International Journal of Scientific Research & Engineering Trends
Volume 8, Issue 5, Sept-Oct-2022, ISSN (Online): 2395-566X

Optimization for Speech to Text Conversion Using Convolutional Neural Network

Rahul Singh Sengar
Specialization in Machine Learning and Data Science,
Bachelor of Technology,
Department in Computer Science,
Turari, ITM University,
Gwalior, Madhya Pradesh, (India))

Vatsal Mehta
Specialization in Machine Learning and Data Science,
Bachelor of Technology
Department in Computer Science,
APSIT, Mumbai University,
Maharashtra, (India))

Abstract- The field of machine learning has taken a dramatic twist in recent times, with the rise of the Artificial Neural Network (ANN). These biologically inspired computational models are able to far exceed the performance of previous forms of artificial intelligence in common machine learning tasks. One of the most impressive forms of ANN architecture is that of the Convolutional Neural Network (CNN). CNNs are primarily used to solve dif icult image-driven pattern recognition tasks and with their precise yet simple architecture, of er a simplified method of getting started with ANNs. This document provides a brief introduction to CNNs, discussing recently published papers and newly formed techniques in developing these brilliantly fantastic image recognition models. This introduction assumes you are familiar with the fundamentals of ANNs and machine learning. The ability to accurately represent audio signals is central to language understanding. The network uses Conv1d, a global pooling operation over linear sequences. The network handles input audio signals of varying lengths and induces a feature graph over the audio signals that are capable of explicitly capturing short and long-range relations. The network does not rely on a parse tree and is easily applicable to any language. We test the CNN in Modeling Audio signals to Texts. The network achieves the excellent performance of a greater than 25% error reduction in the last task with respect to the strongest baseline.

Keywords- CNN, Artificial Neural Networks, Machine Learning, Conv1d Analysis, Max Pooling.

I. INTRODUCTION

Traditional solutions use a mix of deep learning and Natural Language Processing techniques to parse through our query, retrieve the answer and present it in the form of both audio and text. The problem is that we might be on the verge of having too many digital screens around us. As it seems like every day, new versions of common objects are “re-invented” with built-in wifi and bright touchscreens, promising some antidote to our screen addiction is the voice interfaces.

TensorFlow recently released the Speech Commands Datasets. It includes 65,000 one-second long utterances of 30 short words, by thousands of different people. So, our solution will be to build a speech recognition system using Convolutional Neural Networks (CNN) that understands simple spoken commands. CNN differs from other forms of Artificial Neural networks in that instead of focusing on the entirety of the problem domain, knowledge about the specific type of input is exploited.

1. Concept:
We understand a dynamic convolutional neural network that uses the dynamic k-max pooling operator as a nonlinear subsampling function. The featured graph induced by the network is able to capture word relations of varying size. The net-work achieves high performance on question and sentiment classification without requiring external features as provided by parsers or other resources. It also provides depth knowledge about (NLP) is the computerized approach to analyzing text that is based on both a set of theories and a set of technologies. Being a very active area of research and development, there is not a single agreed-upon definition that would satisfy everyone, but there are some aspects, which would be part of any knowledgeable person’s definition.

2. Current Limitations:

Python’s Speech Recognition	Google Cloud Speech MySTT Service
Free but limited provision of service	ll be free to use and would be integrated to form a python library.
Free but limited provision of service	ll be free to use and would be integrated to form a python library.

One of the biggest gaps is equitable access to resources. Quality resources like Google Cloud Speech Services have a pay-as-per-use policy that restricts a large portion of society while freely available resources are not up to the mark in comparison.

II. MATERIALS AND METHODS

1. Literature Survey:

1.1 Google Cloud Speech Service: It contains the following features: Speech adaptation, Domain-specific models, Streaming speech recognition, etc.

1.2 Python's Speech Recognition: Library for performing speech recognition, with support for several engines and APIs, online and offline. It contains a long requirement list to work.

1.3 Gaps in the literature: One of the biggest gaps is that If the service is free, then it either requires huge library sizes, or it will not be free and requires you to pay the bill with every use.

1.4 Briefly state the Objectives and Proposed Approach:

- Strategy and/or methodology of work: By NLP I would extract the audio signal to something that the machine can understand and then comes the CNN to take that process data to analyze and return the expected text. As we will be coding in python, so it's expected to use python libraries for audio files.
- Scope and boundaries of the work, including any issues that will not be covered. This service is strictly only a Speech To Text converter and it can't be either trained or tested with huge data size (as my computer can't process such huge data). However, I will provide the full data for any individual who wants to run train, and test the model.
- Data analysis (sample size, data collection)
  - Size of train data: 2.1 GB
  - Size of test data: 5.1 GB
  - Size of sample submission: 4.3 MB

2. Audio Signals:

This is pretty intuitive – any object that vibrates produces sound waves. Have you ever thought of how we are able to hear someone's voice? It is due to the audio waves. Let's quickly understand the process behind it. When an object vibrates, the air molecules oscillate to and fro from their resting position and transmit their energy to neighboring molecules. This results in the transmission of energy from one molecule to another which in turn produces a sound wave.

Parameters of an audio signal:

- **Amplitude:** Amplitude refers to the maximum displacement of the air molecules from the rest position.
- **Crest and Trough:** The crest is the highest point in the wave whereas the trough is the lowest point.
- **Wavelength:** The distance between 2 successive crests or troughs is known as a wavelength.
- **Cycle:** Every audio signal traverses in the form of cycles. One complete upward movement and downward movement of the signal form a cycle.
- **Frequency:** Frequency refers to how fast a signal is changing over a period of time.

3. Spectrogram:

It's a 2D plot between time and frequency where each point in the plot represents the amplitude of a particular frequency at a particular time in terms of the intensity of color. In simple terms, the spectrogram is a spectrum (broad range of colors) of frequencies as it varies with time.

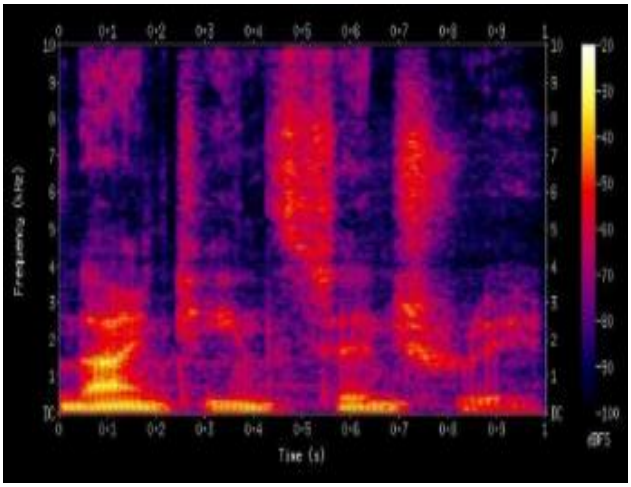

Fig 1. The right features to extract from audio depend on the use case.

Data Exploration and Visualization helps us to understand the data as well as pre-processing steps in a better way.

4. Visualization of Audio signal in time series domain:

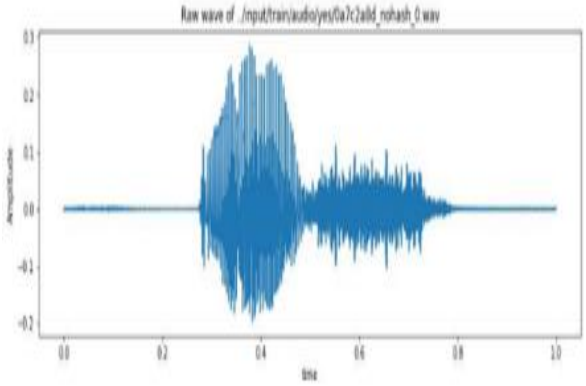

Fig 2. Audio signal in the time series domain.

5. Preprocessing the audio waves:

In the data exploration part earlier, we have seen that the duration of a few recordings is less than 1 second and the

sampling rate is too high. So, let us read the audio waves and use the below-preprocessing steps to deal with this.

Here are the two steps we'll follow:
- Resampling
- Removing shorter commands of less than 1 second

```
train_audio_path = '../input/tensorflow-speech-recognition-challenge/train/audio/'

all_wave = []
all_label = []
for label in labels:
    print(label)
    waves = [f for f in os.listdir(train_audio_path + '/'+ label) if f.endswith('.wav')]
    for wav in waves:
        samples, sample_rate = librosa.load(train_audio_path + '/' + label + '/' + wav, sr = 16000)
        samples = librosa.resample(samples, sample_rate, 8000)
        if(len(samples)== 8000) :
            all_wave.append(samples)
            all_label.append(label)
```

Fig 3. Preprocessing Steps.

Here is the complete architecture of our SpeechToText Model: Here is the Model architecture for the conversion of audio files to text.

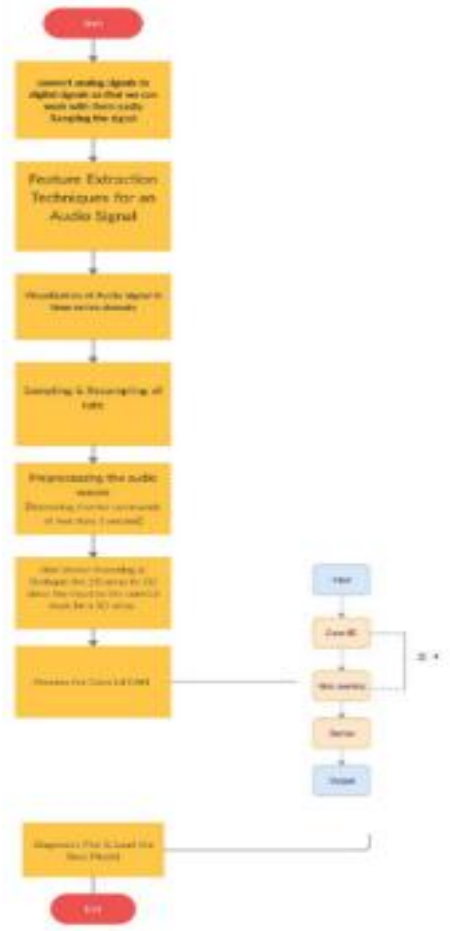

Fig 4. Complete workflow diagram.

III. RESULTS AND DISCUSSIONS

1. Output Shape:

```
=================================================================
=======================
```

Total params: 1,611,498
Trainable params: 1,611,498
Non-trainable params: 0

2. Diagnostic Plot:

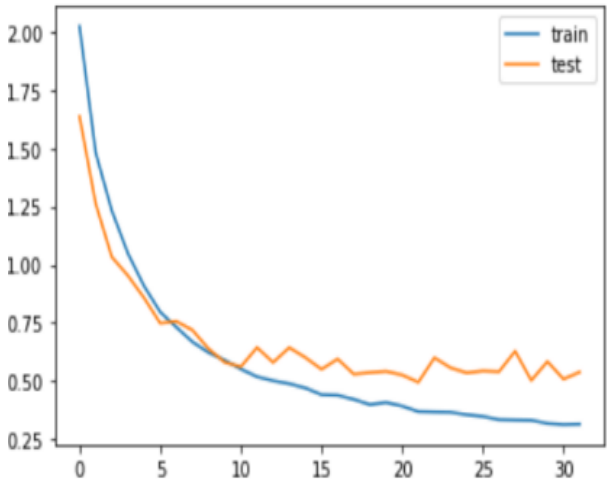

Fig 5. Line graph of train and test data.

IV. CONCLUSION

We have successfully achieved the following accuracy results: accuracy: 0.6157 - val_loss: 0.9529 - val_accuracy: 0.6636

```
In [19]:
history=model.fit(x_tr, y_tr ,epochs=100, callbacks=[es,mc], batch_size=32, validat

Epoch 1/100
533/533 [==============================] - 70s 128ms/step - loss: 2.
2076 - accuracy: 0.1532 - val_loss: 1.6353 - val_accuracy: 0.4114
WARNING:tensorflow:Can save best model only with val_acc available,
skipping.
Epoch 2/100
533/533 [==============================] - 66s 124ms/step - loss: 1.
5837 - accuracy: 0.4143 - val_loss: 1.2620 - val_accuracy: 0.5534
WARNING:tensorflow:Can save best model only with val_acc available,
skipping.
Epoch 3/100
533/533 [==============================] - 70s 131ms/step - loss: 1.
2705 - accuracy: 0.5418 - val_loss: 1.0332 - val_accuracy: 0.6399
WARNING:tensorflow:Can save best model only with val_acc available,
skipping.
Epoch 4/100
533/533 [==============================] - 67s 125ms/step - loss: 1.
0768 - accuracy: 0.6157 - val_loss: 0.9529 - val_accuracy: 0.6636
WARNING:tensorflow:Can save best model only with val_acc available,
```

Fig 6. Model Fitting.

VI. ACKNOWLEDGEMENT

We would like to thank ITM University Gwalior for providing the necessary prerequisites for conducting this research. We hereby acknowledge that everything mentioned in this paper is true to our knowledge. We further declare that the work is free from any plagiarism.

REFERENCE

[1] O & 39; Shea, K. and Nash, R., 2021. An Introduction to Convolutional Neural Networks. [Online] arXiv.org. Available at: <https://arxiv.org/abs /1511.08458> [Accessed 27 April 2021].
[2] Kalchbrenner, N., Grefenstette, E. and Blunsom, P., 2021. A Convolutional Neural Network for Modelling Sentences. [Online] arXiv.org. Available at: <https://arxiv.org/abs/1404.2188> [Accessed 27 April 2021].
[3] [3] Ieeexplore.ieee.org. 2021. Understanding of a convolutional neural network. [online] Available at: <https://ieeexplore.ieee.org/abstract/document/8308186> [Accessed 27 April 2021].
[4] Ieeexplore.ieee.org. 2021. Speech Emotion Recognition from Spectrograms with Deep Convolutional Neural Network. [online] Available at: <https://ieeexplore.ieee.org/abstract/document/7883728> [Accessed 27 April 2021].
[5] Hertel, L., Phan, H. and Mertins, A., 2021. Classifying Variable-Length Audio Files with All-Convolutional Networks and Masked Global Pooling. [Online] arXiv.org. Available at: <https://arxiv.org/abs/1607.02857> [Accessed 27 April 2021].
[6] Chowdhury, Gobinda G. "Natural Language Processing." Annual Review of Information Science and Technology, vol. 37, no. 1, 31 Jan. 2005, pp. 51–89, strathprints.strath.ac.uk/2611/1/strathprints 002611.pdf, 10.1002/aris.1440370103.
[7] Hirschberg, J., and C. D. Manning. "Advances in Natural Language Processing." Science, vol. 349, no. 6245, 16 July 2015, pp. 261–266, science.sciencemag.org/content/349/6245/261, 0.1126/science.aaa8685.
[8] Manning, Christopher, et al. The Stanford CoreNLP Natural Language Processing Toolkit. Association for Computational Linguistics, 2014.
[9] Nadkarni, Prakash M, et al. "Natural Language Processing: An Introduction." Journal of the American Medical Informatics Association, vol. 18, no. 5, Sept. 2011, pp. 544–551, academic.oup.com/jamia/article/18/5/544/829676, 10.1136/amiajnl-2011-000464. Accessed 4 Sept. 2019