ulm university universität
uulm

Ulm University | 89069 Ulm | Germany

Faculty of
Engineering, Computer Science and Psychology
Neural Information Processing

A long title splitted into two lines

Master thesis at Ulm University

Submitted by:
Ethan Swistak
ethan.swistak@uni-ulm.de

Reviewer:
Prof. Dr. James Butler
Prof. Dr. Jing Zhou

Supervisor:
Joe Help

2018

Version from December 5, 2020

Abstract

Speaker identification has gained wide traction as an application area for artificial neural networks over recent years. However, the majority of implementations have utilized i-Vector methods. An emerging novel approach to speaker identification has been the application of convolutional neural networks

==========

ChangeLog:

2018-10-10: Small changes in structure
2018-10-09: Faculty and department name adjusted

Acknowledgment

Acknowledgement goes here. Thanks to Guido de Melo for providing Version 1.0 of this template!

Contents

1

Introduction

This project will investigate the marginal cost and methods of adding new speakers into a convolutional neural network trained for speaker identification. The goal is to identify an approach that will allow additional speakers to be added to the network and additionally be identified when they issue a new, previously unheard utterance. The project will investigate different approaches to adding additional speakers to a trained neural network and attempt to quantify the additional loss incurred, what, if any, effect on accuracy this has, and the training time such an approach would require. Ideally, this could result in systems that could learn the identities of their users and have applications in both human-computer interraction and security areas.

1.1 Problem statement

Given a pre-trained convolutional neural network that is able to identify a speaker given a 3-second sample utterance or a series of 3-second sample utterances add additional speakers to the neural networks corpus incrementally such that the network learns to identify new speakers.

1.2 Contributions and results

In the process of adding speakers we want to ensure that both the additional training time required is minimized and that the accuracy of the network remains high.

1.3 Structure of the thesis

This report will first go over the architecture and pre-processing pipeline for the neural network and give a brief overview of the data which the network was trained on. We will additionally summarize the results of previous experiments conducted by the researchers and finally, give an overview of what changes we intend to make to the network in order to allow for it to be trained incrementally.

2

Methods

The dataset used in this project is the VoxCelebA dataset, which consists of a corpus of .wav files generated from youtube with labeled celebrity speaker identities. The speakers have been hand labeled along with their gender and nationality. There are, in total 148,642 utterances divided among 1,211 speakers in the training dataset. The dataset contains a good separation of male and female speakers, although the majority of the speakers hail from english speaking countries, which may lead to some bias.

The convolutional neural network used in this experiment is a state of the art design which consists of 3 components, a pre-training layer which, in addition to generating 3-second windowed samples from the longer .wav files applies some pre-processing steps to make the resultant data more amenable to training the neural network. A ResNet layer which learns a representation of the resultant utterance and outputs a feature vector, and a self-attention/decision layer that applies a weighting to the resultant feature vector and applies a probability to the speaker's identity.

2.1 Pre-processing

Given that the audio samples are of a variable length in the training set, it is necessary to cut them down into 3-second chunks for further processing. There are also some mathematical conversions applied to the audio signal in order to better emphasize features relative to training and limit the influence of approximation errors.

Pre-emphasis

Signal energies in speech identification tasks tend to drop off at higher frequencies. The exact rate of decrease tends to vary among speakers and environmental conditions but a good rule of thumb is a dropoff of 2 dB/kHz. This presents a problem for finite discretization of a signal during a fourier transform since floating point values tend to lose precision given large variations in min and max values. To compensate for this fact, a linear pre-emphasis filter can be applied to the speech signal in the time domain to boost the signal energies in higher spectral ranges. The formula is:

$$\forall X \in X, x[n] = x[n] - \alpha x[n-1]$$

where α is a pre-determined constant.

Framing

Since the speech signals occur in files of varying length, the speech signals must be windowed into 3-second chunks to be fed into the neural network. The effect of doing so is essentially a convolution with a function equal to one in the sample range and zero elsewhere. This can lead to some undesirable distortions of the spectrum whent he signal is discretized. Specifically, since the fourier transform doesn't know about the remaining signal outside of the window it may attempt to compensate for very low frequencies that the window does not capture by skewing other higher frequencies to compensate for that part of the signal. This leads to a noisy signal with too much energy in the higher frequency bands that is not "real" signal energy.

Hamming Window

The hamming window is an attempt to suppress the noise create by attempting to sample an infinite signal in a finite window. It essentially attempts to generate some "counter-noise" to cancel out the noise resulting from the windowing function. A variety of window functions have been proposed in an attempt to counteract this high frequency noise but one of the most widely used is the hamming window:

$$H(\theta) = 0.54 + 0.46\cos[\frac{2\pi}{N}n]$$

While a mathematical proof of this formula is outside the scope of this paper, suffice it to say that this results in a signal that is a more pure approximation the frequencies actually contained in the window and reduces the impact of frequencies that are too low to be detected in such a finite window.

Fourier Transform

The fourier transform is used to obtain the frequency components and signal energies associated with these frequency components given a signal in the time domain. It is widely used in a variety of signal processing tasks. Most computers use an algorithm to compute the fourier transform known as the fast fourier transform that takes advantage of the orthogonality of the transform in different dimensions. This brings the time complexity of the transformation down from $O(n^2)toO(n \log n)$.

Log-Mel Features

The final step in the pre-processing pipeline is to apply emphasis to the signal to better approximate how it would be perceived by a human listener. Since, in this case, the network is being trained to identify human speech patterns it makes sense to modify the signal such that it is more in line with human auditory perception. The mel-scale is based on research into the human auditory system and applies a transformation to the input signal such that sounds of equal distance from eachother also "sound" as if they are equal distance, ie, if one tone is twice as far away from some base tone as another tone then the tone would sound twice as high. The formula is:

$$m[f] = 2595 \log 10(1 + \frac{f}{7000})$$

2.2 ResNet Architecture

The resnet, short for Residual Network, is a state of the art convolutional neural network that attempts to partially solve the vanishing gradient problem by introducing short connections between different layers of the network, thereby lower level information to flow directly to a higher level activation function. The network is able to consider

information inputs from two layers simultaneously. This also provides a shorter path during backpropagation that mitigates some of the effects of the vanishing gradient. While ResNets have been primarily applied as feature detectors in visual computing applications, they are also finding increasing usage in auditory processing of spectral graphs. Although speech patterns do not contain the same types of local structure as images, the patterns can be identified in much the same way for both types of information.

The Resnet inroduces the concept of an "identity shortcut connection" that skips one or more layers. This partially solves the vanishing gradient problem since it provides a shorter alternate path for gradients to propagate without impacting the overall network performance. The argument of the origional authors was that stacking residual layers should not reduce network performance since the network, in the worst case, could use an identity mapping for all the shortcuts and simply achieve the same performance as before. In this case, this would indicate that a deeper network should not produce more error than it's shallower counterpart. Much like a Long-Term Short-Term network whose "forget gate" controls how much information is transfered from one time-step to the next, the ResNet architecture learns how to control how much information from lower layers is passed to higher layers in the network.

2.3 Self-Attention Layer

Self-Attention layers have gained increasing attention since Vaswani et al wrote the seminal paper "Attention is All You Need" which advanced the state of the art in sequence transduction models using solely attention cells as the building blocks and achieved outstanding results. However, attention layers have been used for a long time previously in RNN and CNN architectures to provide an efficient mechanism to relate distant dependencies. In speech comprehension, this would take the form of relating different terms in a long sentence. However, in our case, we are attempting to extract sections of the feature vector which are most relevant to the task of speaker classification.

Self-attention, sometimes called intra-attention, is an attention mechanism relating different positions of a single sequence in order to compute a representation of the sequence. An attention function can be seen as mapping a set of key-value pairs to an output, where query, keys, values, and outputs are all vectors. The output is a weighted sum of the input vectors, where each value is computed by a probability function to the key. The self-attention layer used in our network is a Scaled Dot-Product Attention layer, our key in this case is simply a learned series of weights but we add a second layer of keys to further segregate the input feature vector.

The self-attention layer produces, what is in essence, a weighting to apply to different features so that only the most relevant features of the speech sample are considered in classifying the speaker. Since applying too much weight to many different features would defeat the purpose of having an attention layer, a factor is added to the loss function which encourages the network to limit the number of attention hops it performs, keeping the features considered limited.

Bibliography

[1] Schwenker, F., Kestler, H.A., Palm, G.: Three learning phases for radial-basis-function networks. Neural networks **14** (2001) 439–458

A

Sources

Appendix contains important source code snippets.

```
1 public class Hello {
2     public static void main(String[] args) {
3         System.out.println("I love machine learning");
4     }
5 }
```

Listing A.1: Lines of code

List of Figures

List of Tables

Name: Ethan Swistak Matriculation number: 1234567890

Honesty disclaimer

I hereby affirm that I wrote this thesis independently and that I did not use any other sources or tools than the ones specified.

Ulm, ...

Ethan Swistak