Advances in STC Russian Spontaneous Speech Recognition System

Ivan Medennikov[1,2(✉)] and Alexey Prudnikov[2,3]

[1] STC-innovations Ltd, St. Petersburg, Russia
[2] ITMO University, St. Petersburg, Russia
{medennikov,prudnikov}@speechpro.com
[3] Speech Technology Center Ltd, St. Petersburg, Russia

Abstract. In this paper we present the latest improvements to the Russian spontaneous speech recognition system developed in Speech Technology Center (STC). Significant word error rate (WER) reduction was obtained by applying hypothesis rescoring with sophisticated language models. These were the Recurrent Neural Network Language Model and regularized Long-Short Term Memory Language Model. For acoustic modeling we used the deep neural network (DNN) trained with speaker-dependent bottleneck features, similar to our previous system. This DNN was combined with the deep Bidirectional Long Short-Term Memory acoustic model by the use of score fusion. The resulting system achieves WER of 16.4 %, with an absolute reduction of 8.7 % and relative reduction of 34.7 % compared to our previous system result on this test set.

Keywords: Spontaneous speech recognition · Bottleneck features · Deep neural networks · Recurrent neural networks

1 Introduction

Spontaneous conversational speech recognition is one of the most difficult tasks in the field of automatic speech recognition (ASR). The difficulties are due to the following characteristics of spontaneous conversational speech: high channel and speaker variability, presence of additive and non-linear distortions, accents and emotional speech, diversity of speaking styles, speech rate variability, reductions and weakened articulation.

There is a large number of studies on recognizing English spontaneous speech, such as [1–5]. Systems proposed in these papers demonstrate high effectiveness, which makes it possible to use them in commercial applications. As far as we know, the state-of-the-art English spontaneous speech recognition system [4] achieves word error rate (WER) of 8.0 % on the Switchboard part and 14.1 % on the CallHome part of the HUB5 2000 evaluation set. This impressive results were obtained by combining various effective techniques of acoustic and language modeling.

A. Ronzhin et al. (Eds.): SPECOM 2016, LNAI 9811, pp. 116–123, 2016.
DOI: 10.1007/978-3-319-43958-7_13

Our goal is to build a speaker-independent system for high-quality Russian spontaneous speech recognition. At present none of the Russian spontaneous speech recognition systems provide recognition accuracy comparable with the above-mentioned English systems. We would like to highlight two reasons of this. First, there are not available training and evaluation datasets for the Russian language, such as the Switchboard and Fisher English speech corpora and the HUB5 2000 evaluation set. Second, Russian is an inflective language with a several times larger number of unique words than English. Moreover, the Russian language is characterized by a relatively free word order in a sentence. This considerably complicates the recognition task [6]. Our previous system achieved WER of 25.1 % [7]. In this work we present the set of recent improvements of the system.

The rest of this paper is organized as follows. Section 2 contains the experimental setup description. Section 3 presents the acoustic modeling approach based on speaker-dependent bottleneck features. Section 4 describes deep BLSTM acoustic models and score fusion of DNN and BLSTM acoustic models (AMs). Section 5 presents the experiments on hypothesis rescoring with language models (LMs) based on Recurrent Neural Networks (RNNs). Finally, Sect. 6 concludes the paper and discusses future work.

2 Experimental Setup

For experiments we used the Kaldi speech recognition toolkit [8]. AM training was performed using a 390 h Russian spontaneous speech dataset (telephone channel, several hundreds of speakers). A test set consisted of about 1 h of Russian telephone conversations. Both training and test sets are the same as used in our previous work [7].

Language models training data consisted of 2 datasets. The first one contained the transcriptions of the AM training dataset. The second one was a large amount (about 200 M words) of texts from Internet forum discussions, books and subtitles from the OpenSubtitles site. The baseline 3-g language model with a vocabulary of 214 K words was built in the SRILM Toolkit [9]. It was obtained by interpolation of 3-g LMs trained on the first and second datasets using Modified Kneser-Ney smoothing. The size of this model was reduced to 4.16 M bigrams and 2.49 M trigrams by the use of pruning.

3 Speaker-Dependent Bottleneck Features

Here we describe the acoustic modeling approach based on speaker-dependent bottleneck (SDBN) features. This approach was proposed in our previous works [7,10]. Its underlying idea is to extract high-level features from the DNN model, which is adapted to the speaker and acoustic environment by the use of i-vectors. The extracted features are applied to training another acoustic model (Fig. 1).

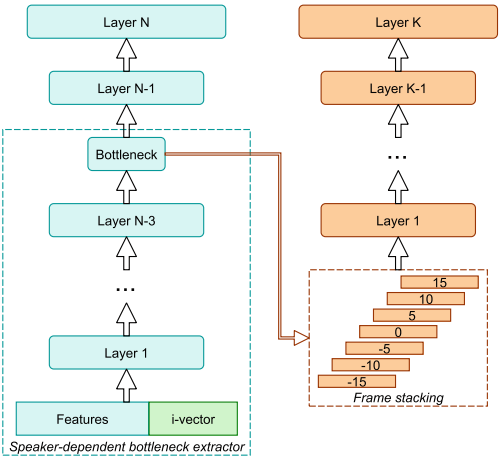

Fig. 1. Speaker-dependent bottleneck approach scheme

Our approach consists of the following main steps:

1. Training the DNN model on the source features using the Cross-Entropy (CE) criterion.
2. Expanding an input layer of the DNN trained at the first step and retraining using an input feature vector appended with i-vector. The regularizing term

$$R = \lambda \sum_{l=1}^{L} \sum_{i=1}^{N_l} \sum_{j=1}^{N_{l-1}} (\mathbf{W}_{ij}^{l} - \bar{\mathbf{W}}_{ij}^{l})^2 \tag{1}$$

   is added to the CE criterion for penalizing parameters deviation from the source model. Here $\mathbf{W}^l$ and $\bar{\mathbf{W}}^l$ are weight matrices of l-th layer ($1 \leq l \leq L$) of the current and the source DNNs, N_l is the size of l-th layer, and N_0 is the dimension of the input feature vector.
3. Transforming the last hidden layer into two layers. The first one is a bottleneck layer with the weight matrix $\mathbf{W}_{bn}$, a zero bias vector and linear activation function. The second one is a non-linear layer with the dimension of the source layer, with weight matrix $\mathbf{W}_{out}$ and the original bias vector $\mathbf{b}$, activation function f and the dimension of the source layer:

$$\mathbf{y} = f(\mathbf{W}\mathbf{x} + \mathbf{b}) \approx f(\mathbf{W}_{out}(\mathbf{W}_{bn}\mathbf{x} + \mathbf{0}) + \mathbf{b}). \tag{2}$$

   These layers are formed by applying Singular Value Decomposition (SVD) to the weight matrix $\mathbf{W}$ of the source layer:

$$\mathbf{W} = \mathbf{U}\mathbf{S}\mathbf{V}^T \approx \tilde{\mathbf{U}}_{bn}\tilde{\mathbf{V}}_{bn}^T = \mathbf{W}_{out}\mathbf{W}_{bn}, \tag{3}$$

   where $_{bn}$ designates reduced dimension.

4. Retraining the network formed at the previous step using the CE criterion with the penalty (1) for parameters deviation from original values.
5. Discarding all layers after the bottleneck and extracting high-level SDBN features using the resulting DNN.
6. Training the GMM-HMM acoustic model using the constructed SDBN features and generating the senone alignment of the training data.
7. Training the final DNN-HMM acoustic model using SDBN features and the generated alignment.

The extractor of 120-dimensional SDBN features was trained using the presented approach. Training was carried out using 23-dimensional log mel filterbank energy (FBANK) features with Cepstral Mean Normalization (CMN), appended with the first and second order derivatives. These features were taken with the temporal context of 11 frames ($\pm$ 5) and appended with an i-vector. We applied 50-dimensional i-vectors extracted by the use of the Universal Background Model with 512 Gaussian, which was trained with our toolset [11] on the full 390 hour training set. We applied the following configuration of the basic network: 6 hidden layers with 1536 sigmoidal neurons in each, the output softmax layer with about 13000 neurons corresponding to senones of the GMM-HMM acoustic model. DNN parameters were updated using the Nesterov Accelerated Gradient algorithm with the momentum value equal to 0.7. Extractor training was initialized using the algorithm presented in the paper [12].

DNN training with the constructed SDBN features (SDBN-DNN) was performed using the temporal context of 31 frames taking every 5th frame. We applied the following DNN configuration: 4 sigmoidal hidden layers with 2048 neurons in each, the output softmax layer with about 13000 neurons corresponding to senones of the GMM-HMM model, which was trained using the same SDBN features. The training was carried out with the CE criterion and the state-level Minimum Bayes Risk (sMBR) sequence-discriminative criterion.

Table 1. SDBN results

Acoustic model	Training criterion	WER, %
DNN-ivec	CE	23.8
SDBN-DNN	CE	22.0 (−1.8)
DNN-ivec	sMBR	21.7
SDBN-DNN	sMBR	19.5 (−2.2)

Table 1 gives the comparison of SDBN-DNN and DNN trained in a speaker adaptive manner using i-vectors (DNN-ivec). It can be seen that the SDBN approach provides a significant gain. Note that SDBN-DNN WER of 19.5 % is much lower than the result of our previous system (25.1 % WER). This is due to the larger SDBN features extractor, more careful tuning of the AM training procedure and the larger language model.

4 Deep Bidirectional Long Short-Term Memory Recurrent Neural Networks

Acoustic models based on deep Bidirectional Long Short-Term Memory (BLSTM) recurrent neural networks demonstrate high effectiveness in various ASR tasks [5,13]. In this section we describe our experiments with these models carried out with the *nnet3* setup of the Kaldi speech recognition toolkit.

We used BLSTM architecture with projection layers described in the paper [13]. The following configuration of the network was applied: 3 forward and 3 backward layers, 1024 cell and hidden dimensions, 128 recurrent and non-recurrent projection dimensions. Training examples consisted of chunks of 20 frames with additional left context of 40 frames and right context of 40 frames. We performed 8 epochs of cross-entropy training with an initial learning rate of 0.0003 and final learning rate of 0.00003. Model parameters were updated using BPTT algorithm with the momentum value equal to 0.5. The models obtained at the iterations of the last epoch were combined into the final BLSTM model. For BLSTM training we used 23-dimensional log mel filterbank energy (FBANK) features with CMN with the first and second order derivatives, appended with the 50-dimensional i-vector described before. Training data alignments prepared using the SDBN-DNN acoustic model were used for the training.

4.1 Score Fusion of SDBN-DNN and BLSTM Acoustic Models

The underlying idea of the score fusion technique is in combining the benefits of both different model architectures and different input features. In this subsection we analyze effectiveness of this technique applied to SDBN-DNN and BLSTM acoustic models. We used log-likelihoods (LLH) determined by the formula

$$\mathrm{LLH} = \alpha \log\left(\frac{\mathrm{P}_1(\mathbf{s}|\mathbf{x})}{\mathrm{P}_1(\mathbf{s})}\right) + (1-\alpha)\log\left(\frac{\mathrm{P}_2(\mathbf{s}|\mathbf{x})}{\mathrm{P}_2(\mathbf{s})}\right) \tag{4}$$

for the decoding with fusion of these acoustic models. Here $\mathrm{P}_1(\mathbf{s}|\mathbf{x})$ and $\mathrm{P}_2(\mathbf{s}|\mathbf{x})$ are posterior probabilities of state $\mathbf{s}$ given an input vector $\mathbf{x}$ on the current frame, $\mathrm{P}_1(\mathbf{s})$ and $\mathrm{P}_2(\mathbf{s})$ are prior probabilities of state $\mathbf{s}$ for SDBN-DNN and BLSTM models respectively. We estimated prior probability of state $\mathbf{s}$ as average posterior probability calculated with the corresponding model on the training data. α value was chosen equal to 0.5. The results of deep BLSTM acoustic model and score fusion are given in Table 2.

Table 2. Deep BSLTM acoustic models and score fusion results

Acoustic model	WER, %
SDBN-DNN	19.5
BLSTM	19.8
score fusion	17.8 (−1.7)

5 RNN-based Language Models

In this section we describe the experiments with sophisticated language models based on recurrent neural networks. Word lattices obtained on the decoding pass with the 3-g LM and the best DNN+BLSTM models fusion in subsection 4.1 were taken as a starting point for these experiments.

We trained two RNN-based language models on shuffled utterances from transcriptions of the AM training dataset. To speed-up the training we used the vocabulary of 45 K most frequent words. All other words were replaced with the <UNK> token. Utterances were divided into two parts: a valid set (15 K utterances) and a train set (all other, 243 K utterances).

Table 3. Rescoring results

Language model	WER, %
3-g	17.8
RNNLM	17.4 (−0.4)
LSTM-LM (medium)	16.7 (−1.1)
LSTM-LM (large)	16.4 (−1.4)

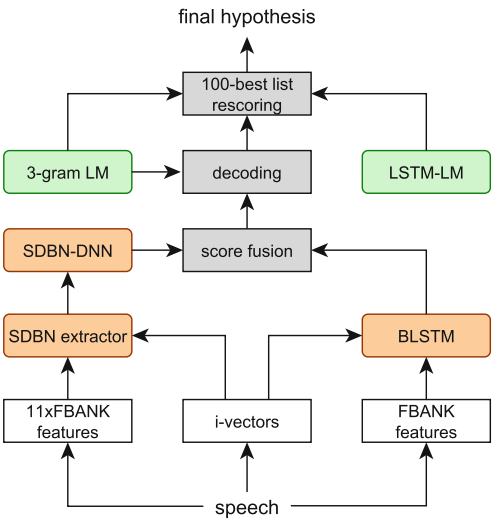

Fig. 2. System architecture

The first RNN-based LM was the Recurrent Neural Network Language Model (RNNLM) [14] which significantly outperforms n-gram LMs in various speech

recognition tasks. We applied the following RNNLM configuration: 256 neurons in the hidden layer and 200 classes in the output layer.

The second RNN-based LM was the LSTM recurrent neural network LM (LSTM-LM) trained with dropout regularization [15]. We trained two LSTM-LMs using the Tensorflow toolkit [16]: "medium" (2 layers with 650 units each, 50 % dropout on the non-recurrent connections) and "large" (2 layers with 1500 units each, 65 % dropout on the non-recurrent connections) configurations from the paper [15].

The trained RNNLM and both LSTM-LMs were applied for hypothesis rescoring. We generated 100-best lists from the word lattices using Kaldi scripts. For the rescoring we took the weighted sum of n-gram LM and RNN-based LM scores. If the sentence contained a word missing in the 45K RNN vocabulary, we added an unigram score of this word from the 3-g model to the RNN score. The results of the rescoring are given in Table 3. It can be seen that RNNLM provided substantial improvement over the n-gram LM, as well as LSTM-LM over RNNLM.

6 Conclusion

The architecture of our system is depicted in Fig. 2. The system achieves WER of 16.4 %, with an absolute reduction of 8.7 % and relative reduction of 34.7 % over our previous system.

We consider several ways of further improvement of our system. First, BLSTM acoustic models improving techniques, such as sequence-discriminative training and dropout regularization, can lead to substantial WER reduction. Second, significant acoustic models improvement can be obtained by the use of the data augmentation approach [17]. Last but not least, we plan to carry out experiments with other promising language model architectures as well as to investigate more complicated approaches of applying sophisticated language models than simple n-best rescoring.

Acknowledgement. This work was financially supported by the Ministry of Education and Science of the Russian Federation, Contract 14.579.21.0057 (ID RFMEFI57914X0057).

References

1. Vesely, K., Ghoshal, A., Burget, L., Povey, D.: Sequence-discriminative training of deep neural networks. In: 14th Annual Conference of the International Speech Communication Association (Interspeech), pp. 2345–2349. Lyon (2013)
2. Saon, G., Soltau, H., Nahamoo, D., Picheny, M.: Speaker adaptation of neural network acoustic models using i-vectors. In: IEEE workshop on Automatic Speech Recognition and Understanding (ASRU), pp. 55–59. Olomouc (2013)
3. Soltau, H., Saon, G., Sainath, T.N.: Joint training of convolutional and non-convolutional neural networks. In: 39th International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5572–5576. Florence (2014)

4. Saon, G., Kuo, H.-K., Rennie, S., Picheny, M.: The IBM 2015 english conversational telephone speech recognition system. In: 16th Annual Conference of the International Speech Communication Association (Interspeech). Dresden (2015)
5. Mohamed, A., Seide, F., Yu, D., Droppo, J., Stolcke, A., Zweig, G., Penn, G.: Deep bi-directional recurrent networks over spectral windows. In: IEEE workshop on Automatic Speech Recognition and Understanding (ASRU), pp. 55–59. Scottsdale (2015)
6. Tampel, I.B.: Automatic speech recognition -the main stages over last 50 years. Sci. Tech. J. Inf. Technol. Mech. Opt. **15**(6), 957–968 (2015). doi:10.17586/2226-1494-2015-15-6-957-968
7. Prudnikov, A., Medennikov, I., Mendelev, V., Korenevsky, M., Khokhlov, Y.: Improving acoustic models for russian spontaneous speech recognition. In: Ronzhin, A., Potapova, R., Fakotakis, N. (eds.) SPECOM 2015. LNCS, vol. 9319, pp. 234–242. Springer, Heidelberg (2015)
8. Povey, D., et al.: The kaldi speech recognition toolkit. In: IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), pp. 1–4. Big Island (2011)
9. Stolcke, A.: SRILM – an extensible language modeling toolkit. In: Seventh International Conference on Spoken Language Processing, vol. 3, pp. 901–904 (2002)
10. Medennikov, I.P.: Speaker-dependent features for spontaneous speech recognition. Sci. Tech. J. Inf. Technol. Mech. Opt. **16**(1), 195–197 (2016). doi:10.17586/2226-1494-2016-16-1-195-197
11. Kozlov, A., Kudashev, O., Matveev, Y., Pekhovsky, T., Simonchik, K., Shulipa, A.: SVID speaker recognition system for NIST SRE 2012. In: Železný, M., Habernal, I., Ronzhin, A. (eds.) SPECOM 2013. LNCS, vol. 8113, pp. 278–285. Springer, Heidelberg (2013)
12. Medennikov, I.P.: Two-step algorithm of training initialization for acoustic models based on deep neural networks. Sci. Tech. J. Inf. Technol. Mech. Opt. **16**(2), 379–381 (2016). doi:10.17586/2226-1494-2016-16-2-379-381
13. Sak, H., Senior, A., Beaufays, F.: Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In: 15th Annual Conference of the International Speech Communication Association (Interspeech). Singapore (2014)
14. Mikolov, T., Karafiat, M., Burget, L., Cernocky, J., Khudanpur, S.: Recurrent neural network based language model. In: 11th Annual Conference of the International Speech Communication Association (Interspeech), pp. 1045–1048. Makuhari (2010)
15. Zaremba, W., Sutskever, I., Vinyals, O.: Recurrent neural network regularization. arXiv preprint (2014). arXiv:1409.2329
16. Abadi, M., et al.: TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems (2015). http://tensorflow.org/
17. Ko, T., Peddinti, V., Povey, D., Khudanpur, S.: Audio augmentation for speech recognition. In: 16th Annual Conference of the International Speech Communication Association (Interspeech). Dresden (2015)