Ngoc Thanh Nguyen · Kietikul Jearanaitanakij ·
Ali Selamat · Bogdan Trawiński ·
Suphamit Chittayasothorn (Eds.)

LNAI 12034

Intelligent Information and Database Systems

12th Asian Conference, ACIIDS 2020
Phuket, Thailand, March 23–26, 2020
Proceedings, Part II

Springer

Editors
Ngoc Thanh Nguyen
Department of Applied Informatics
Wrocław University of Science
and Technology
Wrocław, Poland

Ali Selamat
Faculty of Computer
Science and Information
University Teknologi Malaysia
Kuala Lumpur, Malaysia

Suphamit Chittayasothorn
King Mongkut's Institute
of Technology Ladkrabang
Bangkok, Thailand

Kietikul Jearanaitanakij
King Mongkut's Institute
of Technology Ladkrabang
Bangkok, Thailand

Bogdan Trawiński
Department of Applied Informatics
Wrocław University of Science
and Technology
Wrocław, Poland

ISSN 0302-9743 ISSN 1611-3349 (electronic)
Lecture Notes in Artificial Intelligence
ISBN 978-3-030-42057-4 ISBN 978-3-030-42058-1 (eBook)
https://doi.org/10.1007/978-3-030-42058-1

LNCS Sublibrary: SL7 – Artificial Intelligence

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

End-to-End Speech Recognition in Agglutinative Languages

Orken Mamyrbayev[1,3(✉)], Keylan Alimhan[2], Bagashar Zhumazhanov[1], Tolganay Turdalykyzy[1], and Farida Gusmanova[3]

[1] Institute of Information and Computational Technologies, Almaty 050010, Kazakhstan
morkenj@mail.ru, bagasharj@mail.ru, t_tolganai@inbox.ru
[2] Tokyo Denki University, Tokyo 120-8551, Japan
20787@ms.dendai.ac.jp
[3] Al-Farabi, Kazakh National University, Almaty 050040, Kazakhstan
grfarida77@gmail.com

Abstract. This paper considers end-to-end speech recognition systems based on deep neural networks (DNN). The studies used different types of neural networks, CTC model and attention-based encoder-decoder models. As a result of the study, it was proved that the CTC model works without language models directly for agglutinative languages, but the best is ResNet with 11.52% of CER and 19.57% of WER of using the language model. An experiment with the BLSTM neural network using the attention-based encoder-decoder models showed 8.01% of CER of and 17.91% of WER. Using the experiment, it was proved that without integrating language models, good results can be achieved. The best result showed ResNet.

Keywords: Speech recognition · Agglutinative languages · End-to-End models · Deep learning · CTC

1 Introduction

Speech is a system of human-used audio signals, written signs and symbols to represent, process, store and transmit information. It is also a tool for human-machine interaction [1]. To implement the voice interface requires the participation of a wide range of specialists, namely a computer linguist, DNN programmer, etc. The traditional speech recognition system can be divided into several modules, such as acoustic models, language models and decoding [2]. The modularity design is based on many independent assumptions, and even the traditional acoustic model is trained from frames that depend on the Markov model. In automated speech recognition systems, hidden Markov models (HMM) were popular models to represent the temporal dynamics of speech signals and Gaussian mixture models (GMM) probability density function to represent signal distributions over a stationary short period of time that typically corresponds to a unit of pronunciation. The HMM-GMM model dominated research on automatic speech recognition for several years. Today, the neural network is widely used in the field of speech recognition. Many studies have shown that the use of neural networks at each step of the

N. T. Nguyen et al. (Eds.): ACIIDS 2020, LNAI 12034, pp. 391–401, 2020.
https://doi.org/10.1007/978-3-030-42058-1_33

script of the standard speech recognition system improves the quality of its work. The most popular deep learning approach for automatic speech recognition is the so-called HMM-DNN hybrid architecture, where the HMM structure is preserved and the GMM is replaced by a deep neural network (DNN) to model the dynamic characteristics of speech signals. For example, in studies [3], language models were trained using RNN, in [4] the dictionary was obtained using LSTM networks, in [5] deep neural networks showed high results for constructing acoustic models, in [6] the method of feature extraction using limited Boltzmann machines was presented. Consequently, the idea of using artificial neural networks at all stages of speech recognition appeared.

Deep learning methods using high-performance GPUs in speech recognition have been successfully implemented and this approach has been called the end-to-end method. In the end-to-end approach, when learning a neural network, only one model can produce the desired result without the use of other components and such a model is called an end-to-end. End-to-end networks can be created by adding several convolutional neural network (CNN) and recurrent neural network (RNN) layers, which act as acoustic and language models, and directly correlate speech data at the input with transcription. At the moment, there are several methods of implementing end-to-end models, namely, the connection time classification (CTC) and the attention-based encoder-decoder models, conditional random fields (CRF). In speech recognition problems, special attention is still paid to end-to-end approaches than to traditional methods [7]. Many published works have proved that the success of the results of the end-to-end approach depends on an increase in the amount of data for neural network training. There are applications in the world that work on the basis of an end-to-end approach: Baidu Deep Speech, Google Listen, Attend, Spell, Speech to Translator TTS, Voice to Text Messenger. The main reason for this conclusion is that current end-to-end models are trained based on data. From the above analysis we can see the main problem, it concerns the recognition of a few resource languages, such as Kazakh, Kyrgyz, Azerbaijani, Uighur, Tatar, Turkish, etc. These listed languages are included in the group of agglutinative languages. For agglutinative languages, there are no large corps of training data. Other languages have TIMIT, WSJ, LibriSpeech, AMI and Switchboard which have thousands of hours of training data.

To improve the end-to-end approach in CTC and encoder-decoder models based on the attention mechanism, different variants of networks were introduced. Complex encoders consisting of convolutional neural networks (CNN) were introduced to use local correlations in speech signals. These models take advantage of each submodel and introduce more explicit and strict limitations to the entire model. The above studies in this area significantly improve the performance of end-to-end speech recognition systems. Introducing complex computational layers into a model can use better correlations in both the time and frequency domain, but a model with much more parameters is harder to train. In previous studies, it was determined that deep learning models in different languages are successful, and multitask learning (MTL) is better suited for integrated learning [8, 9].

In the end-to-end speech recognition approach, all signal parameters are determined by calculating gradients that are easily influenced by neural network structures. However, the end-to-end recognition systems of agglutinative languages still do not reach the

modern level of research and are not trained. During the analysis, it was determined that the end-to-end models for recognizing agglutinative languages are not sufficiently trained.

In this paper, we propose the recognition of agglutinative languages, which is aimed at solving the problem with a limited speech resource within the framework of the end-to-end architecture. This study is organized as follows. Section 2 describes research in the relevant scientific field. Section 3 describes the principles of the CTC model and the attention-based encoder-decoder models. Below we present our experimental data and describe the equipment for the experiment. Section 4 analyzes the experimental results. The final section provides conclusions.

2 Related Work

Models based on connective time classification (CTC) for speech recognition work without initial alignment of input and output sequences. CTC was designed to decode the language. Hannun et al. [10] and his team used the Baidu model for speech recognition, which implements a parallel network learning algorithm using CTC.

The use of deep recurrent convolutional networks and deep residual networks in conjunction with CTC was proposed in [11]. The best result was obtained with the use of residual networks with batch normalization. Thus, a PER result of 17.33% was obtained on the TIMIT speech corpus.

An alternative to CTC for end-to-end speech recognition is the Sequence to Sequence (Sec2Sec) models with attention [12]. Such models consist of an encoder and a decoder. The encoder compresses audio frame information into a more compact vector representation by reducing the number of neurons from layer to layer, and the decoder recovers a sequence of symbols, phonemes, or even words based on this compressed representation and recurrent neural network.

In [13], a CTC model was proposed using deep convolutional networks instead of recurrent networks. The best model based on convolutional networks had 10 convolutional layers and 3 fully connected layers. The best PER was 18.2%, while the best PER for bidirectional LSTM networks was 18.3%. Tests were conducted on the TIMIT case. It was also concluded that convolutional networks can increase the speed of learning and are more suitable for learning on phoneme sequences.

In a CTC network, the output values of a neural network themselves represent transition probabilities. Bidirectional LSTM networks were chosen as the architecture of the neural network. Three models were compared: the RNN-CTC model, the RNN-CTC model (RNN-WER), the retrained minimized WER, and the basic hybrid model written using Kaldi tools [14].

Soltau et al. [15] performed context-sensitive phoneme recognition by training a CTC-based model in the task of signing a video on YouTube. Sequence-to-sequence models lack recognition by 13–35% compared to base systems.

Graves et al. [16] trained the end-to-end model the CTC criterion without applying frame-level alignment. The Sequence-to-sequence model simplified the problem of automatic speech recognition by training and optimizing the neural network for the

acoustic model, pronunciation model, and language model. These models also work as multi dialectic systems, since they are jointly modeled in different dialects.

The RNN-CTC model without a language model showed 30.1%, of WER, although the basic model cannot be trained without LM. But even when using trigram LM, the basic model showed 7.8% of WER, RNN-CTC - 8.7%, and RNN-WER - 8.2%. A combination of RNN-CTC and the base model was also tested, which showed the best result, equal to 6.7%. The Wall Street Journal corpus [17] was used as a speech corpus.

There is a "generalization" of CTC models - the RNN Transducer, which combines two RNNs into a serial Converter system [18]. One of the networks is similar to a CTC network and processes the same moment of time as the input sequence, and the second RNN models the probabilities of the following labels under the condition of the previous one. As in CTC networks, dynamic programming is used for calculations and the forward-backward algorithm, but taking into account the limitations of both RNNs. Unlike CTC networks, the RNN Transducer allows to generate output sequences longer than input ones. RNN Transducers showed good results in recognition of phonemes with 17.7% of PER on the TIMIT corpus.

In [19, 20] three models with CTC were considered: ResNet, BLSTM and combination of LSTM and CNN. A method for combining models similar to ROVER has also been proposed. So, the result was obtained on the WSJ speech data set using ResNet with 8.99% of WER, and using a combination of the three models mentioned above is 7.65%.

Beyond speech recognition, the neural network achieves success in other fields such as natural language processing [21, 22]. In [21], the author presented a deep learning-based model for Kazakh named entity recognition by projecting the word, root and entity label into a vector space. In [22], a neural network for morphological disambiguation was proposed, which learns context embedding from characters by double-layer of BLSTM and compute the similarity score between context and the corresponding morphological analyses. The idea was that the correct analysis should be the most similar to the context.

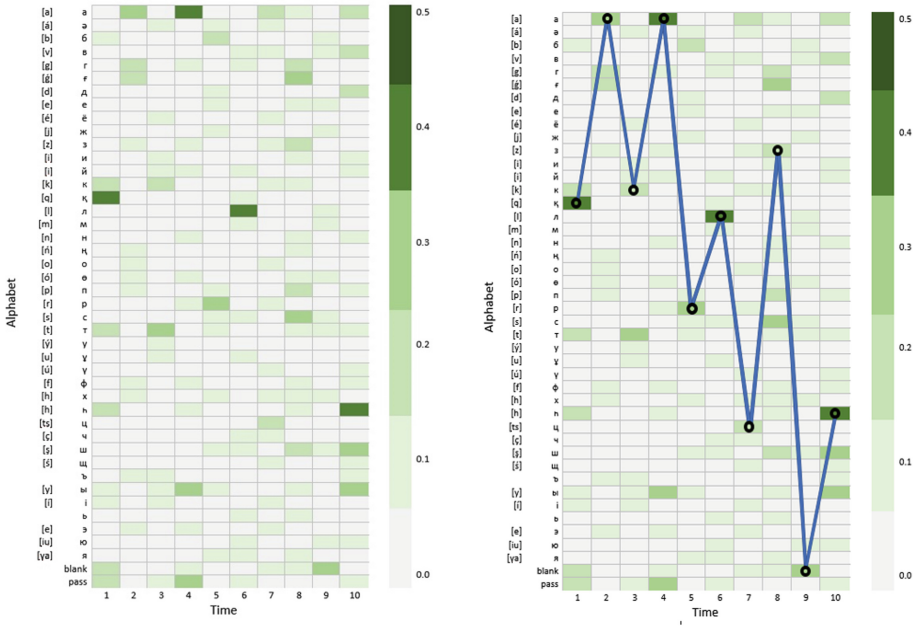
3 Proposed Automatic Speech Recognition System

The methodology of our work is as follows:

3.1 CTC

To train a neural network, the CTC function is used as a loss function. The output sequence of a neural network can be described as follows: $y = f_w(x)$. The output layer of the neural network contains one block for each symbol of the output sequence and one more for the additional "blank" symbol. Each element of the output sequence is a probability distribution vector for each symbol G' at time t. Thus, the element $\boldsymbol{y_k^t}$ is the probability that at time **t** in the input sequence the symbol **k** from the set G' is pronounced (Fig. 1 (a)).

Let, α be a sequence of blanks indices and symbols of length T, according to x. The probability $P(\alpha|x)$ can be represented as the product of the probabilities of the



(a) (b)

Fig. 1. The matrix predicted by the acoustic model.

appearance of symbols at each moment of time:

$$P(\alpha|x) = \prod_{t=1}^{T} y_{\alpha_t}^t, \forall \alpha \in G'^T \tag{1}$$

Let B be the operator that removes symbol repeats and blanks.

$$P(y|x) = \sum_{\alpha \in B^{-1}(y)} P(\alpha|x) \tag{2}$$

The above formula is calculated using dynamic programming, and the neural network will be trained to minimize the CTC function:

$$CTC(x) = -\ln(P(y|x)) \tag{3}$$

Decoding is based on the following assumption:

$$\arg\max_{w} P(y|x) \approx B\left(\alpha^*\right) \tag{4}$$

where $\alpha^* = \arg max_\alpha P(\alpha|x)$. The results of the assumption can be seen in Fig. 1 (b).

3.2 Attention-Based Model

Attention is an Encoder-Decoder mechanism designed to improve RNN performance in speech recognition. Encoder is a neural network, such as: DNN, BLSTM, CNN;

transforms the input sequence $x = (x_1, \ldots, x_{L'})$ for feature extraction in some intermediate representation of $h = (h_1, \ldots, h_L)$.

$$h = Encoder(x_1, \ldots, x_{L'}) \tag{5}$$

Decoder is a regular RNN that uses an intermediate representation to generate output sequences:

$$P(y|x) = AttentionDecoder(h, y) \tag{6}$$

As a decoder, we used an attention-based Recurrent Sequence Generator.

Dataset

The data for the analysis was provided by the laboratory of Computer engineering of intelligent systems. To do this, we used a soundproofing, professional recording studio from Vocalbooth.com.

As speakers, people were selected without any problems with the pronunciation of speech. 380 speakers of different ages (age from 18 to 50 years) and genders were used for recording. Scoring and recording of one speaker took an average of 40–50 min. For each speaker was prepared text consisting of 100 sentences, which were recorded in separate files. Each sentence consists of an average of 6–8 words. Sentences are selected with the most rich phoneme of words. Text data was collected from news sites in the Kazakh language, and other materials were used in electronic form. A total of 123 h of audio data were recorded. During recording, transcriptions were created - a description of each audio file in a text file. The created corpus allows, firstly, to work with large volumes of databases, to check the proposed characteristics of the system and, secondly, to study the impact of database expansion on the recognition speed.

All audio materials have the same characteristics:

- file extension: .wav;
- method of converting to digital form: PCM
- discrete frequency: 44.1 kHz;
- bit capacity: 16 bits;
- number of audio channels: one (mono).

To train the end-to-end recognition system of agglutinative languages, we have chosen 2 corpora:

- Turkish language corpus (9 million words and audio): http://www.tnc.org.tr/
- Tatar language corpus (10 million words and audio): http://www.corpus.antat.ru.

Implementation

End-to-end speech recognition system using CTC function was implemented using TensorFlow. In this system, we used the Eesen toolkit in TensorFlow. This system allows to use language models built in the Kaldi format without additional conversion. We used Tensor2Tensor5 to conduct experiments with attention-based models.

All experiments were carried out using the SuperMicro SYS-7049GP-TRT server. The server configuration has a high-performance NVIDIA TESLA P100 graphics card.

4 Experiments and Results

In the feature extraction experiments, we used the Mel-frequency cepstral coefficients (MFCC) with the first 13 coefficients computed. All training data was divided into training (90%) and cross-validation (10%).

At the second stage of the experiment, we will describe the model results based on the CTC loss function. The results of the corresponding CTC models are presented in Table 1. In our studies, we used several types of neural networks: ResNet, LSTM, MLP, Bidirectional LSTM. Pre-configuration of neural networks without a language model gave us the best results:

Table 1. CTC model results.

Model	CER%	WER%	Decode	Train
	Models that do not use language models			
MLP	48.11	59.26	0.2032	131.2
LSTM	36.43	46.51	0.2152	421.3
Conv+LSTM	34.92	39.31	0.2688	465.2
BLSTM	33.61	37.66	0.2722	491.7
ResNet	32.52	36.57	0.2657	192.6
	Models using language models and MFCC			
MLP	39.11	63.26	0.0192	146.2
LSTM	24.43	46.51	0.0152	521.3
Conv+LSTM	22.92	39.31	0.0088	465.2
BLSTM	13.61	20.66	0.0022	591.7
ResNet	11.52	19.57	0.0051	242.6

- MLP: MLP: there were 6 hidden layers with 1,024 nodes using the ReLU activation function with an initial learning rate of 0.007 and a damping factor of 1.5.
- LSTM: there were 6 layers with 1024 units each with a dropout of 0.5 s, an initial learning rate of 0.001, and a damping factor of 1.5.
- ConvLSTM: one two-dimensional convolutional layer with 8 filters was used, with ReLU activation function. Then it drops out with a retention probability of 0.5.
- BLSTM: used 6 layers with 1,024 units and dropped with a retention probability of 0.5.
- ResNet had 9 residual blocks with batch-normalization.

In the first experiment for the attention-based encoder-decoder models, we used the MFCC algorithm to extract features, and the CTC function was used to train the neural

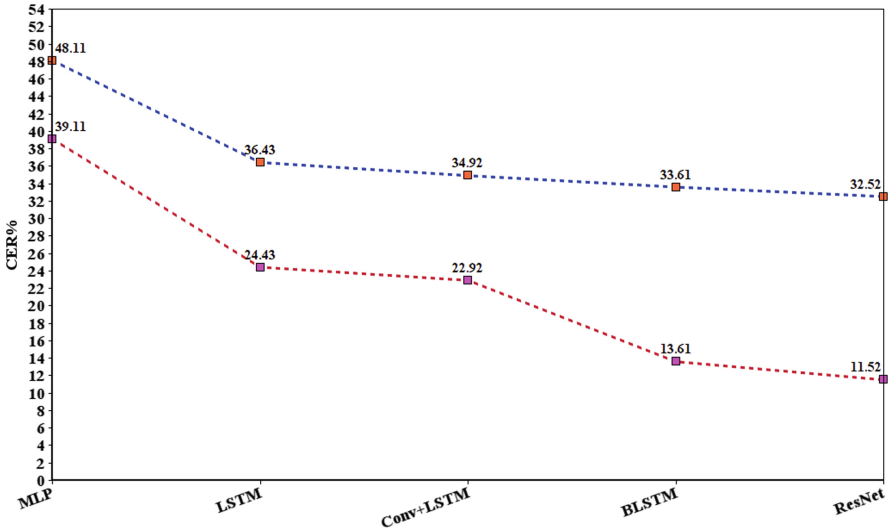


Fig. 2. CTC model results by CER. The blue line is the result of models that do not use language models, as well as the red line - models that use language models and MFCC. (Color figure online)

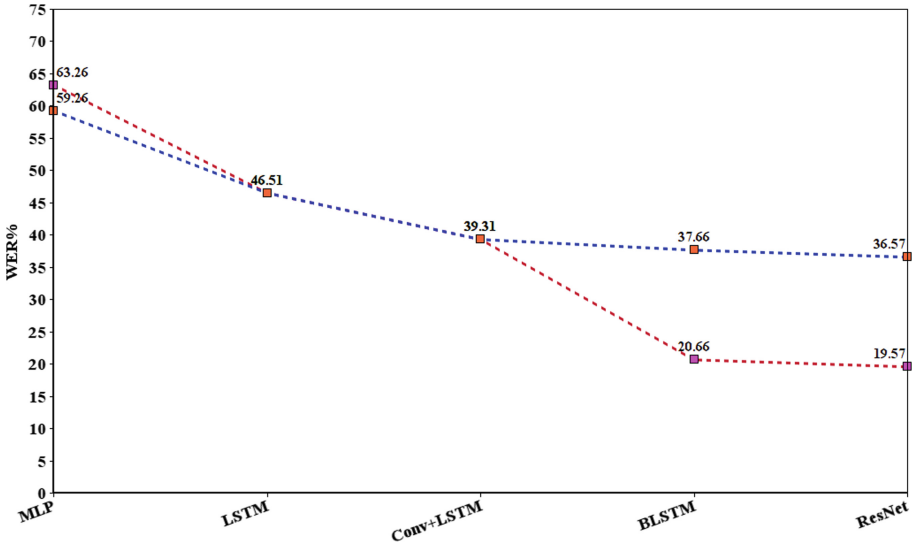


Fig. 3. CTC model results by WER. The blue line is the result of models that do not use language models, and the red line is the result of models that use language models and MFCC. (Color figure online)

network. We did not use language models in this model. In the second experiment, we used MFCC and language models. The results of the experiment can be seen in Figs. 2 and 3.

Table 2. Results of the attention-based encoder-decoder models.

Model	CER%	WER%	Decode	Train
LSTM	8,61	17,58	0,468	476,7
BLSTM	8,01	17,91	0,496	544,3

In the next experiment, we used neural networks LSTM and BLSTM. In our model, 6 layers of 256 units were used with an initial decrease in dropout with a probability of saving in the encoder of 0.7. As a decoder, we used LSTM and an attention-based encoder-decoder models. The results can be seen in Table 2.

Our experiments proved that the CTC model works without language models directly for agglutinative languages, but still the best is ResNet with 11.52% of CER and 19.57% of WER using the language model. Thus, it can be seen that the language model is an important part of speech recognition.

The CTC model makes mistakes in constructing words and sentences from recognized characters, but the resulting phonemic transcription is very similar to the original. But after the experiment, we found that the use of an attention-based encryptor-decoder models for agglutinative languages without integrating language models allows to achieve good results. The BLSTM neural network using the attention-based encoder-decoder models showed 8.01% of CER and 17.91% of WER.

5 Conclusion

In this paper, we consider the problem of recognition of agglutinative languages using an end-to-end approach, such as the CTC model and the attention-based encoder-decoder models. During the experiment we used different types of neural network architectures: MLP, LSTM and their modifications, as well as ResNet. As a result of the experiment, we proved that good results can be achieved without the integration of language models. ResNet showed the best result. In this experiment, good performance was achieved, better than the basic hybrid models.

In the future it is planned to conduct experiments using other types of models for feature extraction and speech recognition. The Conditional Random File model will be applied.

Acknowledgments. This work was supported by the Ministry of Education and Science of the Republic of Kazakhstan. IRN AP05131207 Development of technologies for multilingual automatic speech recognition using deep neural networks.

References

1. Perera, F.P., et al.: Relationship between polycyclic aromatic hydrocarbon–DNA adducts and proximity to the World Trade Center and effects on fetal growth. Environ. Health Perspect. **113**, 1062–1067 (2005)

2. Mamyrbayev, O., Turdalyuly, M., Mekebayev, N., Alimhan, K., Kydyrbekova, A., Turdalykyzy, T.: Automatic recognition of Kazakh speech using deep neural networks. In: Nguyen, N.T., Gaol, F.L., Hong, T.-P., Trawiński, B. (eds.) ACIIDS 2019. LNCS (LNAI), vol. 11432, pp. 465–474. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-14802-7_40
3. Mikolov, T., et al.: Recurrent neural network based language model. Interspeech **2**, 1045–1048 (2010)
4. Rao, K., Peng, F., Sak, H., Beaufays, F.: Grapheme-to-phoneme conversion using long short-term memory recurrent neural networks. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4225–4229 (2015)
5. Jaitly, N., Hinton, G.: Learning a better representation of speech soundwaves using restricted boltzmann machines. In: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5884–5887 (2011)
6. Smolensky, P.: Information processing in dynamical systems: foundations of harmony theory. Colorado University at Boulder Department of Computer Science, pp. 194–281 (1986)
7. Vaněk, J., Zelinka, J., Soutner, D., Psutka, J.: A regularization post layer: an additional way how to make deep neural networks robust. In: Camelin, N., Estève, Y., Martín-Vide, C. (eds.) SLSP 2017. LNCS (LNAI), vol. 10583, pp. 204–214. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-68456-7_17
8. Kim, S., Hori, T., Watanabe, S.: Joint CTC-attention based end-to-end speech recognition using multi-task learning. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2017)
9. Aida-Zade, K., Rustamov, S., Mustafayev, E.: Principles of construction of speech recognition system by the example of Azerbaijan language. In: International Symposium on Innovations in Intelligent Systems and Applications, pp. 378–382 (2009)
10. Hannun, A., et al.: DeepSpeech: scaling up end-to-end speech recognition, arXiv:1412.5567 (2014)
11. Zhang, Z., et al.: Deep recurrent convolutional neural network: improving performance for speech recognition (2016). preprint: arXiv:1611.07174. https://arxiv.org/abs/1611.07174
12. Bahdanau, D., et al.: End-to-end attention-based large vocabulary speech recognition. In: 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4945–4949. IEEE (2016)
13. Zhang, Y., et al.: Towards end-to-end speech recognition with deep convolutional neural networks (2017). preprint: arXiv:1701.02720. https://arxiv.org/abs/1701.02720
14. Povey, D., et al.: The Kaldi speech recognition toolkit. In: IEEE 2011 Workshop on Automatic Speech Recognition and Understanding, 4 p. IEEE Signal Processing Society (2011)
15. Soltau, H., Liao, H., Sak, H.: Neural speech recognizer: acoustic-to-word LSTM model for large vocabulary speech recognition. arXiv:1610.09975 (2016)
16. Graves, A., Fernández, S., Gomez, F., Schmidhuber, J.: Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In: Proceedings of the 23rd International Conference on Machine Learning, pp. 369–376 (2006)
17. Popović, B., Pakoci, E., Pekar, D.: End-to-End large vocabulary speech recognition for the Serbian language. In: Karpov, A., Potapova, R., Mporas, I. (eds.) SPECOM 2017. LNCS (LNAI), vol. 10458, pp. 343–352. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66429-3_33
18. Boulanger-Lewandowski, N., Bengio, Y., Vincent, P.: High-dimensional sequence transduction. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3178–3182 (2013)
19. Wang, Y., Deng, X., Pu, S., Huang, Z.: Residual convolutional CTC networks for automatic speech recognition (2017). preprint: arXiv:1702.07793. https://arxiv.org/abs/1702.07793

20. Rustamov, S., Gasimov, E., Hasanov, R., Jahangirli, S., Mustafayev, E., Usikov, D.: Speech recognition in flight simulator. In: Aegean International Textile and Advanced Engineering Conference. IOP Conference Series: Materials Science and Engineering, vol. 459 (2018)
21. Gulmira, T., Alymzhan, T., Orken, M., Rustam, M.: Neural named entity recognition for Kazakh. In: 20th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing), 7–13 April 2019, La Rochelle, France. Lecture Notes in Computer Science (2019)
22. Toleu, A., Tolegen, G., Makazhanov, A.: Character-aware neural morphological disambiguation. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 666–671. Association for Computational Linguistics, Vancouver (2017)