A General Procedure for Improving Language Models in Low-Resource Speech Recognition

Qian Liu*, Wei-Qiang Zhang*, Jia Liu*, Yao Liu†
* *Beijing National Research Center for Information Science and Technology*
Department of Electronic Engineering, Tsinghua University
Beijing 100084, China
Email: q-liu18@mails.tsinghua.edu.cn, {wqzhang, liuj}@tsinghua.edu.cn
† *China General Technology Research Institute*
Beijing 100084, China
Email: liuyao88@mail.ustc.edu.cn

***Abstract*—It is difficult for a language model (LM) to perform well with limited in-domain transcripts in low-resource speech recognition. In this paper, we mainly summarize and extend some effective methods to make the most of the out-of-domain data to improve LMs. These methods include data selection, vocabulary expansion, lexicon augmentation, multi-model fusion and so on. The methods are integrated into a systematic procedure, which proves to be effective for improving both n-gram and neural network LMs. Additionally, pre-trained word vectors using out-of-domain data are utilized to improve the performance of RNN/LSTM LMs for rescoring first-pass decoding results. Experiments on five Asian languages from Babel Build Packs show that, after improving LMs, 5.4-7.6% relative reduction of word error rate (WER) is generally achieved compared to the baseline ASR systems. For some languages, we achieve lower WER than newly published results on the same data sets.**

***Keywords*-language modeling; speech recognition; low-resource languages; data augmentation;**

I. Introduction

Limited speech and transcripts often lead to poor performance of automatic speech recognition (ASR) systems [1]. In the past decade, special attention has been paid to ASR in this low-resource condition. IARPA BABEL program[1] is aimed to improve the performance of ASR and keyword search (KWS) with limited transcribed speech. Additionally, Open Keyword Search (OpenKWS) and Open Speech Analytic Technologies (OpenSAT) evaluation series[2] encourage researchers to explore novel methods for ASR in low-resource condition.

Language model (LM) is one of the key components of ASR systems, and the performance of LM is crucial for ASR systems. Researchers have made great efforts to improve LMs in low-resource speech recognition. The most commonly used and the simplest method is to acquire more training text data (out-of-domain data) from other resources [2]. With the explosive growth of data on the Internet, a large amount of textual data is available, even for most of the minority languages [3]. Besides, texts generated by machine translation have also been experimented for low-resource language modeling [4]. Nowadays. translation softwares have been able to support translation of many low-resource languages, with which we can easily translate in-domain texts from a common language, such as English, into the target language, such as Georgian. The translated texts prove to be helpful for data augmentation. However, the improvement is not so notable in previous work [4]. In addition, LMs based on sub-word units or even characters are proposed [5][6]. In this way, the training data can be considered to be more adequate and out-of-vocabulary (OOV) rate is reduced. However, these methods are not suitable for word-based ASR systems.

After an investigation into previous work, we find that current methods of utilizing out-of-domain data are still unsystematic and not thoroughly summarized. There is still no general procedure to take the most advantage of out-of-domain data to improve LMs. In this paper, we summarize and extend some effective methods, which prove to be generally effective for improving LMs across different languages and ASR systems.

The rest of the paper is organized as follows. In the following section, we introduce our general procedure for improving the word-level LMs, and explain how each step is performed in detail. The whole procedure is shown in Fig. 1. In Section III, we take Georgian as an example to improve the LMs step by step, along with detailed analyses and evaluation. Experiments on other Asian languages in BABEL program are demonstrated in Section IV. Finally, we conclude our work in Section V.

II. Methods

A. Out-of-Domain Data Acquisition

For data acquisition, the most efficient method is to get more texts from the websites in the language that we need. After properly processed, high-quality texts can be retrieved [2][3]. Translated text can also be useful, but the text quality may be unsatisfactory because of the limit of translation performance [7]. As for text generated by LMs, they are not so effective in low-resource condition, because sentences with grammatical errors are likely to be generated [8]. Based on the previous work and our validation, we highly recommend web texts as the first data source and translated text as an auxiliary for data augmentation.

[1] https://www.iarpa.gov/index.php/research-programs/babel
[2] https://www.nist.gov/itl/iad/mig/opensat

978-1-7281-5014-7/19/$31.00 ©2019 IEEE

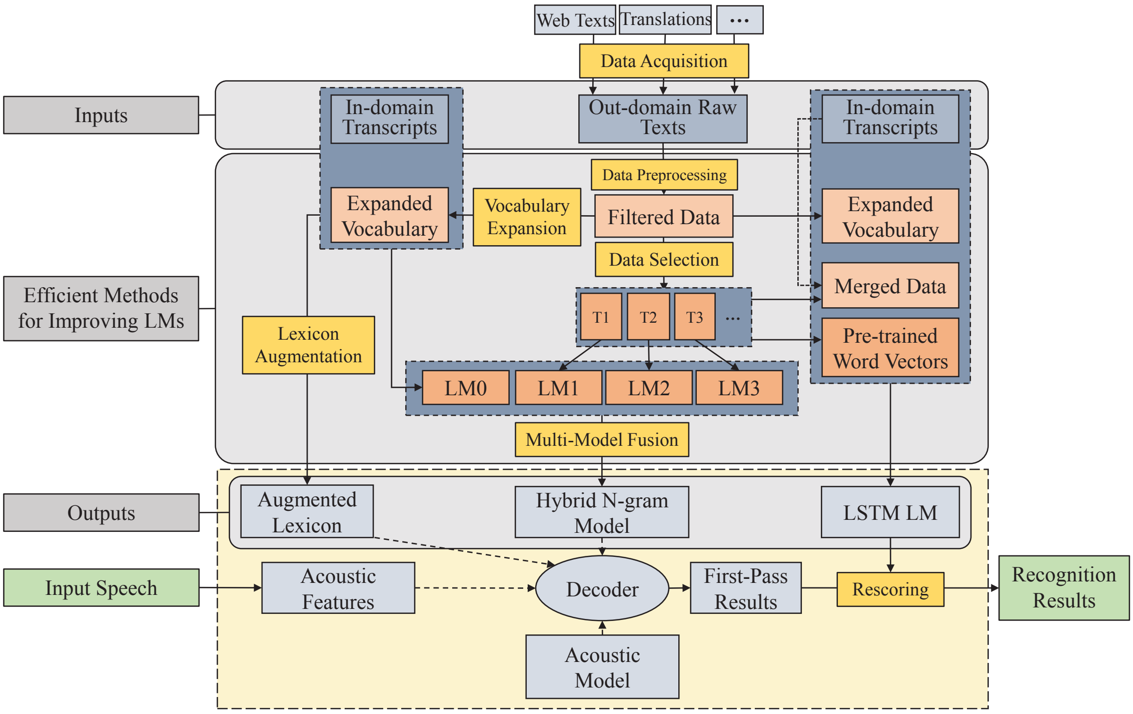

Figure 1. A general procedure for improving LMs in low-resource ASR. **Inputs:** in-domain transcripts and out-of-domain raw text. **Outputs:** an augmented lexicon, a hybrid n-gram LM and an improved LSTM LM. **T1-T3:** text corpus selected by different methods.

B. Data Preprocessing and Selection

The raw text from the Internet cannot be directly used as training materials, because some may be useless and will worsen the LMs in our experiments. What we can do is to select texts based on the similarity to the in-domain transcripts and get rid of irrelevant texts [2][9]. In our work, useless characters and symbols are directly deleted. As for punctuation marks, they can also be removed after sentence segmentation [9]. We regard data selection as a task of text similarity analysis. More methods than previous work [10] for data selection are explored in Section II-B, and three methods achieve satisfactory results and are chosen for improving LMs.

C. Vocabulary Expansion

In low-resource condition, words from in-domain transcripts are not enough to build a powerful vocabulary. Due to the limited size of vocabulary, the OOV rate on development/test data is high and leads to poor performance of LMs. According to previous work, when the vocabulary size is larger, the LM usually performs better because of lower OOV rate. However, if the vocabulary size is too large, LM can be unnecessarily complex and consume more time while decoding [11]. Therefore, efforts should be made to expand the vocabulary appropriately.

The most common method is to increase vocabulary size [9] based on word frequency. In our work, we select a certain number of words with higher frequency in out-of-domain data and merge with original vocabulary. The specific vocabulary size is determined by some primary experiments on development data, which will be demonstrated in Section III.

D. Lexicon Augmentation

After increasing the vocabulary size, the pronunciation lexicon should be modified accordingly. It is vital that the phonemes in the lexicon should be the same as the original lexicon, because the acoustic models are trained based on these phonemes and the acoustic models should remain invariant since our work focuses on improving LMs. For alphabetic languages like Georgian, a grapheme-to-phoneme (G2P) lexicon modeling method is commonly used to generate lexicon for new words [12]. The graphemes are mapped into a set of phonemes as the basic units for acoustic modeling. We include this grapheme-based method in our procedure to generate pronunciation lexicon for newly added words.

E. Multi-Model Fusion

The selected text corpus in Section II-B can be further used to improve LMs. For neural network LMs, we can simply combine the selected texts with original transcripts as training data [9]. However, for n-gram LMs, we notice that little gains are realized with the same method, especially when the size of selected texts are much more than in-domain transcripts. Therefore, we investigate a multi-model fusion method for n-gram LMs [2] based on language model interpolation. It proves that combining more models with different training corpora can help achieve significant improvements.

In Section II-B, we select a few groups of text using three methods. Then n-gram LMs can be trained separately instead of training a single LM. These models are also trained with the augmented vocabulary in Section II-C. We interpolate the auxiliary models linearly with the original LM into a hybrid n-gram LM. The interpolation coefficients can be determined by minimizing the perplexity (PPL) on development data [9].

F. Word Vector Pre-Training

Neural Network Language Models (NNLMs) have been widely used for language modeling [13][14][15]. However, when NNLMs are used for low-resource language modeling, the improvement is limited because the parameters are poorly estimated for lack of training data.

Traditionally, word vectors in NNLMs are trainable parameters and are trained with other parameters together. We notice that a few word-embedding methods are used to pre-train the word vectors in NLP tasks. Skip-Gram and CBOW (Continuous Bags of Words) are two widely used methods [16]. We experiment on these two methods to pre-train word vectors in RNN/LSTM LMs, and two kinds of usage of the pre-trained vectors are proposed for different considerations. The first method is to take the pre-trained word-vectors as the final representation and set them as non-trainable parameters. Because if the vocabulary size is extremely large, the number of parameters in the neural network for word vectors will be huge. The other method is to initialize NNLMs with pre-trained word vectors instead of random initialization. The experiment are done on RNN LM and LSTM LMs, and results show that the RNN/LSTM LMs achieve slightly better performance when word vectors are used for parameter initialization.

G. Rescoring

In our work, the hybrid n-gram LM after model fusion in Section II-E are used for first-pass decoding. Then we use the improved RNN/LSTM LMs to rescore the first-pass lattices as shown in [17]. The experiments show that rescoring with LSTM LM can slightly further improve the speech recognition performance.

III. Experiments on Georgian

In this section, we take Georgian as an example to improve the LMs systematically according to the procedure above. We do a series of experiments to demonstrate how each step is performed and evaluate the improvement of each step. For LMs with the same vocabulary size, perplexity (PPL) and Word Error Rate (WER) are both used to evaluate the performance. As for LMs with different vocabulary size, only WER is a reasonable metric [18]. More experiments are done on other low-resource Asian languages in Section IV to test the generality of our procedure.

A. Data

The in-domain Georgian speech and transcripts are from OpenKWS 2016 Surprise Language build pack, including 80-hour training speech in which only 40-hour speech is transcribed, and 10-hour test speech with transcripts. Out-of-domain texts are collected and preliminarily filtered by BBN WebText Collection System [19]. Details about the in-domain and out-of-domain texts is shown in Table I.

Table I
Details of data for Georgian language modeling. **Time:** time length of speech. **Utts:** number of utterances. **Tokens:** number of tokens. **Vocab:** number of unique words. **trans:** in-domain transcripts. **web:** texts from websites.

	Source	Time	Utts	Tokens	Vocab
Training set	trans	40h	37.7k	314k	30.3k
	web	-	623k	22.8M	1.83M
Test set	trans	10h	9.2k	77.4k	-

B. Baseline ASR System

In order to evaluate the generality of our procedure for improving LMs, we have three baseline systems with different acoustic models. SGMM based acoustic models [20] are commonly used before DNN acoustic models are proposed. DNN and TDNN are two kinds of acoustic models well supported in Kaldi Toolkit [21]. As for acoustic feature, we use bottleneck feature [22] in all three baseline systems.

The pronunciation lexicon of Georgian is not provided in OpenKWS 2016 data, so we extract all unique words from the in-domain transcripts as the original vocabulary, and generate the lexicon with the help of Morfessor [23]. Since the conventional n-gram LM cannot model zero-frequency words in the training data, researchers proposed several smoothing algorithms, including Good-Turing and Kneser-Ney algorithms [24]. An maximum-entropy based n-gram LM also performs well [25]. To find the best method of language modeling, we train several n-gram LMs in a few preliminary experiments using in-domain transcripts. A 3-gram Maximum-Entropy LM (ME3) performs the best among the models and is used as the baseline LM.

In this section, we take an SGMM based ASR system as an example to demonstrate the procedure for improving LMs. Without improving LMs, the WER of SGMM baseline system is 50.7%. Experiments on systems with other acoustic models are demonstrated in Section IV.

C. Procedure of Improving LMs

For out-of-domain data mentioned in Table I, we first filter invalid symbols as mentioned in Section II-B. As for data selection, three methods are compared in the experiments.

The first method is to compute the cross-entropy (CE) difference between in-domain and out-of-domain texts as described in [26] and [10]. More specifically, two 3-gram Maximum-Entropy LMs are trained separately on in-domain and out-of-domain text using the vocabulary of in-domain data. For each sentence in the out-of-domain data, we calculate the cross-entropy using these two LMs and get the difference. The sentences with lower cross-entropy difference are assumed more similar to the in-domain transcripts. XenC [27] tool is used in this method. The other two methods split the out-of-domain texts into several documents, and transform the documents into vectors using TF-IDF and Doc2Vec [28] respectively. Text similarity is evaluated by cosine similarity based on document vectors. Finally, a certain number of sentences more similar to the in-domain transcripts are selected. It is worth noting that random selection is used as the baseline for comparison.

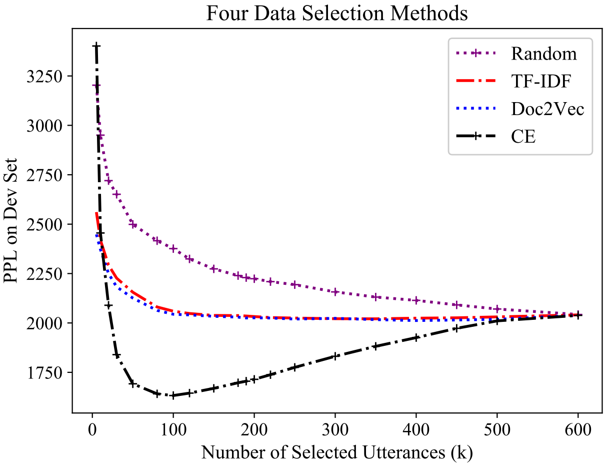

Figure 2. Performance of four data selection methods. Utterances selected from websites are used to train LMs, and evaluate the performance with PPL on development data.

Four groups of n-gram LMs are trained with the sentences selected by different selection methods. The results are shown in Fig. 2. For TF-IDF and Doc2Vec methods, about 200k-300k sentences are enough to reach the lowest PPL, and more data will not further lower the PPL on development data. As for cross-entropy based method, it performs the best and the first 100k sentences achieve the lowest PPL. It is worth noting that the LM performs worse when more data is added. It actually proves that training data should be carefully selected according to the similarity to in-domain data and is not the more the better. Finally, these selected sentences are all kept for further usage in multi-model fusion.

For vocabulary expansion, we directly select words with higher word frequency from the out-of-domain text as described in Section II-C. Meanwhile, we augment the pronunciation lexicon accordingly. Table II shows the results of different vocabulary sizes. As the vocabulary size increases, the OOV rate is significantly reduced and WER is absolutely reduced 3.0% at most. When the vocabulary size is more than 200k, the OOV rate remains around 4.0% and WER remains 47.7%, so 200k words can be assumed enough for vocabulary expansion.

We have the in-domain training transcripts and three groups of texts selected by different methods. Therefore, four different LMs can be trained separately. For language model interpolation, we test with two, three and four n-gram models respectively. The weights (interpolation coefficients) for each model are computed with SRILM Toolkit [29] to minimize the PPL on development data. The results in Table III show that the hybrid model with all four models performs the best, and WER further reduces 0.8% absolutely.

Table II
Performance of LMs with different vocabulary sizes. **Added Vocab**: the number of words from out-of-domain texts. **Merged Vocab**: final vocabulary size.

Added Vocab	Merged Vocab	OOV Rate	WER
0(Baseline)	30.3k	8.71%	50.7%
20k	40.5k	7.14%	50.0%
80k	91.0k	5.46%	48.6%
150k	155.3k	4.57%	47.9%
200k	204.0k	4.30%	**47.7%**
300k	302.0k	4.12%	47.7%
500k	500.2k	**3.98%**	47.7%

Table III
Performance of model interpolation. **trans:** in-domain transcripts. **CE/Doc2Vec/TF-IDF:** three data selection methods. **PPL:** perplexity of LMs on development data. **LM0+LM1 and so on:** a hybrid LM after linear interpolation.

LMs	Source	Utts	Tokens	PPL	WER
LM0	trans	37.7k	314k	429.30	47.7%
LM1	CE	100k	1.18M	1632.8	-
LM2	Doc2Vec	200k	6.72M	2012.2	-
LM3	TF-IDF	200k	9.38M	2021.5	-
LM0+LM1				393.11	47.1%
LM0+LM1+LM2				390.28	47.0%
LM0+LM1+LM2+LM3				**387.32**	**46.9%**

For RNN and LSTM LMs, we select texts from out-of-domain data using the same methods in Section II-B and merge with in-domain training transcripts. The vocabulary is the same as the improved n-gram LM above. Meanwhile, word vectors are pre-trained using Skip-Gram or CBOW methods. Table IV shows that lower PPL can be achieved by LMs initialized with pre-trained word vectors. Another conclusion is that Skip-Gram performs a bit better than CBOW for model initialization.

Table IV
Perplexity on development data of RNN/LSTM LMs using pre-trained word vectors. Usage of pre-trained word vectors: **A)** non-trainable parameters. **B)** model initialization.

	Methods	PPL		
		Baseline	A	B
RNN LM	CBOW	223.18	238.15	222.76
	Skip-Gram		236.22	**221.27**
LSTM LM	CBOW	217.29	230.53	217.54
	Skip-Gram		228.30	**214.70**

With hybrid n-gram LM, the WER of Georgian SGMM ASR system is reduced from 50.7% to 46.9%. Then we utilize the improved LSTM LM to do lattice rescoring, and the WER is further reduced to 46.7%. In summary, after all steps in the procedure, the WER of Georgian ASR system is 4.0% absolutely lower and **7.9%** relatively lower than the baseline system.

IV. Experiments on five BABEL Languages

In order to evaluate the general performance of the whole procedure for improving LMs, we experiment on

Table V

PERFORMANCE OF THE METHODS ACROSS DIFFERENT BABEL LANGUAGES. **TRANS**: IN-DOMAIN TRANSCRIPTS. **WEB**: OUT-OF-DOMAIN TEXTS FROM WEBSITES. **TDNN**: CHAIN TDNN ACOUSTIC MODEL. **WER$_{pub}$**: PUBLISHED RESULTS OF WER ON THE SAME DATA SET. **ARR**: AVERAGE RELATIVE REDUCTION OF WER COMPARED TO BASELINE SYSTEMS AFTER IMPROVING LMS.

Languages	Vocab Size		Tokens		WER (Baseline)			WER (after improving LMs)			WER$_{\text{pub}}$	ARR
	trans	web	trans	web	SGMM	DNN	TDNN	SGMM	DNN	TDNN		
Assamese	22.0k	200k	451.6k	1.08M	52.0	52.6	48.5	48.6	49.2	45.3	**	6.5%
Vietnamese	6.20k	300k	989.1k	4.85M	49.9	49.0	45.1	46.4	45.8	42.4	**	6.5%
Tamil	68.8k	200k	486.4k	6.79M	66.9	66.2	64.9	63.0	62.4	61.8	**	5.4%
Mongolian	20.8k	300k	402.6k	2.48M	56.7	54.5	50.4	53.3	51.1	47.2	48.7 (in [30])	6.2%
Georgian	30.3k	200k	314.1k	9.38M	50.7	50.0	44.5	46.7	46.2	41.3	42.2 (in [30])	**7.6%**

five low-resource languages in total. The in-domain training transcripts are provided by IARPA BABEL program, and the out-of-domain raw texts are collected and formatted by Leipzig Corpora Collection [31]. Additionally, we have three baseline systems with different setup of acoustic models, so we can evaluate the general performance of improved LMs across different ASR systems.

For each baseline system, we improve the LMs according to the procedure shown in Fig. 1. Table V summarizes the results of experiments on five languages. Some related state-of-the-art results on the same data set are shown in the WER$_{\text{pub}}$ column. Significant gains are achieved for each language and baseline system. We evaluate the improvements with an intuitive metric: average relative reduction (ARR) of WER for three baseline systems. ARRs for five languages range from 5.4% to 7.6%. The best performance is on Georgian with a 7.6% ARR. The results prove that the procedure for improving LMs is generally effective in low-resource speech recognition.

V. CONCLUSIONS

We investigate and refine some existing methods to improve the performance of LMs in low-resource speech recognition. Experiments show that high-quality texts can be retrieved from out-of-domain data after pre-processing and selection, which can be used for vocabulary expansion, lexicon augmentation and multi-model fusion. For RNN/LSTM LMs, word vectors pre-trained by out-of-domain texts are used for model initialization. We use the hybrid n-gram LM for the first-pass decoding, and we use improved LSTM LM for lattice rescoring. Experiments show that significant improvements are generally achieved for five languages and three different ASR systems. We believe that it can be taken as a general procedure for improving LMs for low-resource speech recognition.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China under Grant No. U1836219 and the National Key R&D Program of China.

REFERENCES

[1] A. R. Syed, A. Rosenberg, and M. Mandel, "Active learning for low-resource speech recognition: Impact of selection size and language modeling data," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5315–5319.

[2] A. Gorin, R. Lileikyte, G. Huang, L. Lamel, J.-L. Gauvain, and A. Laurent, "Language model data augmentation for keyword spotting in low-resourced training conditions." in *Interspeech*, 2016, pp. 775–779.

[3] G. Mendels, E. Cooper, V. Soto, J. Hirschberg, M. J. Gales, K. M. Knill, A. Ragni, and H. Wang, "Improving speech recognition and keyword search for low resource languages using web data," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[4] G. Huang, T. F. D. Silva, L. Lamel, J. L. Gauvain, A. Gorin, A. Laurent, R. Lileikyte, and A. Messouadi, "An investigation into language model data augmentation for low-resourced stt and kws," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5790–5794.

[5] H. Sak, M. Saraclar, and T. Gngr, "Morphology-based and sub-word language modeling for turkish speech recognition," in *International Conference on Acoustics Speech and Signal Processing (ICASSP)*. IEEE, 2010, pp. 5402–5405.

[6] Y. Kim, Y. Jernite, D. Sontag, and A. M. Rush, "Character-aware neural language models." in *AAAI*, 2016, pp. 2741–2749.

[7] G. Huang, A. Gorin, J.-L. Gauvain, and L. Lamel, "Machine translation based data augmentation for cantonese keyword spotting," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 6020–6024.

[8] I. Sutskever, J. Martens, and G. E. Hinton, "Generating text with recurrent neural networks," in *International Conference on Machine Learning (ICML)*, 2011, pp. 1017–1024.

[9] Z. Zhang, W.-Q. Zhang, K.-X. Shen, X.-K. Yang, Y. Tian, M. Cai, and J. Liu, "Thuee language modeling method for the openkws 2015 evaluation," in *International Symposium on Signal Processing and Information Technology*. IEEE, 2015, pp. 534–538.

[10] T. Fraga-Silva, A. Laurent, J.-L. Gauvain, L. Lamel, V.-B. Le, and A. Messaoudi, "Improving data selection for low-resource stt and kws," in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015, pp. 153–159.

[11] T. Alume, D. Karakos, W. Hartmann, R. Hsiao, Z. Le, N. Long, S. Tsakalidis, and R. Schwartz, "The 2016 bbn georgian telephone speech keyword spotting system," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5755–5759.

[12] M. Bisani and H. Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech Communication*, vol. 50, no. 5, pp. 434–451, 2008.

[13] A. Ragni, E. Dakin, X. Chen, M. J. F. Gales, and K. Knill, "Multi-language neural network language models," *Conference of the International Speech Communication Association*, pp. 3042–3046, 2016.

[14] T. Mikolov, M. Karafiat, L. Burget, J. Cernock, and S. Khudanpur, "Recurrent neural network based language model," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010, pp. 1045–1048.

[15] M. Sundermeyer, R. Schlter, and H. Ney, "Lstm neural networks for language modeling," in *Thirteenth Annual Conference of the International Speech Communication Association*, 2012.

[16] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[17] S. Kumar, M. Nirschl, D. Holtmann-Rice, H. Liao, A. T. Suresh, and F. Yu, "Lattice rescoring strategies for long short term memory language models in speech recognition," in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2017, pp. 165–172.

[18] S. Chen, D. Beeferman, and R. Rosenfeld, "Evaluation metrics for language models," in *Procdarpa Broadcast News Transcription & Understanding Workshop*, 1998.

[19] L. Zhang, D. Karakos, W. Hartmann, R. Hsiao, R. Schwartz, and S. Tsakalidis, "Enhancing low resource keyword spotting with automatically retrieved web documents," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[20] D. Povey, L. Burget, M. Agarwal, P. Akyazi, F. Kai, A. Ghoshal, O. Glembek, N. Goel, M. Karafit, and A. Rastrow, "The subspace gaussian mixture modela structured model for speech recognition," *Computer Speech & Language*, vol. 25, no. 2, pp. 404–439, 2011.

[21] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE Signal Processing Society, 2011.

[22] D. Yu and M. L. Seltzer, "Improved bottleneck features using pretrained deep neural networks," in *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[23] S. Virpioja, P. Smit, S.-A. Grönroos, M. Kurimo *et al.*, "Morfessor 2.0: Python implementation and extensions for morfessor baseline." Aalto University, 2013.

[24] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," *Computer Speech & Language*, vol. 13, no. 4, pp. 359–394, 1999.

[25] T. Alumae and M. Kurimo, "Efficient estimation of maximum entropy language models with n-gram features: an srilm extension," in *Eleventh Annual Conference of the International Speech Communication Association*, 2010, pp. 1820–1823.

[26] R. C. Moore and W. Lewis, "Intelligent selection of language model training data." in *Proceedings of the Meeting of the Association for Computational Linguistics*, 2010, pp. 220–224.

[27] A. Rousseau, "Xenc: An open-source tool for data selection in natural language processing," *The Prague Bulletin of Mathematical Linguistics*, vol. 100, pp. 73–82, 2013.

[28] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *International Conference on Machine Learning (ICML)*, 2014, pp. 1188–1196.

[29] A. Stolcke, "Srilm-an extensible language modeling toolkit," in *Seventh International Conference on Spoken Language Processing*, 2002.

[30] J. Trmal, M. Wiesner, V. Peddinti, X. Zhang, P. Ghahremani, Y. Wang, V. Manohar, H. Xu, D. Povey, and S. Khudanpur, "The kaldi openkws system: Improving low resource keyword search," in *Interspeech*, 2017, pp. 3597–3601.

[31] D. Goldhahn, T. Eckart, and U. Quasthoff, "Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages," in *Eighth International Conference on Language Resources and Evaluation*, 2012, pp. 759–765.