Investigating Issues with Machine Learning for Accent Classification

Zhewen Lou¹, Yitian Ren²

¹Department of Electronic Engineering, Tsinghua University, Beijing, 100084, China
²University of Michigan-Shanghai Jiaotong University Joint Institute, Shanghai Jiaotong University, Shanghai, 200240, China

*Corresponding author’s e-mail: louzw19@mails.tsinghua.edu.cn

Abstract. Speech recognition has become a widely researched topic for decades, and there were already some successful products which has been put into commercial use, like Siri. However, sometimes it is hard to distinguish the word because words of different accents have different pronunciations. For instance, in Japanese English /r/ is usually pronounced as /l/. Therefore, it is natural to think that speech recognition could be divided into two parts. First attach a label to the audio about the accent, then recognize the contents based on the regular pattern of that accent. In this paper, we researched on several characteristics including voice onset region(VOR), vowels and formants to distinguish British English and American English. By applying both linear neural network and neural network with nonlinear classifications and two hidden layers(NN2HL), the accuracy rate reaches 86.67%, which is very satisfying.

1. Introduction

1.1. Speech Recognition

Accent recognition is based on Speech Recognition. Geoffrey Hinton(2012) used Deep Neural Networks(DNNs) to perform GMMs(Gaussian Mixture Models) on a variety of speech recognition benchmarks[1]. Many research groups from big companies also produced their speech recognition systems. Research Group of Baidu(2015) applied High Performance Computing(HPC), resulting in a 7x speedup over their previous system[2]. Microsoft(2016) released a research on conversational speech recognition by systematic use of convolutional and LSTM neural network[3]. Research group of Google(2015) further improved performance of LSTM RNN acoustic models for large vocabulary speech recognition and show that frame stacking and reduced frame rate lead to more accurate models and faster decoding[4].

Muhammad Rizwan(2016) presented the research on a weighted accent classification using multiple words based on ELM and SVM and got the highest accuracy of 77.88%[5]. Buket D. Barkana(2019) researched on standard English and other accents about the vowel and the formant, resulting an accuracy of around 80%[6]. John H.L. Hansen(2010) researched on VOT and used DMT, DMFT, DWTlfr, DWThfr and HMM has a success rate about 70%[7]. Hamid Behravan(2014) used i-Vector based recognition is applied to the Finnish national foreign language certificate (FSD) corpus[8].

In conclusion, most accent classification work is not based on neural networks (More using GMM, HMM and others), and also with an average accuracy rate at about 70%-80%. Our research is to provide...
direct insights into whether there is a simpler way using neural networks and stronger features (like Muti-feature fusion) to have better results.

1.2. Voice Onset Region(VOR)
When we speak an English word, there will be silent period between different syllabus, and VOR starts from the point in time at which vocal-fold vibration starts and ends when the silent part ends and another vocal-fold vibration starts[6].

2. Method
Formants are the result of energy peaks in a more or less narrow zone of the spectrum. And different formant will represent different information. Experiment shows that a vowel can be represented by 3 formants and complex consonant needs five formants to represent. For accent differences, the formant 2 and 3 are most closely related features.

In this section we introduce our work of accent recognition.
The work can be divided into following steps:
1. Form database.
2. Find ways of speech segmentation.
3. Try different ways of input for the accent recognition.
4. Train neural networks to do the classification.
5. Test and comparison.

2.1. The Database

2.1.1. The Word. “The speech accent archive”(last update: 26 April, 2020 accent.gmu.edu/index.php) is used as our database. It contains 2932 samples of people with different English accents speaking the same sentence:

“Please call Stella. Ask her to bring these things with her from the store: Six spoons of fresh snow peas, five thick slabs of blue cheese, and maybe a snack for her brother Bob. We also need a small plastic snake and a big toy frog for the kids. She can scoop these things into three red bags, and we will go meet her Wednesday at the train station.”

The database consists of various accents: American English(AE), Mandarin English, Hindi English, Thai English etc. 30 people with Mandarin accent and 30 speaking in American accent were chosen as the raw material of the database. The word “toy” in their sentence was carefully cut out, and these formed the original database of our own.

2.1.2. The Vowel. The Measure of energy. This work used the method from Hansen et al. (2010) [7] to measure the energy of the audio. Using the physics knowledge of

\[ E \propto A^2 \omega^2 = \dot{x}^2 - x\dot{x} \]  \hspace{1cm} (1)

Teager (1980), Teager and Teager(1983), and Kaiser(1990) formed a way to calculate the energy in any single frequency component.

\[ TEO[x(n)] = x(n)^2 - x(n + 1)x(n - 1) \]  \hspace{1cm} (2)

The energy also can be estimated using the differential TEO, which is defined by Maragos et al.(1993) as follows:

\[ DTEO[x(n)] = \frac{TEO[y(n)] + TEO[y(n+1)]}{2} \]  \hspace{1cm} (3)

where \( y(n) = x(n) - x(n - 1) \).

Using these two energy operators, the Frequency Modulation Component (FMC) and the Amplitude Modulation Component (AMC) of a band-limited amplitude/frequency signal can be extracted. The equations are:
\[ x_f(n) = \arccos \left( 1 - \frac{DTEO[x(n)]}{2TEO[x(n)]} \right) \]  
\[ x_a(n) = \frac{|TEO[x(n)]|}{\sqrt{1 - \cos^2(x_f(n))}} \]  

It’s assumed that the part with the most energy is the vowel. Take the “Mandarin1-toy.wav” in our database as an example:

Figure 1. The energy of "Mandarin1-toy.wav"

Using this method, the energy distribution in the audio can be visualized for further treatment. The extraction of vowel.

Using the method in section 1:

Figure 2. The vowel of "Mandarin1-toy.wav"

It can be seen clearly that the area in the red box contains the most energy compared to others. According to our assumption, this is the vowel we are looking for. The vowel in 60 audios in our database is manually extracted. We believe there are automatic recognition programs, this can be remained as future research.

2.1.3. The Voice Onset Region (VOR). Different accents have spectral difference in their Voice Onset Region (VOR), which is usually known as the consonant part of a word, according to Hansen et al. (2010) [7].

1. The measure of energy. The measurement of energy is exactly the same as the first part in vowel. The identical picture was formed using the method mentioned. However, this time we focused on a whole different area.

2. The extraction of energy. Using the method in section 1:
The area in the red box is the VOR. The 60 VORs in our database was manually extracted.

2.1.4. The Input to the Neural Network.

3. The wav files of the word, vowel and VOR. The MFCCs of the audio was formed and was inputted as features of accents to the neural network.

4. The formant trajectory. Formants indicates the physical movement of one’s vocal cords while pronouncing different words. According to the book “AI 2003: Advances in Artificial Intelligence” [9], the frequency trajectory of formants 2 and 3, as well as the difference between them, can best indicate the feature of different accents.

In our research, we use the Linear Prediction Coding to find out the formants. The LPC is considered extremely useful in a model which typically have a signal source and a random noise source going through a time-varying all-pole filter to the generate the final speech signal. In this case, the random noise is pronounced when the person is speaking a consonant, and the pulse generator can generate a pulse train when the person is speaking a vowel. The time-varying all-pole filter is the airway.

So the speech signal \( s(k) \) can be expressed using function:

\[
s(k) = \sum_{p=1}^{P} a_p s(k - p) + e(k)
\]

where \( P \) is the order of the filter, \( a_p \) is the coefficient of the filter. LPC basically extract the \( a_p \) while knowing the information of \( s(k) \). Using Levinson-Durbin Algorithm, the \( a_p \) can be solved.

Finally, get the roots of the equation using the coefficient \( a_p \). The formants’ frequencies can be calculated using simple mathematical and physical relations between the roots and the frequency.

2.1.5. The Neural Network. After the form of the database, we applied it into the neural network to train. In this paper, two neural networks were used: neural network with linear classification (LNN) and neural network with nonlinear classification and two hidden layers (NN2HL).

5. Neural Network with Linear Classification. This work distinguished the Chinese accent and the USA accent, so, it is also a Binary Neural Network (BNN). A basic BNN is shown in figure 4.
It contains input layer, weight factor and a bias term \( b \), so we have:

\[
    z = \sum_{i=1}^{n} w_i x_i + b \quad (7)
\]

Then it have an activation function, and in this network the logistic function was applied:

\[
    y = \frac{1}{1 + e^{-z}} \quad (8)
\]

The activation function can help us to limit the range of \( y \) about zero to one, which is convenient for us to do further recognition. In that case, the output layer \( y \) was produced.

And for classification, the Mandarin accent is given the target value 0, and gave USA accent target value 1. Then if \( y \) is greater or equals to 0.5, it is classified as belonging to USA accent. And if \( y \) is smaller than 0.5, it is classified as Mandarin accent.

When training, the ANN is trained by adjusting the weights \( w_i \) to make when the feature vectors of Mandarin accent is the applied to the input. And it is the same case for USA accent.

The lost function was the square of the difference between the output \( y \) and its target value \( t \), given a:

\[
    C = \frac{1}{2} (y - t)^2 \quad (9)
\]

Then the learning rate was applied:

\[
    dy = -\eta \frac{dc}{dy} = -\eta(y - t) \quad (10)
\]

However, only \( w_i \) and \( b_i \) can be changed rather than changing \( y \) directly, so we applied backpropagation using gradient descent and finally:

\[
    dw_i = -\eta_w \frac{dc}{dw_i} = -\eta_w x_i y (1 - y)(y - t) \quad (11)
\]

\[
    db = -\eta_b \frac{dc}{db} = -\eta_b y y (1 - y)(y - t) \quad (12)
\]

In that case, we can minimize the cost \( C \) like the figure shows:

![Figure 5. The Cost Function](image)

Then classifying can start by the trained neural network using forward propagation.

6. Neural network with two hidden layers. Although neural network with linear classification is convenient and easy to train, the result of that neural network is not so acceptable, so it is necessary to introduce a more advanced neural network with two hidden layers.

Then layers are assigned, including input layer “a0”, first hidden layer “a1”, second hidden layer “a2” and the output layer “a3”, and the connections between layers were shown in figure 6.
3. Results

In this section, two experimental results based on four different features and two neural networks will be shown. PET is the training classification error probability when the training set is applied. PEG is the generalization classification error probability when the test set is applied. The 0 and 1 on the first column is the labeled value, and the 0 and 1 on the first row is the characterized value.

3.1. The Whole Word

First the whole word was trained by both NN with linear classification and NN with 2 hidden layers. And the result is as follows:

For the neural network with linear classification, the results are shown in Table 1.

Table 1. The accuracy rate of LNN using the whole word

|   | 0 | 1 |
|---|---|---|
| 0 | 5 | 10 |
| 1 | 3 | 12 |

nTrials=30 nErr=13 PEG=0.433

It seemed that it was only a little better than guessing.

For the neural network with nonlinear classification with two hidden layers with eta=0.30 and numRep=2000, the results are shown in table 2.

Table 2. The accuracy rate of NN2HL using the whole word

|   | 0 | 1 |
|---|---|---|
| 0 | 8 | 7 |
| 1 | 7 | 8 |

nTrials=30 nErr=14 PET=0.000 PEG=0.467

It could be seen that it was not even better than training by neural network with linear classification. Therefore, it seemed that training the whole word may not work to recognize the accent.

3.2. VOR

Then the database is applied it into the neural network.

Both NN with linear classification and NN with 2 hidden layers are tried in this section. And the result is as follows:

For the neural network with linear classification with eta= 0.50 and numRep= 1000, the results are shown in table 3.

Table 3. The accuracy rate of LNN using the VOR

|   | 0 | 1 |
|---|---|---|
| 0 | 3 | 12 |
| 1 | 3 | 12 |

nTrials=30 nErr=15 PEG=0.500
It seemed that it doesn’t work even compared with guessing.
For the neural network with nonlinear classification with two hidden layers with eta=0.50 and numRep=1000, the results are shown in table 4.

Table 4. The accuracy rate of NN2HL using the VOR

|             | 0  | 1  |
|-------------|----|----|
|             | 0  | 11 | 4  |
|             | 1  | 5  | 10 |

\[\text{nTrials}=30 \text{ nErr}=9 \text{ PET}=0.000 \text{ PEG}=0.300\]

It seemed that this PEG is at least a presentable result.
And compared with the paper which research on the similar topic, they also have an accuracy of about 70% using Hidden Markov Model(HMM).

3.3. Vowel
Then vowels with two NNs are discussed.
For neural network with linear classification, the results are shown in table 5.

Table 5. The accuracy rate of LNN using the vowel

|             | 0  | 1  |
|-------------|----|----|
|             | 0  | 11 | 4  |
|             | 1  | 5  | 10 |

\[\text{nTrials}=30 \text{ nErr}=16 \text{ PEG}=0.533\]

The result was even worse than guessing.
For neural network with nonlinear classification and two hidden layers, the results were as follows with eta=0.30 and numRep=1200, the results are shown in table 6.

Table 6. The accuracy rate of NN2HL using the vowel

|             | 0  | 1  |
|-------------|----|----|
|             | 0  | 11 | 4  |
|             | 1  | 4  | 11 |

\[\text{nTrials}=30 \text{ nErr}=8 \text{ PET}=0.0009 \text{ PEG}=0.267\]

It seemed that the result is better than VOR.

3.4. Formants
As for formants, three different inputs (f2, f3, and f2-f3) are chosen to figure out which one has the best result.

f2. First the formant 2(f2) in neural network with linear classification was trained and the results are shown in table 7.

Table 7. The accuracy rate of LNN using formant 2

|             | 0  | 1  |
|-------------|----|----|
|             | 0  | 5  | 10 |
|             | 1  | 5  | 10 |

\[\text{nTrials}=30 \text{ nErr}=15 \text{ PEG}=0.500\]

It seemed that it’s accuracy is same as guessing.
For neural network with two hidden layers with $\eta=0.30$ and $numRep=1000$, the results were shown in table 8.

Table 8. The accuracy rate of NN2HL using formant 2

| Trial | 0   | 1   |
|-------|-----|-----|
| nTrials=30 | 15  | 0   |
| nErr=6    | 6   | 9   |
| PET=0.000| PEG=0.200 |

f3. First neural network with linear classifications is used to train, and the results were shown in table 9.

Table 9. The accuracy rate of LNN using formant 3

| Trial | 0   | 1   |
|-------|-----|-----|
| nTrials=30 | 15  | 0   |
| nErr=12  | 12  | 3   |
| PET=0.000| PEG=0.400 |

Although it was better than using formant 2($f_2$) with neural network with linear classification, it was still not a good result.

Then neural network with two hidden layers was used with $\eta=0.30$ and $numRep=1200$ to train, and the results are shown in table 10.

Table 10. The accuracy rate of NN2HL using formant 3

| Trial | 0   | 1   |
|-------|-----|-----|
| nTrials=30 | 15  | 0   |
| nErr=7   | 17  | 8   |
| PET=0.000| PEG=0.233 |

It seemed that although it was a presentable result, the result, however, was worse than $f_2$’s result. Then $f_3$-$f_2$ are researched. For neural network with linear classification, the results were shown in table 11.

Table 11. The accuracy rate of LNN using $f_3$-$f_2$

| Trial | 0   | 1   |
|-------|-----|-----|
| nTrials=30 | 9   | 6   |
| nErr=12  | 6   | 9   |
| PET=0.000| PEG=0.400 |

It was not a good result, then neural network with nonlinear classifications with two hidden layers are tried with $\eta=0.30$ and $numRep=1500$, and the results are shown in table 12.

Table 12. The accuracy rate of NN2HL using $f_3$-$f_2$

| Trial | 0   | 1   |
|-------|-----|-----|
| nTrials=30 | 15  | 0   |
| nErr=5   | 5   | 10  |
| PET=0.000| PEG=0.167 |

Therefore, $f_3$-$f_2$ had the best result of PEG=0.167 and it is better than the whole word, vowel, and VOR.
Concatenation. Then three feature vectors were concatenated as a new input to train the neural network. 100nCCs were still used to train. Using neural network with linear classification, the results are shown in table 13.

Table 13. The accuracy rate of LNN using concatenated feature vector

|      | 0   | 1   |
|------|-----|-----|
| 0    | 11  | 4   |
| 1    | 10  | 5   |

nTrials=30 nErr=14 PET=0.400 PEG=0.467

It was not a good result, so the neural network with nonlinear classifications and two hidden layers with eta=0.30 and numRep=1200 was used, and the results are shown in table 14.

Table 14. The accuracy rate of NN2HL using concatenated feature vector

|      | 0   | 1   |
|------|-----|-----|
| 0    | 15  | 0   |
| 1    | 4   | 11  |

nTrials=30 nErr=4 PET=0.000 PEG=0.133

The table below shows the results:

Table 15. The accuracy rate comparison

| Input Type         | Linear Classification Error | Non-Linear Classification Error |
|--------------------|------------------------------|---------------------------------|
| The Word           | 0.433                        | 0.467                           |
| The VOR            | 0.500                        | 0.300                           |
| The Vowel          | 0.533                        | 0.267                           |
| Formant 2(f2)      | 0.500                        | 0.200                           |
| Formant 3(f3)      | 0.400                        | 0.233                           |
| Formant3 - Formant 2(f3-f2) | 0.400 | 0.167 |
| Concatenation      | 0.467                        | 0.133                           |

4. Discussion

The results show that our work had obvious effect on the classification of the accents. If the input is the MFCC of the word, the Error rate is 0.433 and 0.467 respectively for linear and non-linear classification. Compared to the 0.5 error rate of guesswork, it does not see much improvement to that. So simply input the MFCC of the word itself produce unsatisfactory results. This indicates that it cannot be treated as a typical feature for accent classification.

The linear classification rate of VOR is about 0.5, which is exactly the possibility of guesswork. The computer binary linear classification network failed in this case. One possible explanation is the VOR are simply pronounced in different accents but still the same word, the hyper planes’ resolution power isn’t strong enough for such similar inputs. The non-linear classification, which is the network with two hidden layers, preformed much better. The classification error decreased to 0.300. In Hansen et al.(2010)[7], the Hidden Markov Model classifies the accents with an accuracy rate of 71.67%. Compared to their results, the neural network has the ability to produce similar results while using much simpler coding.

For the vowels, the linear classification results are even worse than guesswork. The non-linear
classification performs better with an error rate of 0.267, similar to that of VOT. However, because the vowel is the part where contains most information of the word, most of the features should be similar to the whole word’s MFCC. The reason why it can perform much better than the word might be that during the manually extraction process of the vowel, we also cut off the silence and noise part in the audio, which reduces the distractions from these unwanted parts.

For the formant analysis, the formant 2 and formant 3 act similarly with a success rate of 80.00% and 76.67% respectively under the non-linear classification. The f3-f2 has an amazing accuracy rate at 83.33%. The formants can best represent the movement of the physical vocal organ, the vocal cord, which is the strongest feature while doing accent classifications. This might because that different people have different formant frequencies due to gender differences and other physical differences of the vocal cord. Using the differences between the formant 2 and 3, the base differences of the frequencies are eliminated.

In the end, all the input vectors are concatenated to form a big vector as a new input. The accuracy rate is further improved to 86.67%.

However, there are several factors that may influence the results. Firstly, it’s noteworthy that the audio files in the original database (the speech accent archive) might not be recorded strictly under a quiet environment, nor recorded using the same microphone, for the energy of the audio varies. Secondly, for the reason that it only offers the online play service, it requires to re-record it as database, which can introduce extra noise. Thirdly, the manually extracted VORs and vowels might not be so accurate due to objective reasons, and the automatic extraction programs are essential for real-life application. Finally, for the neural network, it tried numerous combinations of learning rate and number of repetitions to make sure it can quickly convergent without falling into the local minimum. But still, it’s possible that some bad results were produced due to the network itself.

There are also limitations of our project. Regarding those limitations, some future research remains to be done.

1. Automatic extraction of VORs and vowels is necessary if the method is put into real-life situations.
2. More consonants and more vowels should be examined.
3. Measures should be taken to further reduce the background noise of the audio.
4. Using multiple neural networks to do the “voting process” of the classification can be considered.

However, comparing to the accuracy rate from previous studies, which is around 70-80%, we have made some noticeable improvements at about 10-20%.

5. Conclusion
Except from the MFCC of the word itself, all the others (the VOR, vowel, formants) can represent the features of a certain accent. Among them all, the f3-f2 tops the list with an accuracy rate of 83.33%. Compared to previous results in the accent classification, simpler networks and produced better results were used. By concatenating these feature vectors together, the results can be improved to an accuracy rate of 86.67%, which is one of the best results comparing to previous research.

This work indicates that accent recognition can be achieved through various and relatively easy ways, which can be a reference for future researchers when it comes to accent recognition. Also, the results can possibly be repeated on other words, and this can thus provide more valuable features for the purpose of accent classification.

Acknowledgement
Zhewen Lou and Yitian Ren contribute equally to this work and should be considered co-first authors.

References
[1] A Senior, V Vanhoucke, P Nguyen, T Sainath, et al. Deep neural networks for acoustic modeling in speech recognition. IEEE Signal processing magazine, 2012.
[2] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, et al. Deep speech 2: End-to-end speech recognition in english and mandarin. In Pro-ceedings of
the 33rd International Conference on International Conference on Machine Learning - Volume 48, ICML’16, page 173–182. JMLR.org, 2016.

[3] Wayne Xiong, Jasha Droppo, Xuedong Huang, Frank Seide, Mike Seltzer, Andreas Stolcke, Dong Yu, and Geoffrey Zweig. Achieving human parity in conversational speech recognition. arXiv preprint arXiv:1610.05256, 2016.

[4] Hasim Sak, Andrew Senior, Kanishka Rao, and Françoise Beaufays. Fast and accurate recurrent neural network acoustic models for speech recognition, 2015.

[5] Muhammad Rizwan and David V. Anderson. A weighted accent classification using multiple words. Neurocomputing, 277:120 – 128, 2018. Hierarchical Extreme Learning Machines.

[6] Buket D. Barkana and Aarchi Patel. Analysis of vowel production in mandarin/hindi/american-accented english for accent recognition systems. Applied Acoustics, 162:107203, 2020.

[7] John H.L. Hansen, Sharmistha S. Gray, and Wooil Kim. Automatic voice onset time detection for unvoiced stops (/p/,/t/,/k/) with application to accent classification. Speech Communication, 52(10):777 – 789, 2010.

[8] Hamid Behravan, Ville Hautama’ki, and Tomi Kinnunen. Factors affecting i-vector based foreign accent recognition: A case study in spoken finnish. Speech Communication, 66:118 – 129, 2015.

[9] Koji Murakami, Kenji Araki, Makoto Hiroshige, and Koji Tochinai. Effectiveness of a direct speech transform method using inductive learning from laryngectomee speech to normal speech. In Tama’s (Tom) Domonkos Gedeon and Lance Chun Che Fung, editors, AI 2003: Advances in Artificial Intelligence, pages 686–698, Berlin, Heidelberg, 2003. Springer Berlin Heidelberg.