh	147.1k	64k	9.4	44.9	0.507	0.579	0.748	0.048
C		+ MT 1-best	23.0M	61k	9.4	44.9	0.509	0.579	0.782	0.057
D		+ MT 20-best (CE filt)	300.0M	63k	9.4	44.8	0.511	0.581	0.784	0.053

Table 3: Results using training transcripts and Web data for ASR language modeling: raw Web data (A); filtered Web data with document frequency (B), and cross-entropy approaches (C); adding filtered Web texts in MT system to enrich translation LM (D); and adding pseudo parallel data in translation phrase table (E).

ID	Lang	LM			OOV	WER	MTWV			
		transcripts	tokens	vocab	%	%	All	INV-INV	OOV-INV	OOV-OOV
A	Lit	Web: Raw	48.0M	1504k	1.4	42.0	0.608	0.612	0.617	0.379
B		Web: DF filtered	15.4M	615k	1.8	40.3	0.631	0.637	0.653	0.372
C		Web: CE filtered	25.0M	803k	1.6	39.8	0.632	0.639	0.660	0.289
D		MT 20-best with Web in LM	350.0M	194k	4.0	41.5	0.608	0.627	0.662	0.195
E		MT 20-best with Web in LM & PT	300.0M	188k	3.9	41.2	0.613	0.632	0.670	0.213
A	Amh	Web: Raw	73.0M	2054k	2.7	43.5	0.556	0.583	0.548	0.107
B		Web: DF filtered	14.3M	618k	4.0	43.4	0.550	0.585	0.571	0.084
C		Web: CE filtered	20.0M	1065k	3.2	43.3	0.556	0.586	0.556	0.109
D		MT 20-best with Web in LM	250.0M	60k	9.5	44.8	0.513	0.583	0.781	0.058
E		MT 20-best with Web in LM & PT	50.0M	54k	9.5	44.8	0.512	0.582	0.785	0.060

sizes were tested based on selection with unigram probabilities in the DF filtered texts. It was found that using the full vocabulary gave the best results. For Lithuanian, a 100k word list provided by BBN performed less well in terms of WER (1%) and MTWV (1.5%) compared to the full vocabulary. As in Section 4, CE filtering was also investigated. The best selections are given in lines C of Table 3. For Lithuanian this leads to a 0.5% absolute WER reduction compared to DF filtering, but only a tiny improvement is observed for Amharic. For Lithuanian using the filtered texts is better than the raw one, which is not as important for Amharic. We attribute this to the fact that Amharic texts are well normalized with the simple out-of-charset approach applied in all our experiments.

We also investigated using the filtered Web data to improve the MT systems, thereby producing better translations. Moses target LMs are built by interpolating 4-gram LMs estimated from parallel target language, Web and ASR training data. The results shown in lines D of Table 3 are achieved with the filtered 20-best translations. There is no improvement for Lithuanian, and only a small gain in MTWV is observed for Amharic (compare to Table 2 line D, without Web data). Our interpretation is that the 20-best hypotheses already extract most of the relevant information from phrase tables. Comparing the results across columns INV-INV and OOV-INV we conclude that MT data augmentation improves both vocabulary and language model for Lithuanian (INV-INV and OOV-INV), but most of the gain for Amharic comes from adding new words (only OOV-INV).

In addition, the Web data were used to augment the phrase table with pseudo-parallel data inspired by [34]. For that, an auxiliary MT system is trained in the reverse direction using the same parallel data and the English CTS data in MT LM. Then, Web texts are translated to English and a new MT system is produced using these pseudo-parallel data. This system is then used to translate the English CTS data and the resulting 20-best filtered transcriptions are added in ASR LM. The results are given Table 3 lines E. For Lithuanian, this semi-supervised approach results in an additional WER reduction of 0.3% and 0.3% improvement in MTWV. For Amharic, the performance is comparable to that obtained by just adding Web data in MT LM. In both cases, the best result is obtained with a smaller number of tokens and a smaller vocabulary compared to 20-best MT transcripts without semi-supervised training (Table 3 lines D and Table 2 lines D).

6. Character LSTM for text generation

Extending text resources without the need for large additional text corpora (especially parallel data required for MT training) is attractive, as it might be difficult to find such data for some languages. An alternative to the MT method is generating texts using a recurrent neural network. Inspired by [14, 15], a character long short-term memory (LSTM) RNN with 2 hidden layers and 512 neurons per layer was trained using only the Babel training transcripts (reserving 5% for validation). This LSTM was used to produce about 33M text tokens for Lithuanian and 97M for Amharic, with respective vocabulary sizes of 2.9M and 7.2M words. Although the network is trained on character sequences, it produces quite meaningful sentences with some new words in correct grammatical forms. However, many of the infrequent words in the new texts do not exist in the languages. Thus, generated texts were filtered using the CE approach described in Section 4 and used for ASR LM training (Table 4, lines A). Selecting about a half of the LSTM generated texts (8k and 6k vocabulary for Lithuanian and Amharic) reduces the OOV rate by about 25%, which results in roughly 2% improvement in MTWV with respect to the baseline system (Table 2, lines A). For Amharic, the KWS performance improvement is even larger than the one achieved with the best MT approach.

Next, additional Web texts are used in two manners to im-

Table 4: Results of training language models with texts generated by character LSTM. Comparing LSTM training using ASR train transcripts (A,B) and Web data (C,D). Filtering generated texts with cross-entropy (A,C) and with Web data vocabulary (B,D).

ID	Lang	LSTM	Text	Text size		OOV	WER	MTWV			
		training data	filtering	tokens	vocab	%	%	All	INV-INV	OOV-INV	OOV-OOV
A	Lit	trn	CE	16.8M	811k	5.2	42.8	0.587	0.612	0.605	0.296
B		trn	Web vocabulary	32.8M	142k	4.7	42.6	0.599	0.621	0.612	0.305
C		trn+Web	CE	22.8M	803k	2.5	40.8	0.621	0.632	0.654	0.260
D		trn+Web	Web vocabulary	78.9M	479k	2.1	40.6	0.623	0.632	0.650	0.327
A	Amh	trn	CE	19.0M	612k	7.8	45.2	0.514	0.574	0.491	0.094
B		trn	Web vocabulary	97.0M	266k	5.9	44.6	0.534	0.578	0.528	0.155
C		trn+Web	CE	60.8M	1437k	3.2	44.3	0.555	0.588	0.488	0.103
D		trn+Web	Web vocabulary	105.1M	657k	3.8	43.6	0.549	0.583	0.554	0.100

Table 5: Interpolating language models vs combining system outputs.

ID	Lang	Combine	Models	Text size		OOV	WER	MTWV			
		level		tokens	vocab	%	%	All	INV-INV	OOV-INV	OOV-OOV
A	Lit	LM	Web + MT	325.0M	840k	1.5	39.8	0.633	0.639	0.653	0.314
B		LM	Web + LSTM	103.9M	877k	1.5	39.8	0.632	0.638	0.654	0.340
C		LM	Web + LSTM + MT	403.9M	912k	1.5	39.8	0.633	0.639	0.650	0.315
D		output	Web + MT	–	–	–	39.7	0.634	0.640	0.668	0.324
E		output	Web + LSTM	–	–	–	39.6	0.633	0.638	0.667	0.333
F		output	Web + LSTM + MT	–	–	–	39.6	0.635	0.640	0.669	0.340
A	Amh	LM	Web + MT	70.0M	1070k	3.2	43.4	0.555	0.585	0.554	0.109
B		LM	Web + LSTM	125.1M	1136k	3.0	43.3	0.556	0.586	0.542	0.110
C		LM	Web + LSTM + MT	175.1M	1141k	3.0	43.2	0.559	0.559	0.551	0.105
D		output	Web + MT	–	–	–	43.1	0.559	0.587	0.558	0.110
E		output	Web + LSTM	–	–	–	43.1	0.561	0.587	0.587	0.137
F		output	Web + LSTM + MT	–	–	–	43.1	0.562	0.586	0.574	0.141

prove LSTM generated data. First, the words in the generated texts are filtered using raw Web vocabulary by mapping unobserved words to <UNK> (Table 4, lines B). About 5% of LSTM generated words are found in the raw Web texts, but the vocabulary is increased by a factor of 5 compared to that of the training transcripts. This filtering results in a 1-2% MTWV improvement for both languages. Second, a 3 hidden layer LSTM with 650 units per layer is trained using the training transcripts and the CE filtered Web texts. 79M tokens (3.1M vocabulary words) are generated for Lithuanian and 105M (5.1M vocabulary words) for Amharic. The LSTMs trained on larger data sets produce fewer new words since the starting vocabulary is larger. The generated data are again filtered with CE approach using Babel training LM (Table 4 lines C), or using the vocabulary shared with the unfiltered Web data (Table 4 lines D). Compared to the LSTM trained only on the Babel transcripts (rows A and B), the WER is improved by 1-2% absolute and the average MTWV improved by 2%.

7. Combining approaches

Table 5 gives the results obtained by interpolating LMs (lines A-C) trained on Web data, translated texts and LSTM generated data, and by combining outputs (lines D-F). ROVER [35] is used for STT combination, and KWS hits are combined by picking word with maximal normalized score. *Web* denotes CE filtered Web texts (Table 3, lines C), *MT* denotes 20-best translations using MT system trained with pseudo-parallel data (Table 3, lines E), and *LSTM* denotes texts generated with LSTM trained using Web data and filtered using Web vocabulary (Table 4, lines D). Interpolating language models for Lithuanian and Amharic results in only 0.1% and 0.3% MTWV improvement. Combining outputs of three systems is costly, but yields a 0.3-0.6% improvement in MTWV, and about 7% over the LM built using the 40-hour training transcripts.

8. Conclusions

This research reported on applying MT-based data augmentation for the Lithuanian and Amharic language modeling, without and with additional Web text resources. Exploiting 20-best filtered translations of English CTS data outperforms 1-best translations. When adding translated texts in ASR LM, the STT and KWS performance is correlated with the quantity of MT training data. The best English-Lithuanian MT system improves the Lithuanian MTWV by 4.7% and WER by 1.8% absolute. Smaller improvements were obtained for Amharic likely due to the lack of MT training data.

Adding Web texts in ASR LM was shown to be the most efficient way to improve the performance of both STT and KWS. By filtering more conversational style data and significantly expanding the vocabulary size, both WER and MTWV are improved. As some languages may have little or no text resources on the Web, a character LSTM was explored for the text generation. This approach led to a reduction in OOV and improvement in KWS performance.

9. Acknowledgements

We would like to thank BBN for sharing Web data and suggesting tools, and BUT for the bottleneck features.

This research was supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Defense US Army Research Laboratory contract number W911NF-12-C-0013. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DoD/ARL, or the U.S. Government.

10. References

[1] G. Mendels, E. Cooper, V. Soto, J. Hirschberg, M. Gales, K. Knill, A. Ragni, and H. Wang, "Improving speech recognition and keyword search for low resource languages using Web data," in *Proc. of INTERSPEECH*, 2015.

[2] A. Gandhe, L. Qin, F. Metze, A. Rudnicky, I. Lane, and M. Eck, "Using Web text to improve keyword spotting in speech," in *Proc. of ASRU*, 2013, pp. 428–433.

[3] L. Zhang, D. Karakos, W. Hartmann, R. Hsiao, R. Schwartz, and S. Tsakalidis, "Enhancing low resource keyword spotting with automatically retrieved web documents," in *Proc. of INTERSPEECH*, 2015.

[4] M. Harper, "IARPA Babel program," http://www.iarpa.gov/-index.php/research-programs/babel.

[5] S. Khudanpur and W. Kim, "Using cross-language cues for story-specific language modeling," in *Proc. of INTERSPEECH*, 2002.

[6] W. Kim and S. Khudanpur, "Lexical triggers and latent semantic analysis for cross-lingual language model adaptation," *ACM Transactions on Asian Language Information Processing (TALIP)*, vol. 3, no. 2, pp. 94–112, 2004.

[7] A. Jensson, K. Iwano, and S. Furui, "Development of a speech recognition system for Icelandic using machine translated text," in *Proc. of SLTU*, 2008, pp. 18–21.

[8] H. Cucu, A. Buzo, L. Besacier, and C. Burileanu, "SMT-based ASR domain adaptation methods for under-resourced languages: Application to Romanian," *Speech Communication*, vol. 56, pp. 195–212, 2014.

[9] G. Huang, A. Gorin, J.-L. Gauvain, and L. Lamel, "Machine translation based data augmentation for Cantonese keyword spotting," in *Proc. of ICASSP*, 2016.

[10] T. Mikolov and G. Zweig, "Context dependent recurrent neural network language model," in *Proc. of SLT*, 2012, pp. 234–239.

[11] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens *et al.*, "Moses: Open source toolkit for statistical machine translation," in *ACL'07*. Association for Computational Linguistics, 2007, pp. 177–180.

[12] I. Sutskever, J. Martens, and G. E. Hinton, "Generating text with recurrent neural networks," in *Proc. of ICML*, 2011, pp. 1017–1024.

[13] A. Karpathy, "The unreasonable effectiveness of recurrent neural networks," 2015. [Online]. Available: http://karpathy.github.io/2015/05/21/rnn-effectiveness

[14] A. L. Maas, Z. Xie, D. Jurafsky, and A. Y. Ng, "Lexicon-free conversational speech recognition with neural networks," in *Proc. of NAACL HLT*, 2015.

[15] H. Kyuyeon and S. Wonyong, "Character-level incremental speech recognition with recurrent neural networks," in *Proc. of ICASSP*, 2016.

[16] J. Tiedemann, "Parallel data, tools and interfaces in OPUS," in *Proc. of LREC*, 2012.

[17] C. Cieri, D. Graff, O. Kimball, D. Miller, and K. Walker, "Fisher english training speech part 1 transcripts LDC2004T19," *Web Download*, 2004.

[18] J. J. Godfrey and E. Holliman, "Switchboard-1 release 2," *Linguistic Data Consortium, Philadelphia*, 1997.

[19] A. Canavan, D. Graff, and G. Zipperlen, "Callhome american english speech," *Linguistic Data Consortium*, 1997.

[20] J.-L. Gauvain, L. Lamel, and G. Adda, "The LIMSI broadcast news transcription system," *Speech communication*, vol. 37, no. 1, pp. 89–108, 2002.

[21] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus in speech recognition: word error minimization and other applications of confusion networks," *Computer Speech & Language*, vol. 14, no. 4, pp. 373–400, 2000.

[22] W. Hartmann, V.-B. Le, A. Messaoudi, L. Lamel, and J.-L. Gauvain, "Comparing decoding strategies for subword-based keyword spotting in low-resourced languages," in *Proc. of INTERSPEECH*, 2014, pp. 2764–2768.

[23] D. Karakos, R. Schwartz, S. Tsakalidis, L. Zhang, S. Ranjan, T. Tim Ng, R. Hsiao, G. Saikumar, I. Bulyko, L. Nguyen, J. Makhoul, F. Grezl, M. Hannemann, M. Karafiat, I. Szoke, K. Vesely, L. Lamel, and V.-B. Le, "Score normalization and system combination for improved keyword spotting," in *Proc. of ASRU*, 2013, pp. 210–215.

[24] R. Lileikytė, L. Lamel, and J.-L. Gauvain, "Conversational telephone speech recognition for Lithuanian," in *Proc. of SLSP*, 2015, pp. 164–172.

[25] R. Lileikytė, A. Gorin, L. Lamel, J.-L. Gauvain, and T. Fraga-Silva, "Lithuanian broadcast speech transcription using semi-supervised acoustic model training," in *Proc. of SLTU*, 2016, pp. 107–113.

[26] F. Grézl and M. Karafiát, "Combination of multilingual and semi-supervised training for under-resourced languages," in *Proc. of INTERSPEECH*, 2014.

[27] X. Zhang, J. Trmal, D. Povey, and S. Khudanpur, "Improving deep neural network acoustic models using generalized maxout networks," in *Proc. of ICASSP*, 2014, pp. 215–219.

[28] K. Veselỳ, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks," in *Proc. of INTERSPEECH*, 2013, pp. 2345–2349.

[29] J. G. Fiscus, J. Ajot, J. S. Garofolo, and G. Doddingtion, "Results of the 2006 spoken term detection evaluation," in *Proc. of SIGIR*, vol. 7, 2007, pp. 51–57.

[30] C. Dyer, V. Chahuneau, and N. A. Smith, "A simple, fast, and effective reparameterization of IBM model 2." Association for Computational Linguistics, 2013.

[31] F. J. Och, "Minimum error rate training in statistical machine translation," in *Proc. of ACL*. Association for Computational Linguistics, 2003, pp. 160–167.

[32] A. Rousseau, "Xenc: An open-source tool for data selection in natural language processing," *The Prague Bulletin of Mathematical Linguistics*, no. 100, pp. 73–82, 2013.

[33] L. Zhang, D. Karakos, W. Hartmann, R. Hsiao, R. Schwartz, and S. Tsakalidis, "Enhancing low resource keyword spotting with automatically retrieved Web documents," in *Proc. of INTERSPEECH*, 2015.

[34] N. Ueffing, G. Haffari, and A. Sarkar, "Semi-supervised learning for machine translation," 2009.

[35] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER)," in *Proc. of ASRU*, 1997, pp. 347–354.