MERIVALE HIGH SCHOOL

Investigating the Impact of Various Feature Extraction Algorithms on Performance in Automatic Speech Recognition Systems

To What Extent Do Mel-Frequency Cepstrum Coefficients Outperform Other Speech Feature Extraction Algorithms in Speech Recognition Neural Networks?

Author:
Daniel LU

Supervisor:
Courtney EDWARDS

October 7, 2022

Contents

List of Abbreviations

ASR	**A**utomatic **S**peech **R**ecognition
LPC	**L**inear textbfPredictive **C**oding
DFT	**D**iscrete **F**ourier **T**ransform
FFT	**F**ast **F**ourier **T**ransform
IFT	**I**nverse **F**ourier **T**ransform
HMM	**H**idden **M**arkov **M**odel
CTC	**C**onnectionist **T**emporal**C**lassification
DDC	**D**irected **D**ialogue **C**onversations
NLC	**N**atural **L**anguage **C**onversations
NLP	**N**atural **L**anguage **P**rocessing
DWT	**D**iscrete **W**avelet **T**ransform
PLP	**P**erceptual **L**inear **P**rediction

Chapter 1

Introduction

1.1 Abstract

Automatic speech recognition (ASR) is the ability of a machine to recognize language from human speech and convert it into written text. Such a task is difficult due to the sheer complexity of any spoken language, as well as the varying and unpredictable nature of different people and their speaking habits. Even big-name companies such as Alphabet, Microsoft, and Baidu, who pride themselves in their sophisticated multi-million dollar speech recognition systems, have been countlessly ridiculed for their misinterpretations. However, due to the endless number of possible and practical applications of ASR technology in our developing world, major corporations still aspire to develop an accurate yet optimized speech recognition system for their products and services. For instance, Baidu spent a total of $4.7 billion dollars on research and development, of which a large portion headed into developing their voice assistant, DuerOS. Hence, the implementation of ASR can improve the ergonomics of any task that requires human interaction, such as personal assistants, smart healthcare and household appliances, and media transcription to support people with disabilities or language barriers.

Significant development for ASR began in the 50s, with the innovation of Bell's Audrey and IBM's Shoebox. These systems fell into a category of ASR systems that recognized individual words to an isolated vocabulary using manually set parameters for phoneme recognition. By the end of the decade, ASR technology could distinguish words with a maximum of four vowels and nine consonants and by the 80s, systems could recognize up to 3000 words, which is around the vocabulary of a six-year-old child. It wasn't until the late 90s that researchers transitioned outwards of phoneme detection to Hidden Markov Models (HMM) and focused on natural language processing. Deep learning concepts and technologies emerged and were investigated as a solid alternative to HMMs, to the extent to which it is prominently used today.

However, a plausible deep learning system that can recognize continuous speech within a low margin of error is very strenuous to build. A well-designed system can consist of multiple complex layers that can be very complicated to integrate into one another. Nowadays, large tech companies boast an accuracy of almost 95

1.2 Rationale

This report aims to investigate the extent to which Mel-Frequency Cepstrum Coefficients outperform other feature extraction methods in automatic speech recognition systems. I will assess the performances of MFCCs, Discrete Wavelet Transforms (DWT), Linear Predictive Coding (LPC), and Perceptual Linear Prediction (PLP), as well as the absence of a feature extraction layer, processed through identical recurrent neural network architectures. From this investigation, I will better understand the effectiveness and limitations of different speech feature extraction methods and inform my forthcoming decisions in ASR neural networks.

Chapter 2

Background

2.1 Automatic Speech Recognition

Automatic speech recognition models can be split into two broad categories of complexity. First of all, there is single word classification to an isolated vocabulary where directed dialogue conversations are analyzed; its functionality can be seen in early speech recognition efforts such as IBM's Shoebox. The other variant is continuous speech recognition, where natural language conversations (NLCs) are processed and transcripted to a vocabulary of around 20,000 to 100,000 words. Many technologies today utilize this system, including video subtitles, voice searching, and customer services at call centres. Directed dialogue classification models can be fairly simple to understand and produce through various feed-forward networks, but natural language networks require the use of recurrent neural networks.

2.2 Recurrent Neural Networks

Recurrent neural networks are a variant of deep learning neural networks that use time-sequential data without a fixed shape. These systems are inspired by the biological composition of neurons in the brain, as humans have learned long ago to adapt their innovations from nature. Each neuron, out of around 100 million, is interconnected to many other neurons and passes a signal via its axon to receiving axon terminals belonging to other neurons. Contrastingly, in an artificial neural network, a combination of weights and biases stacked in layers are constantly optimized through gradient descent to produce a matrix of probabilities that highlight the best possible output the network can generate at its current state (for the case of classification architectures). In each cell, a dot product of the input matrix and the cell's weight matrix is done and summed with the cell's bias matrix to create an output matrix of values, fed into all of the nodes in the next layer. At first, these matrices use senseless, randomly generated weights and biases that result in a completely random output. However, as the model calculates its loss from an evaluation of the net deviation of the predicted output from the ground truth, a gradient is formulated that adjusts each layer's parameters closer to what would create an output with a lower loss. The higher loss an output receives, the greater magnitude of the gradient is passed to backpropagation and the more each node is changed, with the vice versa applying as well.

In a typical multi-layer deep learning neural network, the input and output shape of the model are fixed and have to be set. Since the length of the NLC audio input is indeterminate in this investigation, recurrent neural networks can be used to process continuous speech data that last an indeterminate length of time. RNNs accomplish this task by accepting an input for each time step into a recurrent cell and feeding the cell's generated output to the succeeding cell through a hidden layer, doing so until the input data is completely used. This results in the coupled nature of a recurrent neural network that enables it to find relationships in continuous data, namely in stock market prediction, and influence future outputs. The framework of a standard RNN cell is laid out below (Figure 1).

The distribution of input data and output data can vary from network to network and is ultimately based on the shape of the input data and the desired shape of the output data. For example, a music generation RNN could take in a dataset of Vivaldi's works as an input and output an infinite length of completely original Vivaldi-styled music: this model would be classified as a one-to-many RNN structure. The following table (Figure 2) describes four main types of RNN structures that satisfy most machine learning purposes.

Due to the continuous nature of NLC, multiple inputs as the audio data and its respective outputs as phonemes, letters, or words are present. Thus, a many-to-many structure will be used as the framework for this investigation's experimentation.

However, conventional recurrent neural networks have two major problems that need to be considered: the vanishing and exploding gradient problem, and the long-term dependency on sound signals. The vanishing and exploding gradient problem appear during the backpropagation stage of the network, where a gradient propagates through each cell and updates the weights and biases of each node in order for the network to learn. The problem arises when the number of layers starts to increase and the gradients start to travel and multiplied through more and more layers; when a gradient with a slight deviation from y=x is returned to the first node during backpropagation, it begins to increase or decrease exponentially (dependent on its sign) as it progresses through each cell. Correspondingly, as it reaches near the end of backpropagation, the gradient is either too minuscule to the point where the end weight is scarcely updated at all, or too large where the weight cannot be optimized properly.

Going on, sound signals can take up a significant portion of input data, dependent on sample frequency, as a one-second audio clip of someone dictating a word can take up to 16,000 time steps of data. The extreme short-term memory of a simple hyperbolic tangent recurrent network cannot remember what sound signal was inputted into the cell some 5,000 iterations earlier, to the extent it is significant towards producing the output. This is detrimental towards the ultimate performance of the neural network, as a large segment of a model's decision on a phoneme, character, or word depends on contextual data beforehand. For example, a model could have a hard time discriminating between the letter “k” and “q” with no context. However, the prerequisite knowledge that it was preceded by a “loo” will likely increase the probability that the following letter is “k”. These two setbacks of recurrent neural networks introduced a revolutionary recurrent cell concept introduced in 1997 and prominently used today, termed the Long Short-Term Memory cell.

2.3 Long Short-Term Memory

Long Short-Term Memory cells counter the two problems above by incorporating an extra hidden memory state that propagates throughout recurrences of the network. The addition of a memory state that can remember important specifics from many recurrences beforehand and influence the current output. Contrary to the perception-based framework of traditional neural networks, an LSTM cell is much more convoluted and contains substantially more weights and biases in the form of specialized gates (Figure 3). The combination of these properties provides the fundamental framework that conceives the LSTM network’s ability to recognize context-sensitive information and is thus the reason why these networks are the standard for building reliable speech recognition systems.

A simple speech recognition system is usually composed of two different components; a feature extraction layer and the neural network itself. The purpose of the feature extraction layer is to process the raw input data into a more usable and space-efficient form. In the case of ASR, input data is usually in the form of waveform data embedded in .mp3 or .wav files. This one-dimensional waveform data has several limitations due to having less useful features than a two-dimensional format of audio signals, such as spectrograms. Spectrograms are a visual representation of the amplitudinal spectrum of frequencies of an audio signal with respect to time. These isolated frequency amplitudes are also called the Discrete Fourier Transform (DFT) of an input and they can be extracted via the Fast Fourier Transform (FFT) algorithm, laid out below.

DFTs are frequently used in convolutional neural networks due to their property to be represented as an image and are the reliable standard in terms of extracting audio features from waveform data. In a DFT, the x-axis refers to time and the y-axis refers to the frequency that’s played at a certain point in time, with a lighter colour indicating that the amplitude is larger with regard to magnitude (Figure 4).

2.4 Feature Extraction Algorithms

However, spectrograms are not very well suited nor optimized for visualizing human speech. For instance, a human voice at a conversational level will normally not exceed a frequency of 500 Hz and fall below 60 Hz and it’s not necessary to include non-phonetically vital properties of speech signals in the input data. Therefore, Mel Frequency Cepstral Coefficients (MFCC) intended to artificially replicate the human hearing system with the assumption that the human ear is a reliable speech recognition system in itself. A precise logarithm function is used to retain the phonetically vital properties of speech signals and plot the result on the Mel scale (Figure 5).

MFCCs are created first by passing spectrogram data through a precise logarithm function, dubbed the Mel-filter bank, to retain the phonetically vital properties of speech signals while also reducing the number of unwanted frequencies in our input data. The result is plotted to a Mel spectrum and applied to the inverse discrete cosine transform, in which it is finally converted into a cepstrum, the inverse of a spectrum. The result is a set of 10-21 coefficients that ultimately present the main speech features of the original waveform data. MFCC data can also be taken as the derivative of and provide further feature data to feed into the neural network.

Other speech feature extraction functions that will be investigated are DWT, LPC, and PLP. They are each unique in their own manner but are also very closely related to each other with reference to the algorithms used to create them. Discrete Wavelet Transforms decompose a signal into a set of mutually orthogonal wavelet basis functions. It uses the Short-Time Fourier Transform, a variant of the FFT, which carries out a Fourier transform on a signal split into small windows of fixed duration instead of a specific time instance. Linear Predictive Coding generates coefficients that, similar to MFCCs, reflect the characteristics of a simplified vocal tract model, overall compressing the signal alongside the process. As the name suggests, the method is competent at predicting the future of a random process given past observations. Finally, Perceptual Linear Prediction claims to improve speaker independent recognition performance and outperform LPCs in speech recognition. It first warps the spectrogram along the Bark frequency scale (Figure 6), convolves it with the power spectrum, and performs a cubic-root amplitude compression, ultimately creating the linear prediction model.

Chapter 3

Experiment Materials & Methodology

In this paper, heavy importance is placed on primary experimental observations and data, as well as a complete understanding of every operation that is carried out. For those reasons, deep learning APIs like Keras or Torch were not implemented, though this might have an effect on the overall performance of my neural network due to the great amount of optimization such open-source projects have. Nevertheless, this doesn't have an impact on the results of my investigation, as the network will remain a control variable throughout all trials. The dependent variables are the speech feature extraction algorithms (MFCC, DWT, LPC, and PLP) and the background noise of each sample (none, white noise, and miscellaneous/random). To properly appraise the extent to which MFCCs can extract speech features, background noise must be a variable due to being a known downside to them. The independent variables are the accuracy and time allotted to each network when processing the test dataset.

3.1 Datasets Used

I used "Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition" published by Pete Warden to train my neural network. The dataset contains 6500 sets of one-word dialogue paired with its transcription as the true output. I later altered this dataset by concatenating copies of the audio data with various types of background noise using Librosa.

However, as I plan to use continuous speech recognition neural network architectures, using this dataset is only a simplistic and intuitive demonstration of what the model is capable of doing, in addition to an effort by myself to not be too ambitious and over-extend my experimentation for this report. Further developments of this model would feature and employ other open-source datasets that include multi-word speech data, such as podcasts or court hearings. I hope to alter this later on.

3.2 Pre-Processing

The dataset provided the sound files in .wav format, which made it very simple to extract its waveform data. I used ARSS's audio analysis program to perform an FFT on the dataset and obtain the respective DFT spectrograms. Going on, I used Librosa to retrieve the spectrograms' speech feature data to ultimately feed into the network.

3.3 Network Architecture

This report uses an end-to-end recurrent neural network consisting of a layer of LSTM cells and a final decoding node to process and display the output. The LSTM model consists of 256 hidden units that accept an input that's of shape (3, 26) and output a one-hot matrix of shape (28, 1) at each time step. The output is a probability matrix of all the letters of the alphabet the model predicts the audio signal refers to at that point in time, as well as a space and a "blank" character that are incorporated to separate concurrent letters in the combined output. A working result of this model would output a sequence of slurred letters that, once decoded with Connectionist Temporal Classification (CTC) loss, would output a word; the model could output "ww_eee_ee_p", which suggests that the original audio input is of a person saying "weep".

The model was trained on 5000 pieces of training data, 10 epochs of such, with a learning rate of 0.01 and evaluated by three validation datasets, for each of the three trials, that held 500 pieces of data to determine the accuracy of each model.

3.4 Experimental Process

Before I elaborate on my experimental process, it is essential to clarify some notable semantics of this investigation. The general performance basis of a model is assessed on a synthesis of its accuracy and its efficiency, though with a greater emphasis on accuracy. Whether an ASR system is successful is determined by whether its accuracy is greater than 50

I will hold three trials for each set of independent variables to answer my research question. The performance of raw spectrograms, MFCCs, DWTs, LPCs, and PLPs will be assessed against each other in three different dataset variants: with no background noise, with white noise, and with other miscellaneous background noises, amounting to a total of 45 trials.

Chapter 4

Background

4.1 Abstract

Purpose of this extended essay

4.2 Rationale

Choice of topic

Chapter 5

Analysis & Conclusions

5.1 Abstract

Purpose of this extended essay

5.2 Rationale

Choice of topic

Appendix A

Frequently Asked Questions

A.1 How do I change the colors of links?

The color of links can be changed to your liking using:

`\hypersetup{urlcolor=red}`, or

`\hypersetup{citecolor=green}`, or

`\hypersetup{allcolor=blue}`.

If you want to completely hide the links, you can use:

`\hypersetup{allcolors=.}`, or even better:

`\hypersetup{hidelinks}`.

If you want to have obvious links in the PDF but not the printed text, use:

`\hypersetup{colorlinks=false}`.