Experimenting with Attention Mechanisms in Joint CTC-Attention Models for Russian Speech Recognition

Irina Kipyatkova[1,2(✉)] and Nikita Markovnikov[1]

[1] St. Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences (SPIIRAS), St. Petersburg, Russia
kipyatkova@iias.spb.su, niklemark@gmail.com
[2] St. Petersburg State University of Aerospace Instrumentation (SUAI), St. Petersburg, Russia

Abstract. The paper presents an investigation of attention mechanisms in end-to-end Russian speech recognition system created by join Connectional Temporal Classification model and attention-based encoder-decoder. We trained the models on a small dataset of Russian speech with total duration of about 60 h, and performed pretraining of the models using transfer learning with English as non-target language. We experimented with following types of attention mechanism: coverage-based attention and 2D location-aware attention as well as their combination. At the decoding stage we used beam search pruning method and gumbel-softmax function instead of softmax. We have achieved 4% relative word error rate reduction using 2D location-aware attention.

Keywords: End-to-End speech recognition · Attention mechanism · Coverage-based attention · 2D Location-aware attention · Russian speech

1 Introduction

In recent years, development of end-to-end systems became the main trend in speech recognition technologies due to fast advances in deep learning approaches. The end-to-end speech recognition methods are mainly based on two types of models: Connectional Temporal Classification (CTC) and attention-based encoder-decoder, as well as on their combination [1].

For example, an investigation of end-to-end model with CTC is described in [2]. It was shown that such system is able to work without language model (LM) well. Training dataset [3] was made up of audio tracks of Youtube videos with duration more than 650 h. Testing dataset was made up of audio tracks of Google Preferred channels on YouTube [4], its duration was 25 h. The lowest word error rate (WER) obtained without using LM was 13.9% and it was equal to 13.4% with the usage of LM.

Attention based encoder-decoder model was used in [5] for experiments on recognition of speech from LibriSpeech corpus. The authors performed data augmentation by speed, tempo, and/or volume perturbation, sequence-noise injection. The authors obtained WER = 4% on test-clean data and WER = 11.7% on test-other.

A. Karpov and R. Potapova (Eds.): SPECOM 2020, LNAI 12335, pp. 214–222, 2020.
https://doi.org/10.1007/978-3-030-60276-5_22

The application of LSTM-based LM results in decreasing of WER to 2.5% and 8.2% on test-clean and test-other sets respectively.

Joint CTC-attention based end-to-end speech recognition was proposed in [6]. Two loss functions are used in these model, which are combined using weighted sum as follows:

$$L = \lambda L_{CTC} + (1 - \lambda) L_{att},$$

where L_{CTC} is an objective for CTC and L_{att} is an objective for attention-based model, λ is a weight of CTC model, $\lambda \in [0, 1]$.

Different types of attention mechanism in end-to-end speech recognition systems are analyzed in many paper. For example, in [7] an application of Self-attention in CTC model was investigated. The proposed model allowed the authors to outperform the existing end-to-end models (CTC, encoder-decoder, joint CTC/encoder-decoder models) in speech recognition accuracy. In [8], the authors proposed Monotonic Chunkwise Attention (MoChA), which adaptively splits the input sequence into small chunks over which soft attention is computed. The authors applied this attention mechanism for two tasks: online speech recognition and automatic document summarization. Experiments on online speech recognition were performed on Wall Street Journal (WSJ) corpus. WER was equal to 13.2%. However, the authors of another research published in [9] found out that this attention mechanism is unstable in their system and proposed modified attention mechanism called stable MoChA (sMoCha). Moreover, the authors proposed to compute truncated CTC (T-CTC) prefix probability on the segmented audio rather than on the complete audio. At the decoding stage, the authors proposed the dynamic waiting joint decoding (DWDJ) algorithm to collect the decoding hypotheses from the CTC and attention branches in order to deal with the problem that these two branches predict labels asynchronously in beam search. Experiments on online speech recognition showed WER equal to 6% and 16.7% on test-clean and test-other of LibriSpeech respectively.

The aim of the current research was to improve the Russian end-to-end speech recognition system developed in SPIIRAS by modification of attention mechanism in joint CTC-attention based encoder-decoder model. We have investigated with coverage-based attention and 2D location-aware attention as well as their combination. The models were trained and tested with the help of EspNet toolkit [10] with a PyTorch as a back-end part. The developed models were evaluated in terms of Character Error Rate (CER) and word error rate (WER).

The rest of the paper is organized as follows. In Sect. 2 we present our baseline end-to-end speech recognition model, in Sect. 3 we describe modifications of attention mechanism, our Russian speech corpora are presented in Sect. 4, the experimental results are given in Sect. 5, in Sect. 6 we make a conclusion to our work.

2 The Baseline Russian End-to-End Model

As a baseline we used joint CTC-attention based encode-decoder model presented on Fig. 1, where h is an input vector, h is a vector of hidden states obtained from encoder, g_i is a weighted vector obtained from attention mechanism on i-th iteration of decoding, y_i is decoder's output on i-th iteration, w_i is i-th symbol of output sequence, s_{i-1} is the decoder's state on the previous iteration.

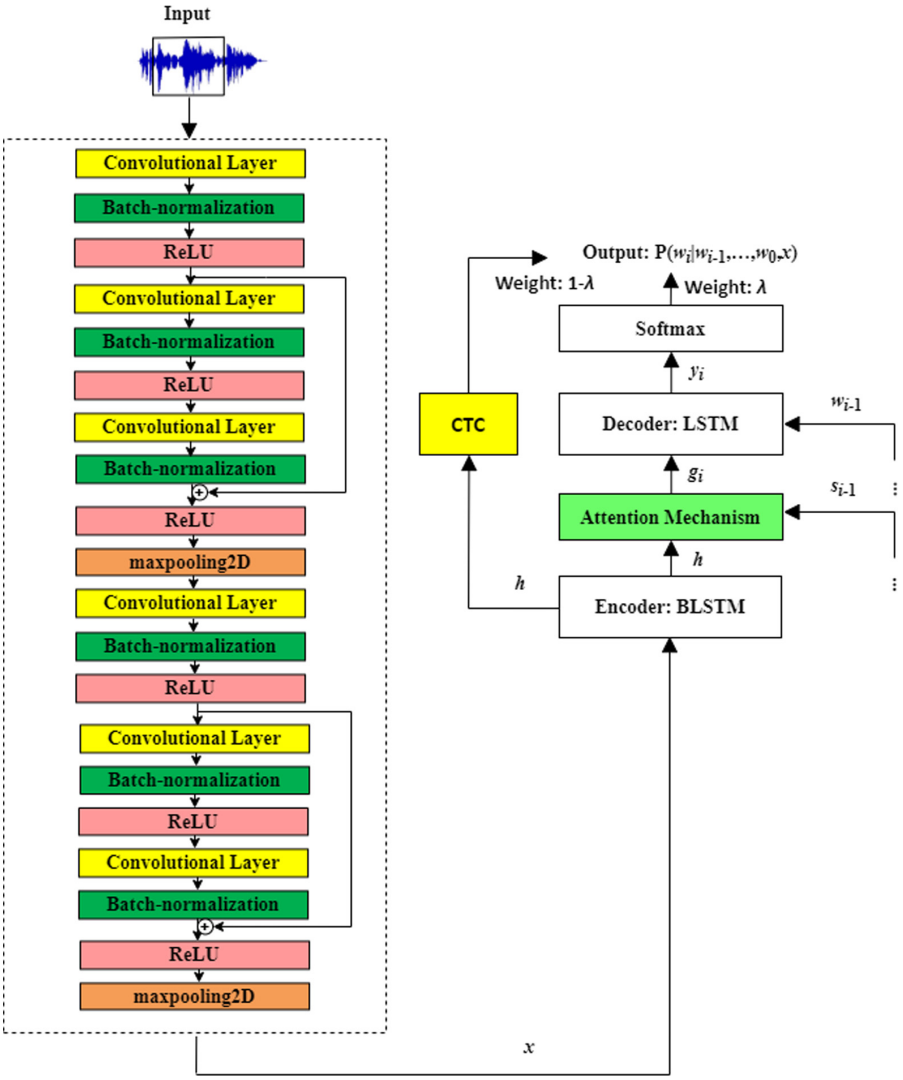

Fig. 1. Joint CTC-attention based encode-decoder model

The model had the following topology. Long Short-Term Memory (LSTM) network contained two layers with 512 cells in each was used as decoder. Bidirectional LSTM (BLSTM) contained five layers with 512 cells in each was used as encoder. Moreover we used highway connection in encoder. In order to prevent our model from making over-confident predictions, we used dropout [11] with probability equal to 0.4 in LSTM layers at every time step in the encoder's network as well we used label smoothing [12] as a regularization mechanism with a smoothing factor of 0.01. Location-aware [13] attention mechanism was used in decoder.

Before the encoder, there was a feature extraction block that was VGG model [14]. Moreover, we added residual connection (ResNet) in this block as it is shown on Fig. 1. Thus, the feature extraction block consisted of two similar parts, which included three convolution layers followed by batch normalization [15] and max-pooling layer; rectified linear unit (ReLU) [16] was used as activation function. The number of output features was equal to 128.

At the training stage, the CTC weight was equal to 0.3. Filter banks features were used as input. The model's training was carried out using the transfer learning approach. At first, the models were pretrained on English speech data from LibriSpeech corpus (we used 360 h of English data for pretraining). Then models were trained on Russian speech data. Our Russian speech corpora are described in detail in Sect. 4.

Due to small size of Russian speech dataset, we additionally used LSTM-based LM at speech recognition experiments. The language model was trained on text corpus collected from online Russian newspapers. The corpus consisted of 350 M words (2.4 GB data). LSTM contained one layer with 512 cells. The vocabulary consisted of 150 K most frequent word-forms from the training text corpus.

3 Attention Mechanisms in Russian End-to-End Speech Recognition Model

3.1 The Baseline Location-Aware Attention

Attention mechanism is a subnetwork in the decoder. Attention mechanism chooses a subsequence of the input and then uses it for updating hidden states of neural network of decoder and predicting an output. On the i-th step decoder generates an output y_i focusing on separate components of h as follows [13]:

$$\alpha_i = \text{Attend}\,(s_{i-1}, \alpha_{i-1}, h);$$

$$g_i = \sum_{j=1}^{L} \alpha_{i,j} h_j,$$

where s_{i-1} is the $(i-1)$-th state of neural network called Generator, $\alpha_i \in \mathbb{R}^L$ are attention weights, vector g_i is called glimpse, Attend denotes a function that calculates attention weight. The step comes to an end with computing a new generator state as $s_i = Recurrency(s_{i-1}, g_i, y_i)$.

In our baseline system we used location-aware attention that calculates as follows:

$$f_i = F * \alpha_{i-1};$$

$$e_{i,j} = w^{\mathrm{T}} \tanh\left(Ws_{i-1} + Vh_j + Uf_{i,j} + b\right),$$

where $w \in \mathbb{R}^m$ and $b \in \mathbb{R}^n$ denote weight vector, $W \in \mathbb{R}^{m \times n}$, $V \in \mathbb{R}^{m \times 2n}$, and $U \in \mathbb{R}^{m \times k}$ are weight matrices, n and m are number of hidden units in the encoder and decoder respectively, vector $f_{i,j} \in \mathbb{R}^k$ are convolutional features. Generally, an attention weights matrix is calculated as $\alpha_i = softmax_i(e)$.

3.2 Coverage-Based Attention

The attention mechanism usually takes into account only the previous time step when calculating the matrix of weights. However, during research and experiments, it was found out that models often miss some, mostly short, words, for example, prepositions. Such words are often swallowed in pronunciation. In [17] a method of taking into account all preceding history of weight matrix was proposed for text summarization. In this method coverage vector is computed as follows [18]:

$$\beta_i = \sum_{l=0}^{i-1} \alpha_l,$$

where α_l is weight matrix of neural network in attention mechanism at l-th time step, i is an index of the current time step. Thus, in this case weight matrix in the attention mechanism is the sum of attention weights at all preceding time steps. The coverage vector is used as extra input to the attention mechanism as follows:

$$e_{i,j} = w^{\mathrm{T}} \tanh\left(Ws_{i-1} + Vh_j + U\beta_{i,j} + b\right).$$

3.3 2D Location-Aware Attention

The general location attention takes into account one frame to compute attention weight vector. Therefore, it was suggested [13] to take into account several frames for computation of weight vector using fixed-size window on frames. It should be noted that convolution in location attention is carried out with the help of a kernel of the size (1, K), where K is the given hyperparameter of the model. It was suggested to use the kernel of the size (w, K), where w is the width of window on frames. At each iteration the window is shifted on one frame updating the obtained matrix. We set w = 5 in our experiments.

3.4 Joint Coverage-Based and 2D Location-Aware Attention

As well we performed combination of coverage-based and 2D location-aware attention mechanisms. In this case attention is computed over 5 frames as 2D location-aware

attention and then coverage vector is compute as the sum of attention distributions over all previous decoder timesteps.

4 Speech Datasets

For training the acoustic models, we used three corpora of Russian speech recorded at SPIIRAS [19, 20]:

- the speech database developed within the framework of the EuroNounce project [21] that consists of recordings of 50 speakers, each of them pronounced a set of 327 phonetically rich and meaningful phrases and texts;
- 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 [22] (these phrases were different for each speaker), and 55 common phrases were taken from a phonetically representative text, presented in [23];
- the audio part of the audio-visual speech corpus HAVRUS [24] 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 [25] and 50 phrases were 7-digits telephone numbers (connected digits);

In addition, we supplemented our speech data with free available speech corpora:

- Voxforge[1] that contains about 25 h of Russian speech recordings pronounced by 200 speakers; unfortunately, some recordings contained a lot of noises, hesitations, self-repairs, etc., therefore some recordings were excluded;
- M-AILABS[2] that mostly contains recordings of audiobooks; Russian part of the corpus contains 46 h of speech recordings of three speaker (two men and one woman).

As a result we had about 60 h of speech data. This speech dataset was splitted into validation and trains parts with sizes of 5% and 95%.

Our test speech corpus consists of 500 phrases pronounced by 5 speakers. The phrases were taken from online newspaper which was not used for LM training.

5 Experiments

At the decoding stage we used beam search pruning method similar to the approach described in [26]. Our method is described in detail in [27], in general terms, we filter beam search output with some condition to remove too bad hypotheses.

[1] http://www.voxforge.org/.

[2] https://www.caito.de/2019/01/the-m-ailabs-speech-dataset/.

Moreover, during decoding we used gumbel-softmax function instead of softmax. The standard decoding algorithm uses softmax function building probability distribution for estimating the character probability on each iteration. However this distribution is rather strict that can influence on the recognition result. Therefore we replaced softmax function by gumbel-sofmax [28]:

$$gumbel\_softmax_i(z) = \frac{e^{\frac{z_i+g_i}{T}}}{\sum_{k=1}^{K} e^{\frac{z_i+g_k}{T}}},$$

where T is smoothing coefficient, g_i is values from Gumbel probability destribution. With T increasing, the probability distribution of the characters become more uniform that does not allow the decoding algorithm to give too confident decisions regarding the characters. During the preliminary experiments we chose $T = 3$.

Using our baseline system we obtained CER = 10.8% and WER = 29.1%. The results obtained with usage of coverage-based and 2D location-aware attention as well as joint usage of coverage and 2D location-aware attentions are presented in Table 1.

Table 1. Experimental results on speech recognition using different types of attention

Model	CER, %	WER, %
Baseline	10.8	29.1
Coverage-based attention	11.0	29.5
2D location-aware attention	10.6	**27.9**
Joint coverage-based and 2D location-aware attention	10.9	29.8

As we can see from the Table, the best result was obtained with 2D location-aware attention (CER = 10.8%, WER = 27.9%). The usage of coverage-based attention, as well as joint coverage and 2D location-aware attention does not results in improvement of speech recognition results.

6 Conclusions and Future Work

In the paper, we have investigated two types of attention mechanisms for Russian end-to-end speech recognition system: coverage-based attention and 2D location-aware attention. Coverage-based attention unfortunately does not results in improving speech recognition result. The usage of 2D location-aware attention allowed us to achieve 4% relative reduction of WER. In further research we are going to research another architectures of neural network for Russian end-to-end speech recognition, for example, Transformer.

Acknowledgements. This research was supported by the Russian Foundation for Basic Research (project No. 18-07-01216).

References

1. Markovnikov, M., Kipyatkova, I.: An analytic survey of end-to-end speech recognition systems. SPIIRAS Proc. **58**, 77–110 (2018)
2. Soltau, H., Liao, H., Sak, H.: Neural speech recognizer: Acoustic-to-word LSTM model for large vocabulary speech recognition (2016). arXiv preprint arXiv:1610.09975 https://arxiv.org/abs/1610.09975
3. Liao, H., McDermott, E., Senior, A.: Large scale deep neural network acoustic modeling with semi-supervised training data for YouTube video transcription. In: Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU), pp. 368–373 (2013)
4. Google preferred lineup explorer – YouTube. https://www.youtube.com/yt/lineups/. Accessed 17 Feb 2018
5. Tüske, Z., Audhkhasi, K., Saon, G.: Advancing sequence-to-sequence based speech recognition. In: INTERSPEECH-2019, pp. 3780–3784 (2019)
6. Kim, S., Hori, T., Watanabe, S.: Joint CTC-attention based end-to-end speech recognition using multi-task learning. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2017), pp. 4835–4839 (2017)
7. Salazar, J., Kirchhoff, K., Huang, Z.: Self-attention networks for connectionist temporal classification in speech recognition. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP-2019), pp. 7115–7119 (2019)
8. Chiu, C.C., Raffel, C.: Monotonic chunkwise attention (2017). arXiv preprint arXiv:1712.05382
9. Miao, H., et al.: Online hybrid CTC/Attention architecture for end-to-end speech recognition. In: INTERSPEECH-2019, pp. 2623–2627 (2019)
10. Watanabe, S., et al.: ESPnet: end-to-end speech processing toolkit. In: INTERSPEECH-2018, pp. 2207–2211 (2018)
11. Srivastava, N., et al.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**(1), 1929–1958 (2014)
12. Szegedy, C., et al.: Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2818–2826 (2016)
13. Chorowski, J.K., et al.: Attention-based models for speech recognition. In: Advances in Neural Information Processing Systems, pp. 577–585 (2015)
14. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition(2014). arXiv preprint arXiv:1409.1556
15. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift (2015). CoRR abs/1502.03167 http://arxiv.org/abs/1502.03167
16. Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: Proceedings of the 14th International Conference on Artificial Intelligence and Statistics, pp. 315–323 (2011)
17. Tu, Z., et al.: Modeling coverage for neural machine translation (2016). arXiv preprint arXiv:1601.04811
18. See, A., Liu, P.J., Manning, C.D.: Get to the point: Summarization with pointer-generator networks (2017). arXiv preprint arXiv:1704.04368
19. Kipyatkova, I.: Experimenting with hybrid TDNN/HMM acoustic models for Russian speech recognition. In: Karpov, A., Potapova, R., Mporas, I. (eds.) SPECOM 2017. LNCS (LNAI), vol. 10458, pp. 362–369. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66429-3_35

20. Kipyatkova, I., Karpov, A.: Class-based LSTM Russian language model with linguistic information. In: Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pp. 2470–2474 (2020)
21. Jokisch, O., Wagner, A., Sabo, R., Jaeckel, R., Cylwik, N., Rusko, M., Ronzhin, A., Hoffmann, R.: Multilingual speech data collection for the assessment of pronunciation and prosody in a language learning system. Proceedings of SPECOM **2009**, 515–520 (2009)
22. State Standard P 50840–95. Speech Transmission by Communication Paths. Evaluation Methods of Quality, Intelligibility and Recognizability, p. 230. Standartov Publication, Moscow (1996). (in Russian)
23. Stepanova, S.B.: Phonetic features of Russian speech: realization and transcription, Ph.D. thesis (1988). (in Russian)
24. Verkhodanova, V., Ronzhin, A., Kipyatkova, I., Ivanko, D., Karpov, A., Železný, M.: HAVRUS corpus: high-speed recordings of audio-visual Russian speech. In: Ronzhin, A., Potapova, R., Németh, G. (eds.) SPECOM 2016. LNCS (LNAI), vol. 9811, pp. 338–345. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-43958-7_40
25. Karpov, A.A., Ronzhin, A.L.: Information enquiry kiosk with multimodal user interface. Pattern Recogn. Image Anal. **19**(3), 546–558 (2009)
26. Freitag, M., Al-Onaizan, Y.: Beam search strategies for neural machine translation (2017). arXiv preprint arXiv:1702.01806
27. Markovnikov, N., Kipyatkova, I.: Investigating joint CTC-attention models for end-to-end Russian speech recognition. In: Salah, A.A., Karpov, A., Potapova, R. (eds.) SPECOM 2019. LNCS (LNAI), vol. 11658, pp. 337–347. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-26061-3_35
28. Jang, E., Gu, S., Poole, B.: Categorical reparameterization with gumbel-softmax (2016). arXiv preprint arXiv:1611.01144