EasyChair Preprint

№ 15235

Deep Learning Speech Recognition: Input Representation Perspective

Elsadig Babiker and Hanan Adlan

October 18, 2024

Deep Learning Speech Recognition: Input Representation Perspective

Elsadig A. M. Babiker[1,2]
ACM Professional Member
Dept. of Computer Science[1,2]
College of Computer Engineering and Science[1],
PSAU, KSA
Faculty of Mathematical Sciences[2],
UofK / Khartoum, Sudan

Hanan H. A. Adlan[2,3]
ACM Professional Member
Dept. of Computer Science[2,3]
Faculty of Computer and Information Sciences[3],
PNU, Riyadh, KSA
Faculty of Mathematical Sciences[2],
UofK / Khartoum, Sudan

ABSTRACT

Convolution neural network is becoming the state of the art models in many applications. With deep architectures, convolution neural network can learn speech patterns effectively. There remains the decision on using raw signals, spectrogram, or other input representation. In this paper Deep Convolution Architectures for Speech Recognition is designed, implemented, and developed. The architectures are implemented on raw data and on spectrogram representations. The architectures composed of two stages networks. Self extracting network and classification networks. First, the architecture uses the spectrogram approach to the feature extraction stage. Then classify the speech patterns into the appropriate class. The second architecture uses raw signal as input to the extraction stage. The two approaches use minimum preprocessing to the speech signal. The architectures recognize the speech patterns in the TI46 corpus. Extensive experiments were conducted to reach the best design in both approaches. Among the many convolution architectures we presented the best results. The architecture on raw signal produced better recognition rate, and achieves excellent performance over reported results.

KEYWORDS

Deep Learning, Convolution Neural Network, Pattern Recognition, Speech recognition.

1 Introduction

The work on developing modern models that automate speech recognition is still emerging. Many approaches are developed during the past years. Excellent results are reported. Based on this, research still comes up with promising architectures.

Speech recognition is considered one of the prominent fields. Reflected in content captioning, hands free interfaces in cars, home devices, mobile devices, voice recognition systems ...etc. With the emergence of Deep Learning, NNs are found to perform complex recognition tasks. Focuses on Enhance optimization, looking for powerful activation functions and enhance architectures, determine the myriad hyper parameters, preprocessing speech for deep neural networks to name a few [1].

Deep Neural Networks (DNNs) are playing a vital role in building systems in today's automation environments. The works on Big data, cloud computing, Internet of Things…etc consider DNNs the state of the art models for intelligent machine learning and artificial intelligence applications. Since 2006 Deep Learning (DL) becomes dominant research for its amazing performance [13]. One of the obstacles that face DNNs is the requirement of powerful processors and large memory requirements. The Graphic Processing Units (GPUs) improve the computation significantly, but still lack fast training sessions. On the other hand, memory requirement can be met by using cloud environments to support the high demands. Very rare SaaS supports free offerings.

Convolution Neural Networks (CNNs) is widely used in computer vision at their infancy. Currently emerging as powerful technologies in all recognition tasks. Previous research considered Hidden Markov Models (HMMs), Recurrent Neural Networks (RNNs) with various stochastic gradients decent that capture the temporal information in sequential data. Explore different types of activation functions with different architectures. In speech, convolution nets outperform HMM and other models. Ranging from representing the speech signal as Spectrograms to using raw signal [3, 10, 12]. Convolution and recurrent neural network is considered basic structure for speech recognition in many architectures [3]. Others are developed based on ensemble learning to improve robustness [4][5][6].

In this paper we design, develop, and implement Deep Convolution Neural Network Architectures for Speech Recognition (DCASR). DCASR1 uses spectrogram representation of the speech signal. DCASR2 accepts raw data. The TI46 Corpus is used in the training and testing of the architecture.

2 Speech recognition

Speech recognition approaches are Pattern Recognition, Acoustic-Phonetic, or Artificial Intelligence. Hybrid approaches incorporate combination of any of the previous. Examples are

models of acoustic-phonetics with pattern recognition. Traditional model of pattern recognition uses fixed/engineered features (or fixed kernel) plus a trainable classifier [14]. Mainstream modern pattern recognition is an unsupervised mid-level features together with a trainable classifier. With the emergence of deep learning, representations are hierarchical and trained.

Deep means it has more than one stage of non-linear feature transformation. This approach enables architectures to self extract appropriate representation from inputs. Many convolution architectures were developed over the past years [3].

3 Deep Convolution Architecture for Speech Recognition

The CNN is a deep architecture that composed of two stages Figure 1. Feature Extraction (FE) stage and a classification stage. The feature extraction stage is a deep neural network composed of cascade of convolution layers and pooling layers. The FE accepts speech signal. Generate feature maps followed by a pooling layer. The kernel filters the features on the map. Output from the extraction stage form the inputs to the classification stage. The classification stage is a fully connected network. Neurons in the input layer accept extracted features and classify them into appropriate classes. The classification network is a 5 layers fully connected network.

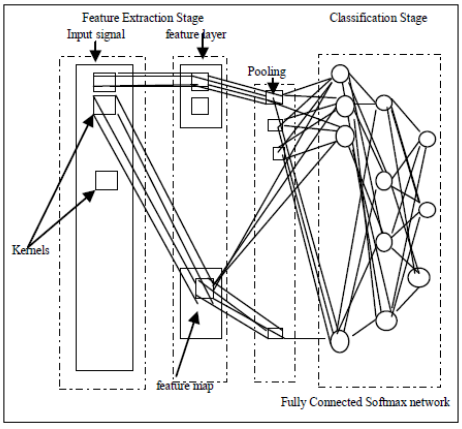

Figure 1: Portion of the DCASR Architecture

Figure 1 Displays Portion of the architecture. The feature extraction stage and the kernels, the feature layer with representative for the feature maps, and a pooling layer. The output from the feature extraction is then fed to the classification stage.

The cascade of the layers composed of four Convolutions, ReLU, and Pooling Layers, Figure 2.

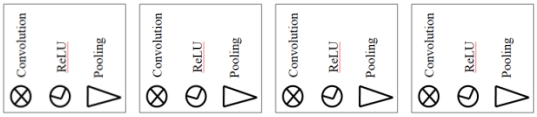

Figure 2: Feature Extractions in the DCASR Architecture

Figure 2 is the one of the cascades in the feature extraction stage. The classification stage is a fully connected network with softmax activation. In this stage each entry from the pooling layer is get a vote, a final output from the architecture recognize special class.

3.1 Speech Corpus

The corpus used in this research is the TI46. Speaker-Dependent Isolated word corpus, collected at Texas Instrument (TI) to provide researchers with training and testing sets. TI46 has 16 speakers, 8 males and 8 females. 26 utterances word for each speaker. 10 designated as training (enrollment) tokens and 16 as testing tokens.

The organization of the TI46 corpus composed of two directories, TI20 and TI Alpha. TI20 contains all utterances of the words for the ten digits plus some control words. TI20 Alpha contains utterances of the words for the English letters. Both TI20 and TI20 Alpha are divided into training and testing directories [7][8][14].

3.2 Preprocessing

The Original signal passes through number of steps

- Normalization: Speech signal is a 1-dimentional array. Each word is normalized in the range [-1, 1] following the pseudo code:

  signal = signal – mean (signal)
  signal = signal /max(abs(signal))

- Silence removal: silence removal performed to detect end points. The is performed following the pseudo code:

  points ⟵ 128 //length of frame
  advance ⟵ 256
  threshold 1e-2
  n ⟵ length of the signal
  high = n
  while high> points + 1 and mean(|frame|)< threshold
  high ⟵ high – advance
  low = 0
  while low < n – points and mean(|frame|)< threshold
  low ⟵ low + advance
  signal = signal [low : high]

- Resizing: resizing is to unify the signal length input to the neural network. Following the pseudo code:

```
q = 8000
n = length(signal)
If n<q
   append signal with q-n zeros
else
    signal = signal [0:q]
```

4 Results

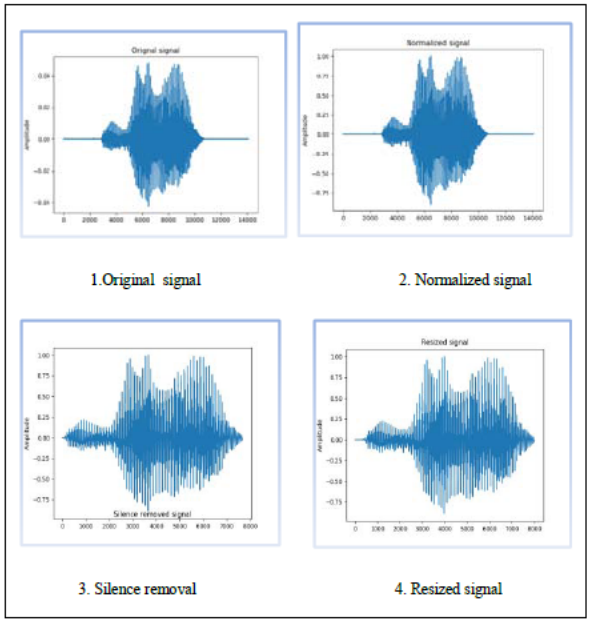

Figure 3: Preprocessing the speech signal

Figure 3 visualizes the speech signal at different preprocessing operations. In Figure 3, 1 displays the original signal, 2 shows the normalization effect on the original signal, the signal is normalized in the rang (-1, 1). 3 gives the signal after silence removal, each signal is divided into frames, we start with the first frame and find the mean of the absolute values then threshold with $1e^{-2}$, values below the threshold is considered silence and removed from the signal till the first frame with value greater than the threshold appears. Then perform the same starting at the end of the signal. 4 display the resizing of the signal to 8000.

First implementation of the architecture DCASR1 represents the signal by spectrogram. Spectrogram contains lots of information for the speech signal. The best recognition rate is found to be 98.7 %. Figure 4 displays the results for training and testing.

Epoch 1070/6000
- 13s - loss: 0.0322 - acc: 0.9951 - val_loss: 0.1387 - val_acc: 0.9831
Epoch 1071/6000
- 13s - loss: 0.0363 - acc: 0.9942 - val_loss: 0.1050 - val_acc: 0.9873
(7086, 20)
(1182, 20)
Train data Large CNN Error: 0.00%
Test data Large CNN Error: 1.27%
[0.014249244976379746, 1.0]
[0.1050018458351392, 0.9873096446700508]
[0, 0, 1, 2, 2, 3, 4, 4, 5, 6, 7, 7, 8, 9, 9, 10, 11, 11, 12, 13, 14, 14, 15, 16, 16]
[0, 0, 1, 2, 2, 3, 4, 4, 5, 6, 7, 7, 8, 9, 9, 10, 11, 11, 12, 13, 14, 14, 15, 16, 16]
100.0 % acc
98.73096446700508 % val_acc
98.90016904337152 % Max val_acc

Figure 4: DCASR1 Recognition Rates for Training and Testing

The model accuracy and model loss are visualized in figures 5 and 6 respectively.

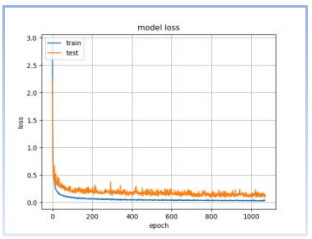

Figure 5: DCASR1 Accuracy Visualization for the Training and Testing

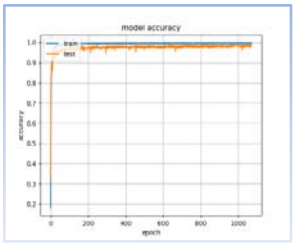

Figure 6: DCASR1 Loss Visualization for the Training and Testing Sets

Figure 5 displays the accuracy for the training and testing sets. While Figure 6 gives the loss for the training and testing. Epochs are shown on the horizontal axis, the accuracy and loss are on the vertical for the first and second figures respectively.

Using TensorFlow backend.
Dataset shape is (8268, 8000)
Dataset shape (8268, 8000)
Max trainY = 19 Max testY = 19
Min trainY = 0 Min testY = 0
Train X shape is (7086, 8000)
Test X shape is (1182, 8000)
Train X shape is (7086, 1, 100, 80)
Test X shape is (1182, 1, 100, 80)
Train Y shape is (7086, 20)
Test Y shape is (1182, 20)

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 1, 100, 320)	25920
max_pooling2d_1 (MaxPooling2	(None, 1, 100, 320)	0
conv2d_2 (Conv2D)	(None, 1, 100, 300)	96300
max_pooling2d_2 (MaxPooling2	(None, 1, 100, 300)	0
flatten_1 (Flatten)	(None, 26000)	0
dense_1 (Dense)	(None, 256)	6656256
dropout_5 (Dropout)	(None, 64)	0
dense_5 (Dense)	(None, 20)	1300
Total params: 6,994,780		
Trainable params: 6,994,780		
Non-trainable params: 0		
Train on 7086 samples, validate on 1182 samples		
Epoch 1/6000		

Figure 7: Portion of the DCASR1 Architecture Parameters

Figure 7 summaries the Tensor flow model for the architecture. The number of convolution layers is four, 5 layers fully connected network. The first and second convolution layers are shown together with the last layer of the fully connected network.

Previous work on speech recognition reported 98.5% recognition rate [14]. The main advantage over the previous results is the self extraction ability of the DCASR1, and the weight sharing methodology in the feature extraction stage. The feature engineering process for a recognition task is very tedious and time consuming. It is a great advantage over traditional systems. This enables the architecture to perform step towards intelligent behavior.

We implement the DCASR2 on raw speech signal. The recognition improved significantly, and reported 99.95% recognition rate on the training data, and 99.02% on the testing data (Figure 8).

Train data Large CNN Error: 0.05%
Test data Large CNN Error: 0.98%
[0.04159180211998975, 0.9994557082596272]
[0.14022109681037098, 0.9902067464635473]
[0, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 18, 19, 0, 1]
[0, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 18, 19, 0, 1]
99.94557082596272 % acc
99.02067464635473 % val_acc
99.34711627524358 % Max val_acc

Figure 8: DCASR2 Recognition Rates for Training and Testing

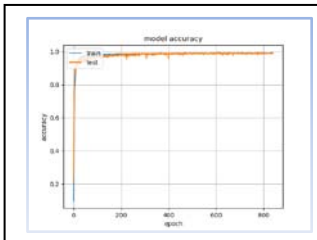

Figure 9: DCASR2 Accuracy Visualization for the Training and Testing Sets

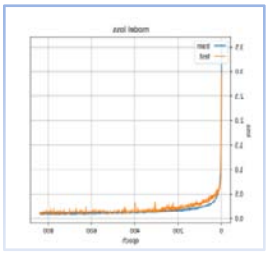

Figure 10: DCASR2 Loss Visualization for the Training and Testing Sets

Figure 9 graph the accuracy model for the training and testing data sets. The horizontal axis gives the epochs and the vertical shows the accuracy. Figure 10 visualize the loss model for both training and testing data. Horizontally the epochs, and vertically the loss values.

Using TensorFlow backend.		
Dataset shape is (8268, 8000)		
Dataset shape (8268, 8000)		
Max trainY = 19 Max testY = 19		
Min trainY = 0 Min testY = 0		
Train X shape is (7349, 8000)		
Test X shape is (919, 8000)		
Train X shape is (7349, 100, 80)		
Test X shape is (919, 100, 80)		
Train Y shape is (7349, 20)		
Test Y shape is (919, 20)		
Layer (type)	Output Shape	Param #
conv1d_1 (Conv1D)	(None, 100, 256)	20736
max_pooling1d_1 (MaxPooling1	(None, 100, 256)	0
conv1d_2 (Conv1D)	(None, 100, 240)	61680
max_pooling1d_2 (MaxPooling1	(None, 100, 240)	0
flatten_1 (Flatten)	(None, 20800)	0
dense_5 (Dense)	(None, 20)	2580
Total params: 11,065,908		
Trainable params: 11,065,908		
Non-trainable params: 0		
Train on 7349 samples, validate on 919 samples		
Epoch 1/6000		

Figure 11: Portion of the DCASR2 parameters

DCASR2 parameters are given in Figure 11. Summaries the model for the DCASR2 architecture. The total number of convolution layers is four. The fully connected network composed of five layers. The first and second convolution layers are shown together with the last layer of the fully connected network.

We implement the architecture using TensorFlow. DCASR1 reported 1.27% test data error, (Figure 4). In the other hand DCASR2 reported 0.98% test data error (Figure 8). The total parameters for DCASR1 are 6,994,780 compared to 11,065,908 for DCASR2, Figures 7 and 11 respectively. DCASR1 train on 7086 samples and validate on 1182 (Figure 7). While DCASR2 train on 7349 and validate on 919 (Figure 11).

Despite the large number of parameters, DCASR2 outperform DCASR1 in the recognition ability of the network. Moreover the use of the raw signal in DCASR2 is considered advantageous over DCASR1.

5 Conclusions

In this work, a convolution neural network is used to deep learning architectures for speech recognition. Preprocessing enhances the self extraction ability of the CNN. It is sequence of normalization, silence removal, and resizing. The architectures consist of two stages, feature extraction and classification. The feature extraction composed of four convolution layers, followed by max pooling layer each. Classification stage is a five layers backpropagation network. DCASR1 achieved 98.7% recognition rate. DCASR2 achieved 99.02% recognition rate, which is considered an outstanding.

REFERENCES

[1] Li Deng, Geoffery Hinton, and Brian Kingsbury, "New Types of Deep Neural Networks Learning for Speech Recognition and Related Applications: An Overview". IEEE ICASSP 2013**.**

[2] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, A. Stolcke, D. Yu, and G. Zweig, " The Microsoft 2016 Conversational Speech Recognition System", IEEE ICASSP 2017.K. Elissa, "Title of paper if known," unpublished.

[3] Y. LeCun and Y. Bengio, : Convolutional Networks for Image, Speech, and Time Series", The Hand Book of Brain Theory and Neural Networks, Vol. 3361, PP 276-279, 2003.

[4] Y.Qian, M. Bi, T. Tan, and K. Yu, " Very Deep Convolutional Neural Networks for Noise Robust Speech Recognition", IEEE/ACM Transaction on Audio Speech, and Language Processing, Vol 24, PP 2263-2276, 2016.

[5] T. Mikolov and G. Zweig, " Context Dependent Recurrent Neural Network Language Model", in SLT PP. 234-239, 2012.

[6] I. Sutskever, O. Vinyals, and Q. V. Le, " Sequence to Sequence Learning with Neural Networks", in Advances in Neural Information Processing Systems, PP 3104-3112, 2014.

[7] Doddington, George R. and Schalk, Thomas B., "Speech Recognition: Turning Theory to Practice", in IEEE Spectrum, September, 1981, PP. 26-32.

[8] Schalk, Thomas B., "The Design and Use of Speech Recognition Data Bases", in "Proceedings of the Workshop on Standardization for Speech I/O Technology", March 18-19, 1982, PP. 211-214.

[9] Ossama Abdel-Hamid; Abdel-Rahman Mohamed; Hui Jiang; Li Deng; Gerald Penn; Dong Yu. "Convolutional Neural Networks for Speech Recognition", IEEE/ACM Transactions on Audio, Speech, and Language Processing. Vol. 22 Issue 10, 2014 PP. 1533-1545.

[10] Chin Jen-Tzung; Misbullah Alim, " Deep Long Short Memory Networks for Speech Recognition". IEEE 2016 978-1-5090-4294-4.

[11] Yu Zhang, William Chan, Navdeep Jaitly. " Very Deep Convolutional Network for End- to –End Speech Recognition", 2017, IEEE International Conference on Acoustic Speech and Signal Processing Proceedings, 4845-4849 DOI:10 1109/ICASSP 2017.7953077.

[12] Palaz, Dimitri, "Convolutional Neural Networks-based continuous speech recognition using raw speech signal", Book: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2015. ISSN: 1520-6149, ISBN: 1-4673-6997-7, 978-1-4673-6997-8, PP 4295-4299 . DOI: 10.1109/ICASSP.2015.7178781.

[13] Yanmin Qian, Woodland, P.C. "Very Deep Convolutional Neural Networks for Robust Speech Recognition". *2016 IEEE Spoken Language Technology Workshop (SLT) Spoken Language Technology Workshop (SLT), 2016 IEEE*.:481-488 Dec, 2016

[14] Elsadig Ahmed, AbdRahman Ramli, and Hanan Adlan. "A Combined Rough Set-K-Means Vector Quantization Model for Arabic Speech Recognizer", (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 7 (1), 2016, 260-265.

[15] Ossama Abdel-Hamid; Abdel-Rahman Mohamed; Hui Jiang; Gerald Penn; " Applying Convolution Neural Networks Concepts to Hybrid NN-HMM model for Speech Recognition" . IEEE International Conference on Acoustic Speech and Signal Processing, ICASSP 2012, PP 4277-4280.