An End-to-End Speech Recognition for the Nepali Language

Sunil Regmi[1] **Bal Krishna Bal**[2]

Information and Language Processing Research Lab
Department of Computer Science and Engineering
Kathmandu University
Dhulikhel, Kavre, Nepal
[1]sunilregmi233@gmail.com
[2]bal@ku.edu.np

Abstract

Most Nepali speech recognition systems have followed the traditional methods of Speech recognition which involve separately trained acoustic, pronunciation, and language model components. Developing such components from scratch requires domain expertise and is also time-consuming. Similarly, adoption of attention-based approaches, which is the latest technology trend is also not that popular in Nepali speech recognition. The only method found to be applied is the CTC method. In this work, we present an End-to-End ASR approach, which uses a joint CTC- attention-based encoder-decoder and a Recurrent Neural Network based language modeling through which we not only eliminate the need of creating a pronunciation lexicon from scratch but also take the fullest advantage of the state-of-the-art deep learning technologies. We use the ESPnet toolkit which uses Kaldi Style of data preparation framework. The speech and transcription data used for this research is freely available on the Open Speech and Language Resources (OpenSLR) website. We have obtained a Character Error Rate (CER) of 10.3% on 159k transcribed speech (159k utterances taken from OpenSLR.

1 Introduction

There has been increasing use of Automatic Speech Recognition (ASR) technologies in many application domains like Hospital Information Systems to IoT devices, industrial robotics, forensic, defense and aviation, etc. to name a few. According to Jelinek (1976), a conventional speech recognition system consists of several modules that comprise the acoustic, lexical, and language models supported by a probabilistic model for noisy channels. To build an acoustic model, we first need to build a Hidden Markov Model (HMM) and a Gaussian Mixture Model (GMM). In addition, the system requires linguistic knowledge based on a lexical model, usually based on a handmade pronunciation dictionary that does not have an explicit word limit. The lexical model requires language-specific tokenization modules for language modeling to develop ASR for new languages. Finally, decoding must be done with the synergistic action of all the modules, resulting in a complex decoding process (Hori, et al., 2017). Nevertheless, today's systems rely heavily on the End-to-End architectures that have emerged around traditional techniques(Geofferey et al., 2012). Unlike in the traditional methods like Hidden Markov Model (HMM) based model, the end-to-end approach addresses a single network architecture within a deep learning framework that directly maps language features to words or characters (Hori et al., 2017). There are two main architectures for end-to-end ASR. Connectionist Temporal Classification (CTC) allows the training of acoustic models without frame-level alignment between transcripts and acoustic frames, while attention models perform alignment between acoustic frames and identifiers (Wang & Li, 2019). Kim et al (2017) present a CTC/Attention-based collaborative end-to-end ASR that uses the CTC objective from loss function during the attention model training. The joint CTC-Attention-based encoder-decoder utilizes both the benefits of CTC and attention during training. The CTC predictions are also used in the decoding process. CTC can be interpreted as just a loss function used for training neural networks such as cross-entropy models. It is used in a difficult situation where the availability of aligned data is an issue, like in ASR. The capability to model temporal correlations with appropriate context information can be found in Convolutional Neural Network (CNN) (Ying et al., 2016; Zhang et al., 2017). Also, the language model (LM) integration is widely used for the HMM-based systems, something that is still applicable and effective for End-to-End ASR as

Proceedings of the 18th International Conference on Natural Language Processing, pages 180–185
Silchar, India. December 16 - 19, 2021.

well (Mikolov et al., 2010). The attention-based approach is used with CTC and Recurrent Neural Network-based Language Model (RNN-LM) as a joint decoder and a CNN-based shared encoder to achieve a state-of-the-art accuracy (Kim et al., 2017).

Most of the works on Speech Recognition for Nepali are based on the traditional methods. Nepali is written in the Devanagari script, which is essentially phonetic, so its pronunciation is very similar to how it is written. In Nepali, there are a total of 11 vowels and 33 consonants (Bal, 2004).

We use an End-to-End Speech Recognition Architecture that is based on a joint CTC/Attention-based encoder-decoder with a Recurrent Neural Network-based language modeling developed by Hori et al. (2017). We use the ESPnet framework (Watanabe et al., 2018; Hayashi et al., 2020; Inaguma, S., 2020; Chenda et al., 2021), which is an End-to-End speech processing toolkit. It sits on top of the Kaldi speech recognition toolkit (Povey et al., 2011) and the deep learning frameworks based on PyTorch (Paszke et al., 2019).

This paper is divided into five sections. Section I gives an introduction about the problem, section II gives an overview of the related works in Nepali speech recognition, section III describes the methodology for developing RNN-LM and CTC/attention-based models, section IV discusses the model and presents the experimental results, and finally section V presents the conclusion and future works.

2 Related Works

Some prior works on Nepali ASR in the character, word, and sentence level have been conducted. Google, for example, provides cloud-based Speech Recognition for more than 80 languages including Nepali. Unfortunately, there is not any publicly available documentation on the underlying methods and techniques used for the Project. Prajapati et al (2008) analyzed existing models for speech recognition and upon finding the shortcomings of Dynamic Time Wrapping (DTW's) approach, they proposed a new model called the Ear Model. They report to have obtained better accuracy than existing methods, but only for single alphabets. The classical, most commonly used model for Speech Recognition is Hidden Markov's model which is used by Ssarma et al (2017) where the authors have obtained a fairly good accuracy of 75% for isolated words. Similarly,, Regmi et al. (2019) used the RNN-CTC model and obtained a CER of nearly about 34% accuracy making use of the language model. Bhatta et al. (2020) proposed a model comprising CNN, GRU, and CTC networks. The dataset used by the authors is provided by Open Speech and Language Resources. Their build model recognizes speech with the WER of 11%. Baral & Shrestha (2020) present a comparative study of popular speech recognition methods for the Nepali language where they built a phonetic dictionary from scratch and presented the findings on 50K vocabulary for DNN and GMM based techniques with speaker adaptation. The experiments were carried out in the Kaldi toolkit. The lowest WER for different GMM- HMM models were 29.45% and similarly, the lowest WER for different DNN models using DNN- TDNN-LSTM was 11.55%.

Our study reveals that the researches till now, in Nepali speech recognition, have used the traditional methods except for Bhatta et al (2020) and Regmi et al (2019) who have used the CTC based End-to-End approach for small vocabulary tasks. The main difference between the CTC and attention method is that the conditional independence assumptions are not employed in the attention-based methods whereas CTC requires several conditional independence assumptions to get the probability of a label sequence. From this perspective, the attention mechanism, on the other hand, is surprisingly flexible as it allows extremely non-sequential alignments. However, for speech recognition tasks, the alignment is usually monotonic. As a regularization method, Kim et al (2017) uses a CTC objective from loss function to an attention-based encoder network which encourages the alignments' monotonicity. This method improves the accuracy of ASR compared to CTC or attention-based methods alone.

3 Architecture for Joint CTC/Attention Model

There are a series of steps involved in a speech recognition system requiring different components like data collection and preparation, data pre-processing, feature extraction, model building, training and testing, and decoding. For a Machine Learning System, clean and processed data are a

basic prerequisite and thus represents a very crucial resource.

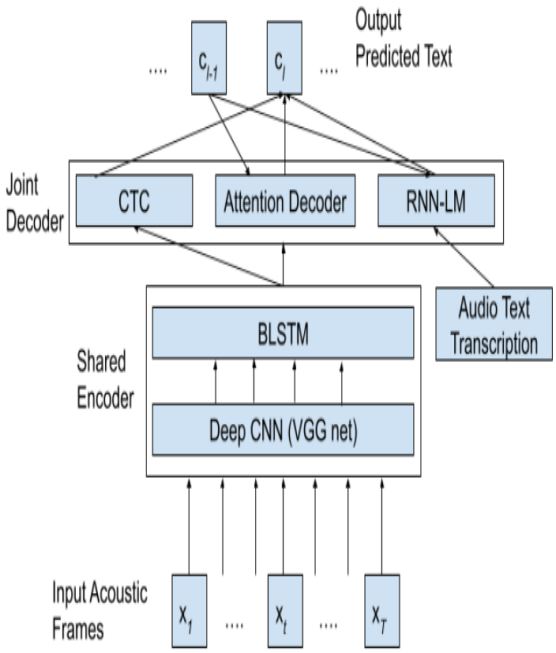

Figure 1: Joint CTC-attention ASR Architecture with Deep CNN and RNN-LM. Source: (Hori et al., 2017)

Audio data should basically be collected and recorded in a controlled environment, without background noise such as coughing, wheezing, and throat cleansing, or environmental sounds such as beeps, phone rings, and door slams. In the Nepali Language, there are some speech corpora provided by the Open Speech and Language Resource for research purposes. Fortunately, the speech data that we use from the aforementioned source is noise-free and thus ideal for our research.

In this research, a CTC-attention based joint encoder-decoder is used that takes advantage of both CTC and attention during training. For the language modeling part, Recurrent Neural Network is used. The RNN-LM network shows good improvement over the hybrid/HMM model and is merged in parallel with the attention decoder, which can be trained individually or jointly. The training is done with character sequences without word-level knowledge. Figure 1 shows the architecture of a CTC-attention based joint encoder-decoder (Hori et al., 2017). This approach's performance is superior to the different state-of-the-art hybrid ASR systems.

In the given architecture, at first analog electrical signals are converted to digital signals. This is done in the feature extraction part where Mel Frequency Cepstral Coefficients (MFCC) are used to extract audio features to distinguish different sounds or letters of a language. After that, the features are passed to a Deep CNN-based encoder that uses the Visual Geometry Group (VGG) network. The input for the CNN network is the Mel-scale feature from the raw speech features. The initial layers of the VGG net architecture (K. Simonyan and A. Zisserman, 2014) with 6-layer CNN architecture are used followed by BLSTM layers in the encoder network (Hori et al., 2017). The output is used by the CTC and the Attention Decoder as a shared encoder. The joint CTC-Attention-based encoder-decoder utilizes both the benefits of CTC and attention during training. The CTC predictions are also used in the decoding process. The model outputs the L-length character sequence as a set of individual characters U.

U = {'अ', 'आ', 'इ', 'ई', 'उ', 'ऊ', 'ए', 'ऐ', 'ओ', 'औ', 'क', 'ख', 'ग', 'घ', 'ङ', 'च', 'छ', 'ज', 'झ', 'ञ', 'ट', 'ठ', 'ड', 'ढ', 'ण', 'त', 'थ', 'द', 'ध', 'न', 'प', 'फ', 'ब', 'भ', 'म', 'य', 'र', 'ल', 'व', 'श', 'ष', 'स', 'ह', 'ँ', 'ं', 'ः', '्', 'ा', 'ि', 'ी', 'ू', 'ृ', 'े', 'ै', 'ो', 'ौ', '्', 'ॐ', 'ऋ', '।', '०', '१', '२', '३', '४', '५', '६', '७', '८', '९', ' ’ }

Out of 129 Unicode characters, 71 characters are used which are extracted from the text transcription of the speech data used in this research. The character set is indexed from 0 to 70. The input of CTC is the last hidden layer of the Bidirectional Long-Short Term Memory (BLSTM). Joint decoding is performed with a one-pass beam search algorithm that combines both attention-based scores and CTC scores to further eliminate irregular alignments. Use one tuning parameter to linearly interpolate both objectives, the multitasking learning (MTL) rate, which is typically set to 0.3. For the language modeling part, Recurrent Neural Network is used. The RNN-LM probabilities of output label prediction are used in conjunction with the decoder network because they assign a probability to each clause in such a way that the more probable strings (in a sense) get a higher probability and we tend to choose one. Similarly, an additional rescoring step is not needed if we combine the LM probabilities while decoding (Hori et al., 2017). Thus, this model can be viewed as a single gigantic neural network, even though its parts are pretrained independently.

4 Experiments and Results

The audio data and the text transcription are collected from the openslr.org website which contains transcribed audio data for Nepali Language Kjartansson et al., Sodimana et al (2018). The dataset contains Nepali Speech Data containing ~157K utterances and a text file that contains the utterance id and the text of the respective speech utterances. All speech utterance sums up to speech of duration of 9,278 minutes and 11 seconds. This corpus contains 86,062 unique utterances. Nepali Speech corpus from Open speech and language resource named "Multi-speaker TTS data for Nepali (ne-NP)" (SLR43) has been also used to the train models. It contains about 2064 long sentences-based utterance spoken by 18 different female speakers. All speech utterances sum up to speech of duration of 167 minutes and 45 seconds. This corpus contains 2064 unique utterances. Altogether, ~159k utterances are used.

For conducting the experiments, RTX 2060 GPU is used for training the model and Ryzen 9 CPU with 16 cores is used for the decoding process. The dataset was split in the ratio of 8:2 for the train and test dataset and from the remaining 80% of the training data, again it was split for train and validation set in a ratio of 8:2. The CNN BLSTM encoder uses 80 mel-scale filter banks with the delta and delta-delta features as input features. A 4-layer BLSTM with 320 units per layer and direction is used. To extract the convolutional features, 10 centered convolutional filters with a width of 100 were used. A 1-layer LSTM with 320 cells units are used as a decoder network. A single-layer LSTMs for RNN-LMs are individually trained using transcription, combined with a decoder network, and optionally retrained in collaboration with encoders, CTC networks, and decoders. The training is done for 20 epochs with the patience of value 2. Here, no extra text data were used but the use of additional un-transcribed data can further improve the results. The AdaDelta algorithm with gradient clipping was used for the optimization (Hori et al., 2017). The beam width and CTC-weight were set to 20 and 0.3 in decoding process. The CER is used as the evaluation metric which is defined as the sum of characters that is substituted, inserted, and deleted in particular sentence divided by the entire number of characters within the dataset.

	Sub	Del	Ins	Total	Total/Total Characters	% CER
Validation	28317	16652	8757	53726	53726/458271	11.7
Test	31117	18627	9739	59483	59483/576870	10.3

Table 1: CER using joint CTC/Attention model for mixed SLR54 and SLR43 datasets (~159K utterances)

Ground Truth	Prediction
कानपुर भारतका सर्वाधिक	कानपुर भारतका सर्वाधिक
पत्ता लगाउनको लागि	पत्ता लगाउनको लागि
२ हजार ३ गरी	२००० ३ गरी
छोराहरूको भविष्य भनी	चोराहरूको भविष्य बने
सूर्य चन्द्रमा र पृथ्वीका माझ उत्पन्न हुने दुईवटा छाँयालाई राहु र केतु भनिन्छ।	सूर्य चन्द्रमा र पृथ्वीका माझ उत्पन्न हुने देवटा छयालाई राहु र केतु भनिन्छ।
खसी बाख्रालाई घाँस	खसी बाख्रालाई गाँस

Table 2: Ground truth and prediction for sample test data

Table 1 provides the CER for the mixed SLR54 and SLR43 datasets. This model recognizes Nepali speech input data with 10.3% CER. Similarly, Table 2 provides a sample of ground truth and prediction for the sample test data. There are cases where the joint CTC/Attention model with 10.3 % CER in test data incorrectly predicts the utterance - २ हजार ३ गरी as २००० ३ गरी. Here, the 2 हजार is predicted as २००० which is not necessarily wrong but this can affect the accuracy of the model. We also noted that the model sometimes gets confused with छ and recognizes as च, भ as ब, and घ as ग. In some cases, the characters ँ is not recognized by the model. The obtained CER can be further improved by fine-tuning the parameters i.e, ctc-weight and multi-task learning rate.

5 Conclusion and Future Works

We experimented with the Nepali speech datasets using the End-to-End Automatic Speech Recognition framework called ESPnet. The framework uses CTC, attention, their joint form with RNN based LM, transformer, conformer, and transducer-based models. In addition, we experimented with an advanced architecture that

includes common decoding, a deep CNN encoder, and an RNN-LM network proposed by Hori et al (2017). The proposed approach eradicates the need of components which are essential in any conventional ASR model. We have achieved a speech recognition of CER – 11.7% and 10.3% for Nepali Language, respectively for the validation and the test data using the End-to-End model. Also, using substantial amounts of unlabeled data in conjunction with a pre-trained RNN-LM, this model can be improved further.

We recommend future works on End-to-End speech recognition for Nepali to be focused on employing the transformer, conformer, and transducer-based models. In recent studies, the conformer (Convolution-augmented Transformer) (Gulati et al., 2020) network showed a significant improvement and has outperformed the performance of Transformer and CNN-based models with different ASR standard datasets (Guo et al., 2021). It is also recommended to use multiple GPUs for fast training and decoding time. Additionally, the toolkit "ESPRESSO" suggested by Yiming et al. (2019) can be used to gain 4x faster accuracy in decoding instead of the ESPnet.

References

Bal, Bal Krishna. (2004). *Structure of Nepali Grammar.* PAN Localization, Working Papers 2004-2007, 332– 396.

Baral, Elina & Shrestha, Sagar (2020). Large Vocabulary Continuous Speech Recognition for Nepali Language. 8(4), 68–73. https://doi.org/10.18178/ijsps.8.4.68-73

Bhatta, R. (2020). Nepali Speech Recognition Using CNN, GRU and CTC. In Proceedings of the 32nd Conference on Computational Linguistics and Speech Processing (ROCLING 2020) (pp. 238–246). The Association for Computational Linguistics and Chinese Language Processing (ACLCLP).

Chenda Li, Jing Shi, Wangyou Zhang, Aswin Shanmugam Subramanian, Xuankai Chang, Naoyuki Kamo, Moto Hira, Tomoki Hayashi, Christoph Boeddeker, Zhuo Chen, & Shinji Watanabe (2021). ESPnet-SE: End-to-End Speech Enhancement and Separation Toolkit Designed for ASR Integration. In Proceedings of IEEE Spoken Language Technology Workshop (SLT) (pp. 785–792).

Frederick Jelinek. 1976. Continuous speech recognition by statistical methods. Proceedings of the IEEE 64(4):532–556

Geoffrey Hinton, Li Deng, Dong Yu, George E Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Tara N Sainath, et al. 2012. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal Processing Magazine 29(6):82–97.

Gulati, A., Qin, J., Chiu, C. C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., & Pang, R. (2020). *Conformer: Convolution-augmented transformer for speech recognition.* Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, 2020-Octob, 5036–5040. https://doi.org/10.21437/Interspeech.2020-3015

Guo, P., Boyer, F., Chang, X., Hayashi, T., Higuchi, Y., Inaguma, H., Kamo, N., Li, C., Garcia-Romero, D., Shi, J., Shi, J., Watanabe, S., Wei, K., Zhang, W., & Zhang, Y. (2021). *Recent Developments on Espnet Toolkit Boosted By Conformer*. 5874–5878. https://doi.org/10.1109/icassp39728.2021.9414858

Hayashi, T., Yamamoto, R., Inoue, K., Yoshimura, T., Watanabe, S., Toda, T., Takeda, K., Zhang, Y., & Tan, X. (2020). Espnet-TTS: Unified, reproducible, and integratable open source end-to-end text-to-speech toolkit. In Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 7654–7658).

Hori, T., Cho, J., & Watanabe, S. (2019). *End-to-End Speech Recognition with Word-Based Rnn Language Models.* 2018 IEEE Spoken Language Technology Workshop, SLT 2018 - Proceedings, 389–396. https://doi.org/10.1109/SLT.2018.8639693

Hori, T., Watanabe, S., & Hershey, J. R. (2017). *Joint CTC/attention decoding for End-to-End speech recognition.* ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers), 1, 518–529. https://doi.org/10.18653/v1/P17-1048

Hori, T., Watanabe, S., Zhang, Y., & Chan, W. (2017). *Advances in joint CTC-attention based end-to-end speech recognition with a deep CNN encoder and RNN-LM.* Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, 2017- Augus, 949–953. https://doi.org/10.21437/Interspeech.2017-1296

Inaguma, S. (2020). ESPnet-ST: All-in-One Speech Translation Toolkit. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations

(pp. 302–311). Association for Computational Linguistics.

Kim, S., Hori, T., & Watanabe, S. (2017). Joint CTC-attention based End-to-End speech recognition using multi-task learning. ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, 4835–4839. https://doi.org/10.1109/ICASSP.2017.7953075

K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

Kjartansson, O., Sarin, S., Pipatsrisawat, K., Jansche, M., & Ha, L. (2018). Crowd-Sourced Speech Corpora for Javanese, Sundanese, Sinhala, Nepali, and Bangladeshi Bengali. August, 52–55. https://doi.org/10.21437/sltu.2018-11

Mikolov, Tomas & Karafiát, Martin & Burget, Lukas & Cernocký, Jan & Khudanpur, Sanjeev. (2010). Recurrent neural network based language model. Proceedings of the 11th Annual Conference of the International Speech Communication Association, INTERSPEECH 2010. 2. 1045-1048.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., … Chintala, S. (2019). *PyTorch: An imperative style, high-performance deep learning library.* ArXiv, NeurIPS.

Povey, D., Boulianne, G., Burget, L., Motlicek, P., & Schwarz, P. (2011). The Kaldi Speech Recognition. IEEE 2011 Workshop on Automatic Speech Recognition and Understanding, January. http://kaldi.sf.net/

Prajapati, C., Nyoupane, J., Shrestha, J. D., & Jha, S. (2008). Acknowledgment. 24208.

Regmi, P., Dahal, A., & Joshi, B. (2019). Nepali Speech Recognition using RNN-CTC Model. International Journal of Computer Applications, 178(31), 1–6. https://doi.org/10.5120/ijca2019918401

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings, 1–14.

Sodimana, K., De Silva, P., Sarin, S., Kjartansson, O., Jansche, M., Pipatsrisawat, K., & Ha, L. (2018). A Step-by-Step Process for Building TTS Voices Using Open Source Data and Frameworks for Bangla, Javanese, Khmer, Nepali, Sinhala, and Sundanese. 66–70. https://doi.org/10.21437/sltu.2018-14

Ssarma, M. K., Gajurel, A., Pokhrel, A., & Joshi, B. (2017). HMM based isolated word Nepali speech recognition. Proceedings of 2017 International Conference on Machine Learning and Cybernetics, ICMLC 2017, 1, 71–76. https://doi.org/10.1109/ICMLC.2017.8107745

Wang, S., & Li, G. (2019). Overview of End-to-End speech recognition. Journal of Physics: Conference Series, 1187(5). https://doi.org/10.1088/1742-6596/1187/5/052068

Watanabe, S., Hori, T., Karita, S., Hayashi, T., Nishitoba, J., Unno, Y., Soplin, N. E. Y., Heymann, J., Wiesner, M., Chen, N., Renduchintala, A., & Ochiai, T. (2018). *ESPNet: End-to-End speech processing toolkit.* Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, 2018-Septe,2207–2211. https://doi.org/10.21437/Interspeech.2018-1456

Yiming Wang, Tongfei Chen, Hainan Xu, Shuoyang Ding, Hang Lv, Yiwen Shao, Nanyun Peng, Lei Xie, Shinji Watanabe, & Sanjeev Khudanpur (2019). Espresso: A Fast End-to-end Neural Speech Recognition Toolkit. In 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU).

Zhang, Ying, Pezeshki, M., Brakel, P., Zhang, S., Laurent, C., Bengio, Y., & Courville, A. (2016). *Towards End-to-End speech recognition with deep convolutional neural networks*. Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, 08-12-Sept,410–414. https://doi.org/10.21437/Interspeech.2016-1446

Zhang, Yu, Chan, W., & Jaitly, N. (2017). *Very deep convolutional networks for End-to-End speech recognition.* ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, 4845–4849. https://doi.org/10.1109/ICASSP.2017.7953077