Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 2470–2474
Marseille, 11–16 May 2020

Class-based LSTM Russian Language Model with Linguistic Information

Irina Kipyatkova[1,2], Alexey Karpov[1]
[1] St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS),
[2] St. Petersburg State University of Aerospace Instrumentation (SUAI)
St. Petersburg, Russia
{kipyatkova, karpov}@iias.spb.su

Abstract

In the paper, we present class-based LSTM Russian language models (LMs) with classes generated with the use of both word frequency and linguistic information data, obtained with the help of the "VisualSynan" software from the AOT project. We have created LSTM LMs with various numbers of classes and compared them with word-based LM and class-based LM with word2vec class generation in terms of perplexity, training time, and WER. In addition, we performed a linear interpolation of LSTM language models with the baseline 3-gram language model. The LSTM language models were used for very large vocabulary continuous Russian speech recognition at an N-best list rescoring stage. We achieved significant progress in training time reduction with only slight degradation in recognition accuracy comparing to the word-based LM. In addition, our LM with classes generated using linguistic information outperformed LM with classes generated using word2vec. We achieved WER of 14.94 % at our own speech corpus of continuous Russian speech that is 15 % relative reduction with respect to the baseline 3-gram model.

Keywords: speech recognition, class-based language model, long short-term memory, Russian speech

1. Introduction

One of the main state-of-the art approach to language modeling is application of neural networks (NNs). For language modeling, the usage of recurrent NNs (RNNs) is preferable because this type of NNs can store the whole context preceding the given word in contrast to feed-forward NNs which store a context of restricted length. One type of recurrent NNs is a long short-term memory (LSTM) network which contains special units called memory blocks. Each memory block is composed of a memory cell, which stores the temporal state of the network, and multiplicative units named gates (an input gate, an output gate, and a forget gate) controlling the information flow (Hochreiter and Schmidhuber, 1997). LSTM-based language models (LMs) can be used in speech recognition systems at N-best or lattice rescoring stage. It was shown in many papers that such models outperform standard n-gram models in term of both perplexity and word error rate (WER) (Sundermeyer et al., 2015; Kumar et al., 2017). Detail review of researches on application of RNN-based LNs for speech recognition is presented in (Kipyatkova and Karpov, 2016).

There are only few researches on application of NNs for Russian language modeling. RNN LM for Russian was firstly used in (Vazhenina and Markov, 2013). RNN LM was trained on the text corpus containing 40M words with vocabulary size of about 100K words. An interpolation of the obtained model with the baseline 3-gram and factored LMs was carried out. Obtained LM was used for rescoring of 500-best list that allowed the authors to achieve WER relative improvement of 7.4%.

Another research on NN for Russian language modeling was described in (Medennikov and Bulusheva, 2016). The baseline 3-gram LM was trained on transcriptions of telephone conversations (390 hours of speech) as well as on text corpus (about 200M words) containing materials from Internet forum discussions, books, etc. Vocabulary for the baseline model contained 214K words. NN-based LMs were trained only with a part of the test corpus, and for this corpus the vocabulary of 45K most frequent words was used. LSTM-based LM was used for rescoring of 100-best list. Relative WER reduction was equal to 8%.

There are several toolkits for training RNN-based LM. One of them is RNNLN toolkit (Recurrent Neural Network Language Modeling Toolkit) (Mikolov et. al., 2011). This toolkit allows creating RNNs with one hidden layer. TF-LM (TensorFlow-based Language Modeling Toolkit) allows training LSTM and bidirectional LSTM (BLSTM) applying several optimization methods (Verwimp et. al., 2018). For our experiments, we chose TheanoLM toolkit (Enarvi and Kurimo, 2016). It supports training NNs of different types: RNN, Gated Recurrent Unit (GRU), LSTM, BLSTM, highway networks. It also provides application of several optimization methods and different stop criteria.

Training of RNN LM on a corpus with a large vocabulary is a very time-consuming task. One method for reducing the training time is to use hierarchical softmax that factors the output probabilities into the product of multiple softmax functions (Morin and Bengio, 2005). The hierarchical softmax is realized in TheanoLM toolkit, where a two-level hierarchy is used. The first level performs a softmax between $\sqrt{V}$ word classes and the second level performs a softmax between $\sqrt{V}$ words inside the correct class, where $\sqrt{V}$ is the vocabulary size (Enarvi, 2017). In this case, word probability is computed as follows:

$$P(w_t|w_0 \dots w_{t-1}) = \\ = P(\mathrm{c}(w_t)|w_0 \dots w_{t-1})P\big(w_t\big|w_0 \dots w_{t-1}, c(w_t)\big),$$

where w_t is a word at time step t, $\mathrm{c}(w_t)$ is a function that maps a word to a class.

Another technique for training speed-up is to train class-based LM instead of word-based LM. Class-based models use a function that maps every word w_i to a class c_i: f: $w_i \rightarrow f(w_i)=c_i$. At first, probability distribution over classes is computed. Then, probability distribution for the words that belong to a specific class is computed:

$$P(w_t|w_{t-n+1} \dots w_{t-1}) = P(w_t|c_t)P(c_t|c_{t-n+1} \dots c_{t-1})$$

There are several methods for word-clustering. The simplest one is to cluster the words according to their unigram frequencies (Mikolov, 2011).

Clustering can be performed basing on the contexts in which the words occur. Such method was described in (Brown et. al., 1992). In the paper, it was proposed to initially assign each word to a distinct class and to compute the average mutual information between adjacent classes, and then to merge pair of classes for which the loss in average mutual information was least.
Clustering based on Continuous Bag-of-Words (CBOW) model (Mikolov et. al., 2013) is carried out by creating word embedding vectors and clustering them using K-means.
Also for word-clustering rule-based methods can be used. For example, in (Enarvi et. al., 2017) clustering was performed with the help of a set of rules describing the usual reductions and alterations in colloquial words. In (Han et. al, 2005) rule-based word clustering was made for document metadata extraction. In the paper, the clusters were formed from various domain databases and the word orthographic properties.
In (Song et al., 2017) word-clustering was carried out basing on part-of-speech (POS) tagging. In the paper, the initial word clusters were defined by the word's tags and then the clusters larger than a predefined size were randomly broken into smaller ones in order to generate a specified number of word classes.
In our previous research (Kipyatkova, 2019), we have trained word-based LSTM LM using TheanoLM toolkit. We have obtained relative WER reduction of 22% as compared to the result obtained with our 3-gram model. Although we have applied hierarchical softmax function, training of the model takes several weeks. In the current research, we try to train class-based LSTM LM with class generation using both word's frequency and linguistic information.

2. Development of LSTM Language Model for Russian

2.1 Russian Text Corpora and Baseline LM

Our corpora of Russian texts for training and testing of both baseline and LSTM LMs are described in detail in (Kipyatkova and Karpov, 2013). The training corpus was collected from recent news published in freely available Internet sites of four on-line Russian newspapers[1]. The database contains text data that reflect contemporary Russian including some spoken language. At first, the texts were divided into sentences. Sentences containing direct and indirect speech were treated as separate sentences. These sentences can be of the following types: (1) direct speech is placed after indirect speech; (2) direct speech is before indirect speech; (3) indirect speech is within direct speech. Then, a text written in any brackets was deleted, and sentences consisting of less than six words were also removed. Uppercase letters were replaced by the lowercase ones, if a word started with an uppercase letter. If a whole word was written by the uppercase letters, then such change was made, when this word was in the dictionary of Russian words (Karpov et. al., 2013). The training corpus consists of 350M words (2.4 GB data), and it has more that 1M unique word-forms.

[1] www.ng.ru, www.smi.ru, www.lenta.ru, www.gazeta.ru

The size of the test corpus used for perplexity evaluation was 33M words. Text material for test corpus was taken from a newspaper[2] that was not used for training.
The vocabulary consists of 150K most frequent word-forms from the training corpus. As a baseline, we used word-based 3-gram model with Kneser-Ney discounting created using SRI Language Modeling Toolkit (SRILM) (Stolcke et al., 2011). The perplexity of the baseline model was 553.

2.2 Generation of Classes for Class-based LMs

Russian is a morphologically rich inflective language. Words in Russian can inflect for a number of syntactic features: case, number, gender, etc. Word-forms in Russian are derived by adding single or multiple prefixes, suffixes and endings to a stem in accordance with the grammatical category of the word (Ryazanova-Clarke and Wade, 2002). Thus, information about word's grammatical category can be used for word-clustering.
In the current research, for word-clustering we used two criteria: the frequency of word's appearance in the training text corpus and linguistic information.
We applied "VisualSynan" software from the AOT project (Sokirko, 2004) for obtaining linguistic information. As linguistic information we used a grammatical tag that codes some grammatical information about the word: part-of-speech, gender, case, singular/plural number. For example, the grammatical tag of the word "*актрисе*" ("actress") is *bc* that means noun POS, feminine gender, singular, dative case. In total, we obtained 293 different grammatical tags.
The process of word mapping consisted of two stages.
1) At the first stage, the frequencies of appearance of words in the training text were computed. Each word with frequency larger than a certain threshold was mapped to its own class. So, for each frequent word an individual class was created.
2) At the second stage, for the rest of words, grammatical tags were defined. The words having similar grammatical tag were combined in one class. Separate classes were assigned for English words (words written in Latin letters) and for abbreviations (words written by the uppercase letters).
Two lists of words created at these stages were combined. Thus, we obtained the list of words with their classes. We tried three values of word's frequency threshold: 5k, 10k, and 35k. Table 1 presents the total number of classes (including classes obtained using grammatical classes) for these values of threshold.

Threshold of word frequency	Total number of classes
35000	1265
10000	4134
5000	7801

Table 1: The obtained number of classes for several threshold of word frequency

In addition, we performed clustering words into classes with the help of word2vec[3]. For comparison, we chose the number of classes equal to 4134.

[2] www.fontanka.ru

[3] https://github.com/dav/word2vec

2.3 LSTM-based Russian Language Models

At first, we created NN LMs consisting of a projection layer, which maps words to specified dimensional embeddings, one hidden LSTM layer, and a hierarchical softmax layer. NN LM architecture is presented on Figure 1, where w_t is an input word at time t; h_t is the hidden layer state, c_t is LSTM cell state.

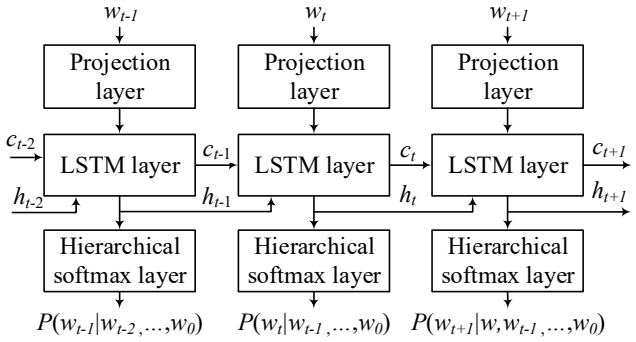

Figure 1: LSTM-based LM architecture.

We created three class-based LSTM LMs with number of classes equal to 1265, 4134, 7801 and compared them with the word-based LSTM LM described in (Kipyatkova, 2019) and class-based LSTM LM with classes obtained using word2vec. All models had the same parameters of LSTMs: the projection layer size was equal to 500, the hidden layer size was equal to 512, and Nesterov Momentum optimization method was used. These parameters gave us the best results in terms of WER in our previous experiments. The initial learning rate was equal to 1. The stopping criteria was "*no-improvement*" which means that learning rate is halved when validation set perplexity stops improving, and training is stopped when the perplexity does not improve at all with the current learning rate (Enarvi et. al., 2017). The maximum number of training epochs was 15. As well, we made a linear interpolation of the LSTM LM and the baseline LM.

The results of estimation of the created models are shown in Table 2. In the table, PPL means the value of perplexity of LSTM LMs (interpolation coefficient equals 1.0). PPL of interp. model means the value of perplexity obtained after interpolation of LSTM LM with the baseline LM. Training time of all models is presented in the table as well. Training was performed on Nvidia GeForce GTX 1080 GPU with CUDA.

As we can see from the table, the value of perplexity of LSTM LM exceeds the perplexity of 3-gram model, however the perplexity of interpolated models is lower than 3-gram model. Training of class-based LSTM was three times faster than word-based LSTM.

After that, we tried to increase the number of LSTM layers and to create a model with one LSTM layer followed by a highway network with *tanh* activation and dropout rate of 0.2 after LSTM layer similarly to network architecture described in (Enarvi et. al., 2017). Highway network was introduced in (Srivastava et. al., 2015). It uses gate units, which learn to regulate the flow of information through a network. The output (y_t) of the highway network with one gate $g(x_t)$ is defined as follows:

$$g(x_t) = \sigma(W_\sigma x_t + b_\sigma)$$

$$y_t = g(x_t) \cdot tanh(W x_t + b) + \big(1 - g(x_t)\big) \cdot x_t$$

where x_t is an input to the highway layer, W is a wight matrix, b is bias, σ is simoid activation function.

Model	Number of classes	Training Time, h	PPL	Interp. coeff.	PPL of interp. model
Word-based model					
LSTM	-	336	346	0.7	289
Class-based models					
LSTM (word2vec classes)	4134	116	1023	0.3	431
LSTM	1265	96	1209	0.4	444
LSTM	4134	100	853	0.5	407
LSTM	7801	109	717	0.5	386
LSTM+ highway	4134	106	838	0.5	404
LSTM (2 layers)	4134	128	828	0.5	399
BLSTM	4134	150	273	0.7	**163**

Table 2: Perplexities of LSTM LMs

The usage of NN with 2 LSTM layers and LSTM followed by highway network results only in slight decrease of perplexity.

Also, we created BLSTMs. Perplexity of BLSTM LM is much lower comparing to the 3-gram model and interpolation of this model with 3-gram LM allowed us to obtain additional decreasing of perplexity. In addition, we tried to increase the number of BLSTM layers but it results in overtraining.

3. Experiments

3.1 Speech Recognition Setup

The open-source Kaldi toolkit (Povey et. al., 2011) was used for training the acoustic models and performing speech decoding. We used hybrid DNN/HMMs acoustic models based on time-delay neural network with 5 hidden layers and time context [-8, 8]. Mel-frequency cepstral coefficients (MFCCs) were used as input to the NNs. For speaker adaptation, 100-dimensional i-Vector (Saon et. al., 2013) was appended to the 40-dimensional MFCC input. Detail description of our acoustic models is presented in (Kipyatkova, 2018). Transcriptions for speech recognition vocabulary were generated automatically by application of context-dependent and independent phonetic transcribing rules (almost 100 rules) to the list of word-forms with denoted stress vowel (Karpov et. al., 2011; Karpov et. al., 2013). Position of stress vowel was defined by dictionary of more than 2.3M word-forms with marks of the stressed vowels that was composed of two different morphological databases: AOT[4] and Starling[5], and expanded with some more frequent words that were absent in these databases (generally names, cites, special terms etc.).

For training the acoustic models, we used three corpora of Russian speech recorded at SPIIRAS (Kipyatkova, 2017):

[4] www.aot.ru

[5] starling.rinet.ru/morpho.php

1) the speech database developed within the framework of the EuroNounce project (Jokisch et. al., 2009) that consists of recordings of 50 speakers, each of them pronounced a set of 327 phonetically rich and meaningful phrases and texts;
2) the corpus consisting of recordings of other 55 native Russian speakers; each speaker pronounced 105 phrases: 50 phrases were taken from the Appendix G to the Russian State Standard P 50840-95 (these phrases were different for each speaker), and 55 common phrases were taken from a phonetically representative text, presented in (Stepanova, 1988);
3) the audio part of the audio-visual speech corpus HAVRUS (Verkhodanova, 2016) that consists of recordings of 20 speakers pronouncing 200 phrases: (a) 130 phrases for training were two phonetically rich texts common for all speakers, and (b) 70 phrases for testing were different for every speaker: 20 phrases were commands for the MIDAS information kiosk (Karpov, 2009) and 50 phrases were 7-digits telephone numbers (connected digits).
The total duration of the entire speech data is more than 30 hours.
To test the system, we used another speech dataset consisting of 500 phrases with the length from 6 to 20 words pronounced by 5 speakers. The phrases were taken from the materials of a Russian on-line newspaper that was not presented in the training speech and text data.
The speech data were collected in clean acoustic conditions, with 16kHz sampling rate, 16-bit audio quality. A signal-to-noise ratio (SNR) at least 35-40 dB was provided. The speech data were recorded with 44.1 KHz sampling rate (for ASR downsampled to 16 KHz), 16 bits per sample, SNR was 35dB at least, by a stereo pair of Oktava MK-012 stationary microphones (close talking ≈20 cm and far-field ≈100 cm microphone setup) connected to PC via Presonus Firepod sound board.

3.2 Experimental Results on Russian Speech Recognition

LSTM-based LM was applied for rescoring of 500-best list of hypotheses and for selection of the best recognition hypothesis for the pronounced phrase. Interpolated LMs were used for rescoring as well. Obtained speech recognition results are presented in Table 3. WER obtained with our baseline 3-gram model was equal to 17.62% (Kipyatkova, 2017). The out-of-vocabulary rate for the test set was 1.1%.
As we can see from the table, application of class-based LSTM models solely did not lead to any improvement of speech recognition results. However, after interpolation of LSTM LM with the baseline LM we have obtained reduction of WER. Application of LSTM LM with classes generated using linguistic information results in lower WER that LSTM LM with classes created with help of word2vec. Increasing the number of classes expectedly improves speech recognition results. However, if we compare LSTM models interpolated with the baseline model we can see that whereas increasing the number of classes from 1265 to 4134 decreased WER on 1%, difference in WER obtained using interpolated models with 4134 and 7801 classes was small.

Model	Number of classes	WER, %	Interp. coeff.	WER of interp. model, %
Word-based model				
LSTM	-	14.55	0.7	14.01
Class-based models				
LSTM (word2vec classes)	4134	21.24	0.3	16.82
LSTM	1265	19.31	0.3	16.67
LSTM	4134	18.33	0.4	15.71
LSTM	7801	17.38	0.5	15.51
LSTM (2 layers)	4134	18.05	0.5	15.62
LSTM+ highway	4134	17.85	0.5	15.79
BLSTM	4134	17.32	0.4	**14.94**

Table 3: WER obtained after 500-best list rescoring (%)

The usage of class-based LM results in higher WER than usage of word-based LSTM but training of class-based LSTM took much less time than word-based LSTM. Increasing the number of layers does not result in significant decreasing of WER. The lowest WER (14.94%) was obtained using BLSTM LM interpolated with 3-gram LM with interpolation coefficient equal to 0.4.

4. Conclusions

In the paper, we have investigated class-based LSTM LMs for Russian speech recognition. We have implemented class generation using both word frequency and linguistic information. Application of class-based LSTM LM interpolated with 3-gram LM results in significant improvement in training time comparing to the word-based LM. We have also performed experiments with LSTM followed by highway network and BLSTM. Finally, we have achieved 15% relative reduction of WER using BLSTM LM with respect to the baseline 3-gram model.

5. Acknowledgements

This research is financially supported by the Russian Foundation for Basic Research, projects 19-29-09081 (Section 2), No. 18-07-01216, and No. 18-07-01407.

6. Bibliographical References

Brown, P. F., Desouza, P. V., Mercer, R. L., Pietra, V. J. D., and Lai, J. C. (1992). Class-based n-gram models of natural language. *Computational Linguistics*. 18(4): 467–479.

Enarvi, S. and Kurimo, M. (2016). TheanoLM — An Extensible Toolkit for Neural Network Language Modeling. Proceedings of INTERSPEECH-2016, pages 3052-3056. International Speech Communication Association (ISCA).

Enarvi, S., Smit, P., Virpioja, S., and Kurimo, M. (2017). Automatic Speech Recognition with Very Large Conversational Finnish and Estonian Vocabularies. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(11): 2085-2097.

Han, H., Manavoglu, E., Zha, H., Tsioutsiouliklis, K., Giles, C. L., and Zhang, X. (2005). Rule-based word clustering for document metadata extraction. Proceedings of the 2005 ACM symposium on Applied computing, pages 1049-1053.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*. 9 (8): 1735–1780.

Jokisch, O., Wagner, A., Sabo, R., Jaeckel, R., Cylwik, N., Rusko, M., Ronzhin A., and Hoffmann, R. (2009). Multilingual speech data collection for the assessment of pronunciation and prosody in a language learning system. Proceedings of SPECOM'2009, pages 515–520.

Karpov, A., Kipyatkova, I., and Ronzhin, A. (2011). Very Large Vocabulary ASR for Spoken Russian with Syntactic and Morphemic Analysis. Proceedings of Interspeech'2011, Florence, Italy, pages 3161–3164. International Speech Communication Association (ISCA).

Karpov, A., Markov, K., Kipyatkova, I., Vazhenina, D., and Ronzhin, A. (2014). Large vocabulary Russian speech recognition using syntactico-statistical language modeling. *Speech Communication*. 56: 213-228.

Karpov, A.A., and Ronzhin, A.L. (2009). Information enquiry kiosk with multimodal user interface. *Pattern Recognition and Image Analysis*. 19(3): 546–558.

Kipyatkova, I. (2017). Experimenting with Hybrid TDNN/HMM Acoustic Models for Russian Speech Recognition. *SPECOM-2017. Lecture Notes in Computer Science,* Springer, LNCS 10458, pp. 362-369.

Kipyatkova, I. (2018). Improving Russian LVCSR Using Deep Neural Networks for Acoustic and Language Modeling. *SPECOM-2018. Lecture Notes in Computer Science*, Springer, LNAI vol. 11096, pp. 291-300.

Kipyatkova, I. (2019). LSTM-Based Language Models for Very Large Vocabulary Continuous Russian Speech Recognition System. *SPECOM 2019*, *Lecture Notes in Computer Science*, Springer LNAI 11658, pp. 219-226.

Kipyatkova, I. and Karpov, A. (2013). Lexicon Size and Language Model Order Optimization for Russian LVCSR. *SPECOM 2013*, *Lecture Notes in Computer Science*, Springer LNAI 8113, pp. 219-226.

Kipyatkova, I. and Karpov, A. (2016). Variants of deep artificial neural networks for speech recognition systems. *SPIIRAS Proceedings*, 6(49): 80-103.

Kumar, S., Nirschl, M., Holtmann-Rice, D., Liao, H., Suresh, A. T., and Yu, F. (2017). Lattice rescoring strategies for long short term memory language models in speech recognition. Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), pages 165-172.

Medennikov, I. and Bulusheva, A. (2016). LSTM-based language models for spontaneous speech recognition. *SPECOM 2016. Lecture Notes in Computer Science*, Springer LNCS, Vol. 9811, pp. 469-475.

Mikolov T., Chen K., Corrado G., and Dean J. (2013). Efficient estimation of word representations in vector space. Proceedings of Workshop at ICLR.

Mikolov. T., Kombrink. S., Deoras. A., Burget. L., and Černocký, J. (2011). RNNLM - Recurrent Neural Network Language Modeling Toolkit. Proceedings of ASRU'2011, pages 196-201.

Morin, F. and Bengio, Y. (2005). Hierarchical probabilistic neural network language model. In R. G. Cowell and Z. Ghahramani, editors, Proceedings of International Workshop on Artificial Intelligence and Statistics (AISTATS), pages 246–252.

Povey, D. et al. (2011). The Kaldi speech recognition toolkit. Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding ASRU.

Ryazanova-Clarke, L., and Wade, T. (2002). The Russian Language Today Routledge". Language Arts & Disciplines.

Saon, G., Soltau, H., Nahamoo, D., and Picheny, M. (2013). Speaker adaptation of neural network acoustic models using i-Vectors. Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), pages 55–59.

Sokirko, A. (2004). Morphological modules on the website www.aot.ru. Proceedings of "Dialogue-2004", Protvino, Russia, pages 559–564 (in Rus.).

Song, M., Zhao, Y., and Wang, S. (2017). Exploiting different word clusterings for class-based RNN language modeling in speech recognition. In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2017), pages 5735-5739.

Srivastava, R. K., Greff, K., and Schmidhuber, J. (2015). Highway networks. arXiv preprint arXiv:1505.00387.

Stepanova, S.B. (1988). Phonetic features of Russian speech: realization and transcription, PhD thesis. (in Russian).

Stolcke, A., Zheng, J., Wang, W., and Abrash, V. (2011). SRILM at Sixteen: Update and Outlook. Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop ASRU'2011.

Sundermeyer, M., Ney, H., and Schlüter, R. (2015). From feedforward to recurrent LSTM neural networks for language modeling. *IIEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(3): 517-529.

Vazhenina, D. and Markov, K. (2013). Evaluation of advanced language modelling techniques for Russian LVCSR. *SPECOM 2013*. *Lecture Notes in Computer Science*, Springer LNAI 8113, pp. 124-131.

Verkhodanova, V., Ronzhin, Al., Kipyatkova, I., Ivanko, D., Karpov, A., and Železný, M. (2016). HAVRUS Corpus: High-Speed Recordings of Audio-Visual Russian Speech. *SPECOM 2016*. *Lecture Notes in Computer Science*, Springer LNCS, Vol. 9811, pp. 338-345.

Verwimp, L, Van hamme, H, and Wambacq, P. (2018). TF-LM: TensorFlow-based Language Modeling Toolkit. Proceedings International Conference on Language Resources and Evaluation (LREC'2018), Miyazaki, Japan, 9-11 May 2018, pages 2968-2973. European Language Resource Association (ELRA).