A Model for Albanian Speech Recognition Using End-to-End Deep Learning Techniques

Amarildo Rista

Arbana Kadriu

South East European University,
Faculty of Contemporary Sciences and Technologies,
Tetovo, North Macedonia

Received: 4 April 2022 / Accepted: 25 June 2022 / Published: 1 July 2022

Doi: 10.56345/ijrdv9n301

Abstract

End-to-end Automatic Speech Recognition (ASR) system folds the acoustic model (AM), language model (LM), and pronunciation model (PM) into a single neural network. The joint optimization of all these components optimizes performance of the model. In this paper, we introduce a model for Albanian speech recognition (SR) using end-to-end deep learning techniques. The two main modules that build this model are: Residual Convolutional Neural Networks (ResCNN), which aims to learn the relevant features and Bidirectional Recurrent Neural Networks (BiRNN) aiming to leverage the learned ResCNN audio features. To train and evaluate the model, we have built a corpus for Albanian Speech Recognition (CASR), which contains 100 hours of audio data along with their transcripts. During the design of the corpus we took into account the attributes of the speaker such as: age, gender, and accent, speed of utterance and dialect, so that it is as heterogeneous as possible. The evaluation of the model is done through word error rate (WER) and character error rate (CER) metrics. It achieves 5% WER and 1% CER.

Keywords: Deep learning, Albanian language, End-to-end ASR, Speech Recognition, Corpus

1. Introduction

Nowadays, deep learning has achieved state-of-the-art performance in application of automatic speech recognition (ASR) systems and [Kadriu and Rista 2020] presents an overview of the related works. The success of deep learning in this domain started with the presentation of end-to-end (E2E) models. An E2E ASR system is an integrated model which folds the acoustic model (AM), language model (LM), and pronunciation model (PM) into a single neural network. They ensure much smaller model size, simple training, and low-training time, as well as improve the accuracy. Two of the most popular models today are Deep Speech [Hannun et al. 2014, Amodei et al. 2016], and Listen Attend Spell (LAS) [Chan et al. 2017]. Both models are RNNs based architectures with different approaches. Deep Speech uses the Connectionist Temporal Classification (CTC) loss function for training, while LAS uses attention mechanism. The most advanced architectures based on deep learning models include the convolutional neural networks (CNN) [Zbontar and LeCun 2016], recurrent neural network transducer (RNN-T) [Rao et al. 2013, Zhang et al. 2020] and recently transformers and their updated versions [Vaswani et al. 2017].

The main element that determines the success of a model is the corpus with which it is trained. During the design of the corpus we have considered the characteristics of the Albanian language related to phonetic, semantic, morphology and syntax as well as attributes of speaker such as: age, gender, accent, speed of utterance and dialect, which are very important in a corpus. An efficient ASR system should be able to identify all these attributes and to produce the output (text) corresponding to input (audio).

The aim of this paper is to design a model for SR of Albanian language which should be able to recognize general Albanian speech produced by different speakers with state-of-the-art performance as well as to design and create a corpus for Albanian language, suitable for training and testing different architectures of ASR.

The remainder of this paper is structured as follows: Section 2 provides a brief overview of the Albanian language; Section 3 presents a brief description of the main modules with which the proposed model is built; Section 4 focuses on research methodology; Section 5 presents experiment results; Section 6 lists the conclusions.

2. An Overview of Albanian Language

The Albanian language is one of the oldest languages belonging to the Indo-European group, and is spoken by more than 7 million people [Paçarizi 2008]. The phonological system of Albanian contains 7 vowels and 29 consonants. It has a developed system of grammatical forms, a binary inflection system, prominent and unprominent inflection, five cases and the system of three genders (masculine, feminine and neuter) [Orel 2000]. The structure of the sentence is composed of the subject, verb and predicate. The nominative system consists prominent and unprominent form as well as prominent and unprominent inflection. The nouns in Albanian language are inflected by gender (masculine, feminine and neuter) and number (singular and plural) [Kadriu 2013]. It has 5 declensions with 6 cases (nominative, accusative, genitive, dative, ablative and vocative). The verbal system has 6 types of moods (the indicative, admirative, subjunctive, conditional, optative and imperative) and tenses (3 simple and 5 complex constructions) [Kadriu 2010]. Albanian language has both an active and passive conjugation that can be in synthetic or analytical forms. The two main dialect groups of Albanian are Tosk and Geg. The main differences between them are phonetic, but there are also some grammatical differences, mainly of a morphological nature. Meanwhile, changes in syntax are almost negligible. Differences in the lexicon in supplementary forms are conditioned by the conditions and circumstances in which the inhabitants of these dialectal areas of the same language lived.

3. End-To-End Deep Learning Modules

This section describes the main modules of end-to-end models focusing mainly on the modules used to design the proposed model.

3.1 Recurrent Neural Networks (RNN)

An RNN is a type of neural network that uses previous experiences to predict the upcoming events, where the output of processing nodes feeds back again into the model [Mandic and Chambers 2001]. They are designed for capturing information from sequence or time series data. Figure 1 shows the diagram of a simple RNN. Each node of the network acts as a memory cell and it makes a decision considering the current input as well as the inputs received previously.

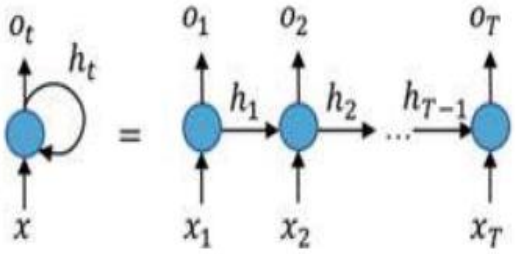

Figure 1: The diagram of a simple RNN [Mandic and Chambers 2001]

Let's assume training a RNN to the sequence $X = \{x_1, x_2, \ldots, x_T\}$, where $X_t \in R^N$ is an input vector at time t, with length T and the output O_t is expressed: $O_t = f(U_{x\_t} + W_{h\_(t-1)})$ (1)

Taking into account that the output is directly dependent on the previous result, the equation 1 can generally be written in this form. W, U $\in R^{NXN}$ and represent the weights of the matrices, f is a function which maps memory and input to an output, while h_t represents memory in time t. The RNN remembers every information through time to model a collection of records to be dependent on previous ones.

3.2 Bidirectional Recurrent Neural Network (BiRNN)

BiRNN considers both the past and the future context into predictions. It is composed of two RNN layers: forward layer and backward layer [Kamath et al. 2019]. Figure 2 shows the diagram of a BiRNN. Let's assume that the input sequence is $X = \{x_1, x_2, \ldots, x_T\}$, the forward sequence context is processed from left to right, t = {1, 2, . . . , T }, and the backward sequence context is processed from right to left t = {T, T − 1, . . . , 1}. The output of the BiRNN is a single output vector and it is expressed: $Y_t = f_0(h^f, h^r)$ (2)

The BiRNN provides better prediction accuracy of the model.

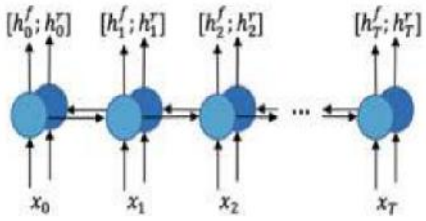

Figure 2: The diagram of a BiRNN [Kamath et al. 2019]

3.3 Convolutional Neural Network (CNN)

CNN is a type of deep neural network, which consists of some convolution layers that can be fully connected or pooled [O'Shea and Nash 2015]. Figure 3 shows the diagram of CNN.

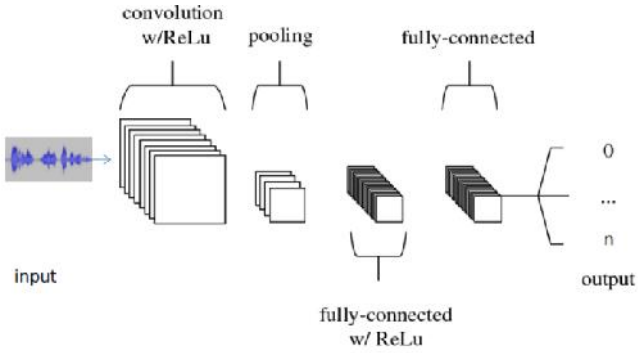

Figure 3: The diagram of CNN [O'Shea and Nash 2015]

The CNN has three main types of layers: convolution layers, pooling layers and fully-connected layers. Convolution layers are the main blocks used in CNNs, which perform an operation called convolution. The convolution is a mathematical operation that aims to create feature maps [Jin eat al. 2017]. If, X_t is the input function, the output is expressed:

$Y_t = (h_x)(t) = \int_{=\infty}^{\infty} h(\tau) \times (t - \tau)\, d(\tau)$ (3)

Where, h (t) is a function which is combined with input function x (t) to output an overlap between x (t) and the

reverse translated version of h (t).

The pooling layers are used to compress the dimensions of the feature maps, reducing the number of parameters as well as the operations performed in the network. Fully Connected Layers are the last layers in the network which are fed by the output of Convolutional Layer. Fully Connected Layers are the last layers in the network which are fed by the output of Convolutional Layer. They use an activation function called softmax [Liu et al. 2016], to classify inputs appropriately and outcome a probability matrix. The advantage of CNN is the capability of automatically detecting important features without any human supervision as well as weight sharing.

3.4 Residual Convolutional Neural Network (ResCNN)

Residual Neural Network (ResNet) is a version of neural networks that simplifies the training of very deep neural networks using skip connections [Vydana and Vuppala 2017]. ResNet consists of a number of residual blocks, where each block contains direct links between the lower layer outputs and the higher layer inputs [Wang et al. 2017]. In Figure 4 is shown the structure of the residual block.

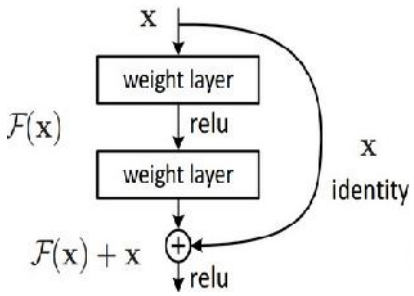

Figure 4: The structure of ResCNN block [Vydana and Vuppala 2017]

The Res block generally contains two convolutional layers. Each layer has the same structure, and the skip connection represents the identity mapping of x. Mathematically, it is expressed: $Y = F(x) + x$ (4)

Where, x and Y represent the input and output of the layers, and $F(x)$ is a function that can be combined with x and it feeds the next layer. ResNet accelerates the convergence in the training phase and improves the accuracy in depth of the network.

4. Methodology

This section defines the analytical methods to achieve the set objectives by providing clarification and rationalization of the model design, corpus development, tools and assessment criteria.

4.1 Model design

The proposed model is based on Recurrent Neural Network (RNN) and it is created and implemented in Pytorch [Paszke et al. 2019]. The two main modules on which this model is based are: Residual Convolutional Neural Networks (ResCNN) and Bidirectional Recurrent Neural Networks (BiRNN). Figure 5 shows the diagram of the proposed model. The audio waves extracted by the corpus are transformed into Mel Spectrograms as features to feed the ResCNN Layer. The ResCNN aims to learn the relevant features using skip connection. Skip connections simplify the training process by skipping some layers in the network. It accelerates training of very deep neural networks as well as reduces degradation problems and vanishing gradients [Ribeiro et al. 2020]. During the training the model will learn to align by itself, as a consequence of using the CTC loss function [Graves 2012, Amodei et al. 2016]. The output of ResCNN feeds the BiRNN, which aims to leverage the learned ResCNN audio features. BiRNN processes data in both directions, increasing the amount of data on the network consequently makes better prediction of the model.

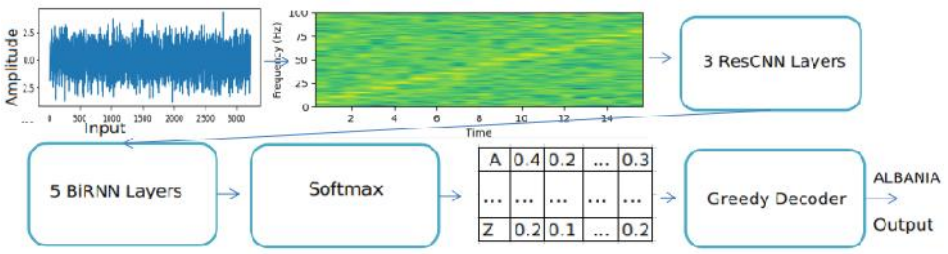

Figure 5: Proposed model

Next node is the Softmax module. It is a mathematical function that converts a vector of numbers into a vector of probabilities, where the probabilities of each value are proportional to the relative scale of each value in the vector. The model outputs a probability matrix that feeds the decoder. The last module is Greedy Decoder that enables the decoding process. It has the trained encoder and decoder models as attributes, and drives the process of encoding an input sentence and iteratively decoding an output [Battenberg 2017], as is shown in Figure 5. The evaluation of the ASR system is based on word error rate (WER) [Morris et al. 2004, Fiscus et al. 2006], loss training word and character error rate (CER) as metrics [Chung et al. 2017].

4.2 Corpus development

The corpus is the basic element that determines the accuracy of the model. To design the corpus, we have taken into account the features of the Albanian language as well as attributes such as: speaker and channel variability which includes attributes such as; phonetics, adverse environment conditions (clean, noisy); speaker attributes such as; age, gender, accents, speed of utterance, dialects; training process and voice recording device [Chan et al. 2016]. The final corpus contains 100 hours of transcribed audio data which will be available for free.

4.2.1 Source Data.

The CASR is based on 200 audio books which are freely available[1]. Each audio book represents a simply read speech style by the speakers. The topics are primarily concerned with biography, social and political sciences, psychology, religion, economics and business, history, philosophy and sociology. The audio books are in MP3 format, compressed in 32 bit-float and have a rate 16 kHz. In total we have selected 100 hours of audio without transcripts. The audio records are transcribed strictly verbatim, listening carefully several times to the speeches. To have a corpus as heterogeneous as possible, we have selected 30 speakers of which 20 are females and 10 males. Speakers are classified into 5 age groups (years old): 20-30, 30-40, 40-50, 50-60 and 60-70. Each age group includes 4 females and 2 males. They speak the standard language as well as both Tosk and Geg dialects of the Albanian language.

4.2.2 Audio and Text Preprocessing.

The audio recordings selected to build the corpus have a length ranging from 5 minutes up to 2 hours. To create audio files suitable for training ASR models the Audacity tool has been used [Audacity 2013]. Each audio record is exported to the Audacity tool where it is segmented in ranges 2 to 15 second length suitable for ASR systems. Next, all audio files are converted into flac format using a 16-bit linear PCM sample encoding (PCM_S16LE) sampled at 22.05 kHz. Each audio file called utterances has an average of 22 words. In total 37 758 utterances are created. The naming of audio files is done randomly with three independent groups of four-digit decimal numbers. Regarding transcripts, we have listened carefully to every audio file, and we have written the corresponding transcripts for every audio file. All transcripts are normalized by converting them into upper-case, removing the punctuation, and expanding common abbreviations [Sproat

[1] http://www.volumiaudiolibra.com/index.php?language=en

et al. 2001]. After the text is normalized, each text file is named with the same number as the audio files. In this way we have created the audio and text files of the CASR corpus.

4.2.3 Corpus description and organization.

After creating the audio files and text files as described above, we have organized them according to the objectives of this study. The final corpus contains 100 hours of transcribed audio, 37 758 utterances and it has a size of 12.3 GB. The CASR is divided in two parts called CORPUS_1 and CORPUS_2. Then, each corpus is divided also in two parts called CORPUS_train_1, CORPUS_test_1 and CORPUS_train_2, CORPUS_test_2. The separation of the corpus in this form helps all researchers who want to train and test their models, overcoming hardware limitations as well as helps us to analyze the impact of dataset size on the accuracy of the model. In Table 1, we have presented specifications of each part of Corpus according to the number of sentences, number of average words for sentences, number of total words, size, number of utterances and average length per utterance.

Table 1: Corresponding transcriptions information

	CORPUS train_1	CORPUS test_1	CORPUS train_2	CORPUS test_2
Number of sentences	30317	5491	1615	335
Avg.words per sentence	21.95	24	21.84	20.93
Total word	674456	131809	35274	7014
Size	76 h	19 h	4 h	1 h
Utterances	30317	5491	1615	335
Avg.duration per utterance	9.02 s	12.5 s	8.9 s	10.7 s

5. Experiments and Results

In this section, the experiments and their results are discussed. First, the model is trained and evaluated using the CASR corpus. The performance of model is measured on training set as well as on testing set in order that the model to ensure that it can generalize well to new data and was not over fitted. Second, it is analyzed how the size of the corpus affects the accuracy of the model. Since for the Albanian language we haven't research paper in this domain, in the last part, we have done an evaluation of CASR in comparison to LibriSpeech.

5.1 Evaluation of the model using CASR in terms of WER and CER on training set and testing set

The model was trained and tested using the CASR corpus which consists of 100 hours of audio recordings of which 80 hours are used for training and 20 hours for testing. The model in both cases gets trained for 50 epochs, because after 50 epochs very small insignificant changes are noticed. Figure 6 indicates the WER and CER results in percentage on training set.

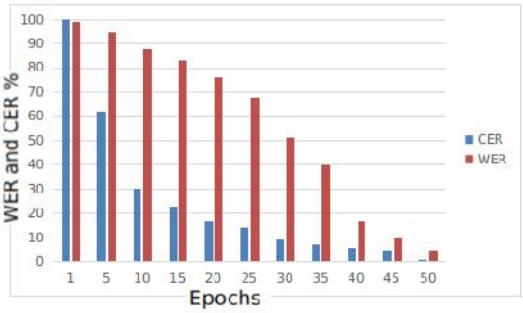

Figure 6: The performance on the training set in terms of WER and CER

As can be shown in Fig.6 the model achieves a very satisfactory WER and CER, where the WER goes to 5% and CER to 1%.

Figure 7 reports the WER and CER results in percentage on testing set.

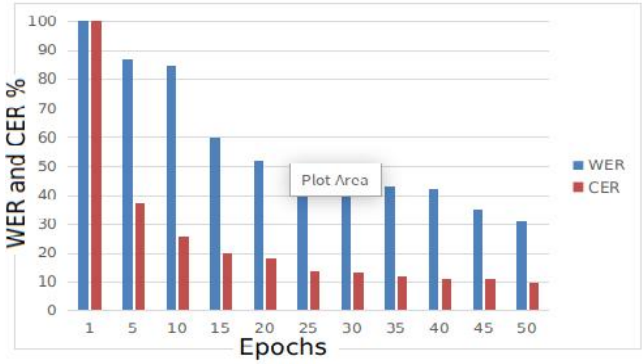

Figure 7: The performance on the testing set in terms of WER and CER

We show that CASR has yielded 32% WER and 9% CER on its own testing set. By comparing the WER and CER results obtained in the training set and testing set we note that have a difference of WER with 27% and CER with 8% among them. But, comparing these results with the results analyzed in the literature, we conclude that the proposed model generalizes well to new data and it was not over fitted.

5.2 Evaluation of corpus size in terms of WER and CER on training set

In this section it is analyzed how the size of the corpus affects the accuracy of the model. For this purpose, the model was trained and evaluated using the CORPUS_1 and CORPUS_2 as described in Table 1. The CORPUS_1 consists of 95 hours of audio recordings, of which 76 hours are used for training and 19 hours for testing. The CORPUS_2 consists of 5 hours of audio recordings, of which 4 hours are used for training and 1 hours for testing. Figure 8 indicates the WER results in percentage for both CORPUS_1 and CORPUS_2. When the model is trained with the CORPUS_1 achieves greater accuracy of the WER, it goes to 5%. While, when the model is trained with the CORPUS_2, the WER goes to 8%. Also, it is noted that when the model is trained with CORPUS_1, it converges faster in term of accuracy, which means that it achieves maximum accuracy in epoch 50, after this the curve becomes linear as is shown in Figure 8. Referring to the WER, we conclude that the size of the corpus affects the performance of the model, where a large corpus has greater accuracy of the WER and converges faster in term of accuracy.

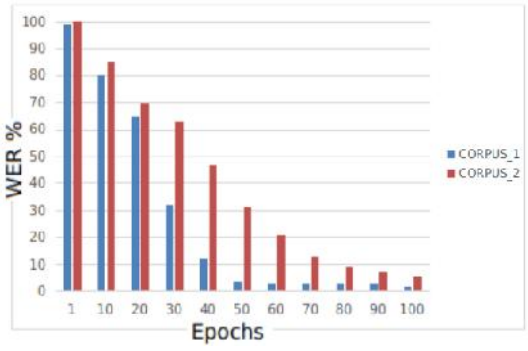

Figure 8: The performance on the training set in terms of WER

Figure 9 shows the CER results in percentage for both CORPUS_1 and CORPUS_2. When the model is trained with the CORPUS_1 achieves an accuracy of the CER to 1%. While, when the model is trained with the CORPUS_2, the CER goes to 3%. Also, it is noted that when the model is trained with CORPUS_1, it converges faster in term of accuracy, which means that it achieves maximum accuracy in epoch 60, after this the curve becomes linear as is shown in Figure 9. Referring to the CER, we conclude that the size of the corpus affects the performance of the model, where a large corpus has greater accuracy of CER and converges faster.

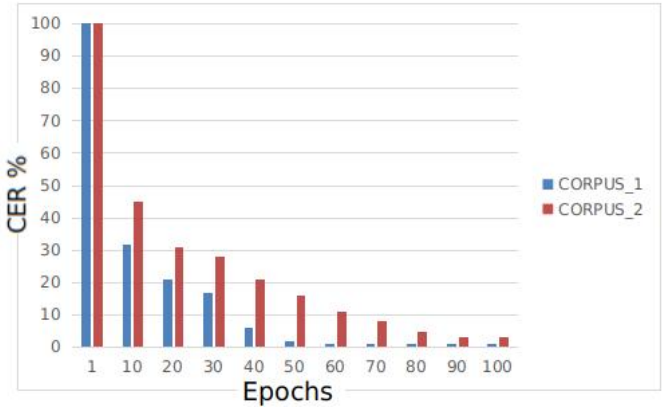

Figure 9: The performance on the training set in terms of CER

5.3 Evaluation of CASR in comparison to LibriSpeech

To address this issue, the model is trained first with CASR and then with LibriSpeech corpora. In both cases, we have created a training set and a testing set with a split ratio 80:20 randomly, specifically training set 80 hours and testing set 20 hours. Figure 10 shows the results of experiments for WER depending on the number of epochs for both CASR and LibriSpeech corpora.

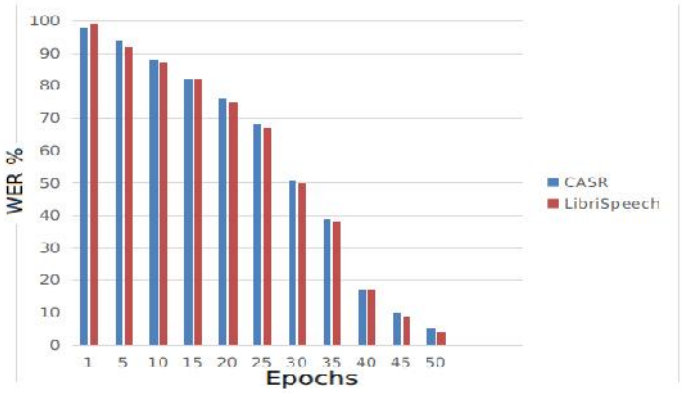

Figure 10: The performance on the training set in terms of WER

As can be shown from Fig.10, CASR has yielded 5% WER, while LibriSpeech has yielded 4 % WER on their training sets.

Figure 11 shows the results of experiments for CER depending on the number of epochs for both CASR and LibriSpeech corpora.

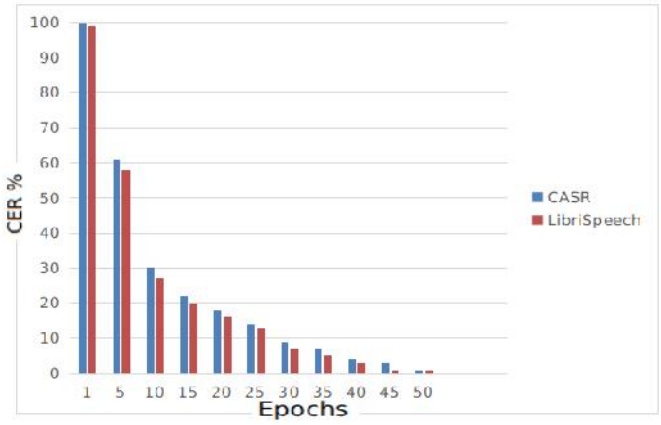

Figure 11: The performance on the training set in terms of CER

We note that CASR has yielded 1% CER, while LibriSpeech has yielded 1% CER on their training sets. Referring to the results presented in figures 10 and 11, we are happy to conclude that, the results obtained when the model is trained with CASR corpus are almost the same when it is trained with LibriSpeech. Where WER changes with 1% while CER is the same for both corpora.

6. Conclusions

This paper introduces a model for Albanian Speech Recognition based on end-to-end deep learning techniques, which is able to recognize general Albanian speech produced by different speakers. In addition, we have described the design and creation of a corpus for Albanian language suitable for training and evaluating various speech recognition architectures. The corpus consists of 100 hours of transcribed audio data, where the speakers have been selected from age group 20 to 70 years old, and use the standard language as well as both Tosk and Geg dialects of the Albanian language. During design of the corpus have been considered the features of the Albanian language as well as attributes such as: phonetics, environment conditions of audio recording; speaker attributes such as; age, gender, accent, speed of utterance and dialects. The proposed models focus mainly on two main neural network modules: Residual Convolutional Neural Networks (ResCNN) and Bidirectional Recurrent Neural Networks (BiRNN). The experimental results show that the proposed model achieves good performance. It achieves a very satisfactory WER, CER and Loss, where the WER goes to 5%, CER to 1% and Loss is negligible. Referring to experimental results, we conclude that the size of the corpus affects the performance of the model, where a large corpus has greater accuracy of WER and CER as well as converges faster. This study introduces a noble contribution to the field of natural language processing, with focus on Albanian language, which is expected to accelerate research within this domain.

References

Kadriu, A., & Rista, A. (2020). Automatic Speech Recognition: A Comprehensive Survey. Seeu Review, 15(2).

Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., & Ng, A. Y. (2014). Deep speech: Scaling up end-to-end speech recognition. ArXiv preprint arXiv: 1412.5567.

Amodei, D., Ananthanarayanan, S., Anubhai, R., Bai, J., Battenberg, E., Case, C., ... & Zhu, Z. (2016, June). Deep speech 2: End-to-end speech recognition in english and mandarin. In International conference on machine learning (pp. 173-182). PMLR.

Chan, W., Jaitly, N., Le, Q., & Vinyals, O. (2016, March). Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In 2016 IEEE international conference on acoustics, speech and signal processing (ICASSP) (pp. 4960-4964). IEEE.

Zbontar, J., & LeCun, Y. (2016). Stereo matching by training a convolutional neural network to compare image patches. J. Mach. Learn. Res., 17(1), 2287-2318.

Rao, K., Sak, H., & Prabhavalkar, R. (2017, December). Exploring architectures, data and units for streaming end-to-end speech recognition with rnn-transducer. In 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU) (pp. 193-199). IEEE.

Zhang, Q., Lu, H., Sak, H., Tripathi, A., McDermott, E., Koo, S., & Kumar, S. (2020, May). Transformer transducer: A streamable speech recognition model with transformer encoders and rnn-t loss. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 7829-7833). IEEE.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

Paçarizi, R. (2008). Albanian Language.

Orel, V. E. (2000). A concise historical grammar of the Albanian language: reconstruction of Proto-Albanian. Brill.

Kadriu, A. (2013, June). NLTK tagger for Albanian using iterative approach. In Proceedings of the ITI 2013 35th International Conference on Information Technology Interfaces (pp. 283-288). IEEE.

Kadriu, A. (2010). Modeling a two-level formalism for inflection of nouns and verbs in Albanian. Modeling Simulation and Optimization-Focus on Applications.

Mandic, D., & Chambers, J. (2001). Recurrent neural networks for prediction: learning algorithms, architectures and stability. Wiley.

Kamath, U., Liu, J., & Whitaker, J. (2019). Deep learning for NLP and speech recognition (Vol. 84). Cham: Springer. U., Liu, J., & Whitaker, J. (2019). (Vol. 84). Cham: Springer.

O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. arXiv preprint arXiv:1511.08458.

Jin, K. H., McCann, M. T., Froustey, E., & Unser, M. (2017). Deep convolutional neural network for inverse problems in imaging. IEEE Transactions on Image Processing, 26(9), 4509-4522.

Liu, W., Wen, Y., Yu, Z., & Yang, M. (2016, June). Large-margin softmax loss for convolutional neural networks. In ICML (Vol. 2, No. 3, p. 7).

Vydana, H. K., & Vuppala, A. K. (2017). Residual neural networks for speech recognition. In 2017 25th European Signal Processing Conference (EUSIPCO) (pp. 543-547). IEEE.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., & Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems, 32, 8026-8037.

Wang, Y., Deng, X., Pu, S., & Huang, Z. (2017). Residual convolutional CTC networks for automatic speech recognition. arXiv preprint arXiv:1702.07793.

Ribeiro, A. H., Tiels, K., Aguirre, L. A., & Schön, T. (2020, June). Beyond exploding and vanishing gradients: analysing RNN training using attractors and smoothness. In International Conference on Artificial Intelligence and Statistics (pp. 2370-2380). PMLR.

Graves, A. (2012). Supervised sequence labelling. In Supervised sequence labelling with recurrent neural networks (pp. 5-13). Springer, Berlin, Heidelberg.

Battenberg, E., Chen, J., Child, R., Coates, A., Li, Y. G. Y., Liu, H., & Zhu, Z. (2017, December). Exploring neural transducers for end-to-end speech recognition. In 2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU) (pp. 206-213). IEEE.

Morris, A. C., Maier, V., & Green, P. (2004). From WER and RIL to MER and WIL: improved evaluation measures for connected speech recognition. In Eighth International Conference on Spoken Language Processing.

Chung, J. S., Senior, A., Vinyals, O., & Zisserman, A. (2017, July). Lip reading sentences in the wild. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (pp. 3444-3453). IEEE.

Fiscus, J. G., Ajot, J., Radde, N., & Laprun, C. (2006, May). Multiple Dimension Levenshtein Edit Distance Calculations for Evaluating Automatic Speech Recognition Systems During Simultaneous Speech. In LREC (pp. 803-808).

Audacity, I. (2013). What is Audacity?

Sproat, R., Black, A. W., Chen, S., Kumar, S., Ostendorf, M., & Richards, C. (2001). Normalization of non-standard words. Computer speech & language, 15(3), 287-333.