Automated Recognition of Palestinian Accents Using Acoustic Analysis

Tariq Odeh - 1190699, Shahd AbuDaghash - 1191448, Tala Alswaitti - 1191068
Department of Electrical & Computer Engineering, Birzeit University
Ramallah, Palestine
Supervised by: Dr. Abualsoud Hanani

***Abstract*—This project develops and evaluates an acoustic-based Palestinian regional accent recognition system, targeting accents from four distinct regions: Jerusalem, Nablus, Hebron, and Ramallah. Leveraging speech data, the system extracts Mel-frequency cepstral coefficients (MFCCs) to classify the speaker's accent. Implemented in Python, the system was trained on a dataset of .wav files specific to each region and tested on a separate dataset to evaluate performance. The Random Forest classifier demonstrated an accuracy of 85%, effectively distinguishing between the different Palestinian accents. Our evaluation, based on percentage accuracy, highlights the system's capability to accurately identify these regional accents. This work lays a foundation for further studies in regional accent recognition and contributes to advancements in speech processing and language identification.**

***Index Terms*—MFCCs, Palestinian accents, accent recognition, speech processing, Random Forest.**

I. Introduction

THE ability to recognize regional accents within a language can be a valuable tool in fields such as linguistics, sociolinguistics, and even for applications in speech recognition and language learning. This project focuses on developing and evaluating a recognition system for identifying Palestinian regional accents. Palestinian accents are diverse and include distinct sub-accents from areas such as Jerusalem, Nablus, Hebron, and Ramallah. Each of these accents carries unique phonetic characteristics that can be captured and analyzed through acoustic features.

The primary goal of this project is to create a system that can accurately classify a given speech segment into one of the four specified Palestinian accents. To achieve this, audio recordings representing each accent will be gathered, Mel-frequency cepstral coefficients (MFCCs) will be extracted, and a Random Forest classifier will be trained on these features. The model will then be tested on a separate dataset and evaluated using metrics like accuracy, classification reports, and confusion matrices.

II. Background and Related Work

The study of accent recognition within the domain of speech processing has emerged as a pivotal area of research, driven by the need to enhance the robustness and accuracy of Automatic Speech Recognition (ASR) systems. Accents, reflective of regional and cultural linguistic variations, present significant challenges to ASR, necessitating the development of specialized methodologies for their detection, identification, and classification.

Early endeavors in accent detection laid foundational frameworks for subsequent research. Zheng et al. (2005) introduced pioneering methodologies employing Maximum A Posteriori (MAP) and Maximum Likelihood Linear Regression (MLLR) techniques, resulting in notable reductions in character error rates. Building upon this groundwork, Wu et al. (2009) investigated the fusion of Gaussian Mixture Models (GMM), Linear Discriminant Analysis (LDA), and Support Vector Machines (SVM), yielding substantial improvements in ASR performance [1][2].

Advancements in accent processing have been marked by the incorporation of novel features and techniques. Hansen et al. (2010) introduced Voice Onset Time (VOT) analysis as a novel approach to accent identification, demonstrating high accuracy rates [3]. Sam et al. (2011) explored modulation spectrum features for accent differentiation, while DeMarco et al. (2012) proposed iterative discriminative algorithms, such as the i-vector method, showcasing superior performance in accent classification tasks [4].

Modeling techniques have evolved to address the intricacies of accent variability. Safavi et al. (2012) investigated the efficacy of Gaussian Mixture Model-Universal Background Model (GMM-UBM) systems, achieving remarkable accuracy rates in accent identification [1]. Lazaridis et al. (2014) extended these efforts by exploring UBM-MAP and i-vector methods, with the TV-SVM system demonstrating superior performance, highlighting the potential of advanced techniques in accent processing [5][6].

Prosodic features have emerged as influential factors in accent detection and classification. Lazaridis et al. (2014) observed a correlation between the number of syllables and classification accuracy, while Lacheret et al. (2014) emphasized the importance of manual and automatic data processing in accent annotation.Recent advancements have witnessed the fusion of i-vector and Phonotactic techniques, as demonstrated by Najafian et al. (2016), resulting in remarkable accuracy rates exceeding 84.87%. These state-of-the-art methodologies integrate innovative techniques and features, marking significant strides in addressing the challenges posed by accent variability in speech processing [7].

Over the past decade, accent recognition research has proliferated, aiming to enhance ASR performance through innovative methodologies. By leveraging a diverse array of

techniques and features, researchers have made notable advancements, laying the groundwork for further progress in this critical area of speech processing.

III. Methodology

A. Front-end processing

During the feature extraction stage, the continuous speech signal (sampling rate equals 44.1 kHz)is converted into a sequence of acoustic feature vectors. In this work, we used Mel-frequency cepstral coefficients (MFCCs) to represent the speech signal. The extraction process involves loading the audio file while preserving its original sample rate using the `librosa.load` function. The signal is then processed to compute the MFCCs using `librosa.feature.mfcc`, which internally performs several critical steps [8]. Figure 1 shows the block disgram of MFCC block diagram.

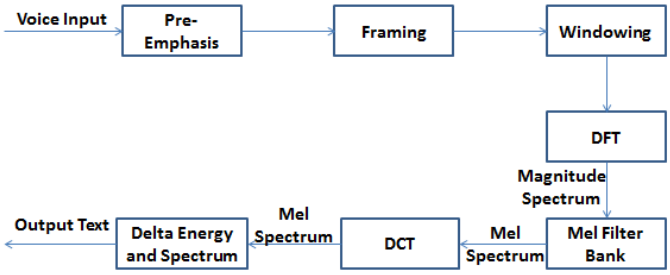

Fig. 1. MFCC Block Diagram.

Initially, the audio signal is divided into overlapping frames with a specific frame length and hop length. Each frame is windowed using a Hann window

$$w(h) = 0.5 - 0.5\cos\left(\frac{2\pi n}{M-1}\right),$$

where M is the number of samples in each frame, to minimize discontinuities at the edges.

These windowed frames are transformed into the frequency domain using the Discrete Fourier Transform (DFT), represented as:

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j\frac{2\pi}{N}kn},$$

where $x(n)$ is the signal, N is the number of points in the DFT, and k is the frequency bin.

The resulting spectrum is filtered using a Mel-scale filter bank to approximate the human ear's response to different frequencies. The Mel scale is defined by:

$$m = 2595 \cdot \log_{10}\left(1 + \frac{f}{700}\right),$$

where f is the frequency in Hz, and m is the Mel frequency.

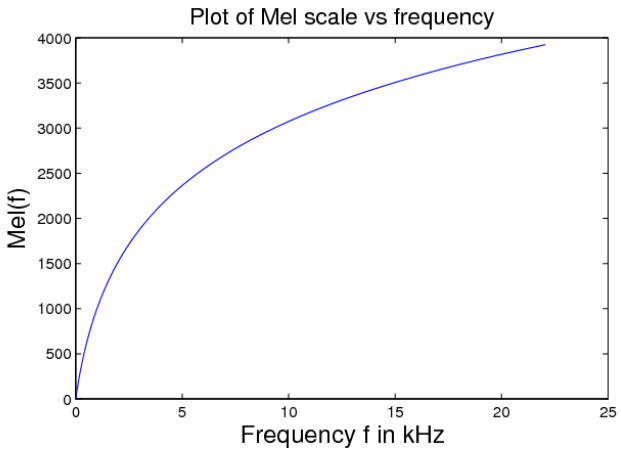

Fig. 2. Diagram of the Mel-scale filter bank.

These filterbank energies are then logarithmically compressed to better represent the perceived loudness levels. The final step involves applying the Discrete Cosine Transform (DCT) to these log energies to obtain the MFCCs, which decorrelate the filterbank coefficients:

$$\text{MFCC}(k) = \sum_{m=0}^{M-1} \log(E(k)) \cdot \cos\left[k\,(m-0.5)\,\frac{\pi}{M}\right],$$

where $E(k)$ is the log energy of the m-th filter, M is the total number of filters, and k is the MFCC index.

The MFCCs are computed for each frame, and to create a more compact representation, the mean of each MFCC across all frames is calculated, resulting in a 20-dimensional feature vector for each audio file. This feature vector encapsulates the essential acoustic characteristics of the speech signal, making it suitable for subsequent stages of speech processing and classification.

B. Back-end processing

The back-end approach for our accent recognition system employs a structured machine learning pipeline to classify regional Palestinian accents using extracted acoustic features. This subsection details the creation, training, and evaluation of the model.

1) Machine Learning Pipeline: To streamline the modeling process, we use a pipeline that integrates both preprocessing and classification steps. This ensures that each step is executed in sequence and automates the workflow.

a) Standardization: The first step in the pipeline is standardization, which adjusts the features to have a mean of zero and a standard deviation of one. This ensures that all features contribute equally to the model's learning process and improves the performance and convergence of the classifier.

b) Classification: The classification step is handled by a Random Forest classifier, an ensemble learning method that constructs multiple decision trees during training. This classifier is chosen for its balance between accuracy and computational efficiency. The model parameters are carefully selected to optimize performance:

- n_estimators=200: The model builds 200 decision trees.

- max_depth=10: The maximum depth of each tree is restricted to 10 levels to prevent over-fitting.
- random_state=42: A fixed random seed ensures reproducibility of the results.

2) Training the Model: The model is trained using the extracted features from the training data and their corresponding labels. During training, the standardization step computes the necessary statistics (mean and standard deviation) from the training data to transform it appropriately. The transformed data is then used by the Random Forest classifier to learn the patterns that distinguish between different accents.

3) Predicting and Evaluating: Once trained, the model is used to predict the accents of the testing dataset. These predictions are then compared to the true labels to assess the model's performance.

IV. Experiments and Results

The experiment involves extracting the acoustic features from the training data using MFCC, and using it in a Random Forest classifier that would construct decision trees during training. Then, test data is used to evaluate the system.

We are using a balanced speech dataset containing `.wav` files for training and testing. The dataset encompasses four distinct Palestinian Arabic sub-accents: Jerusalem, Nablus, Hebron, and Ramallah. For training purposes, the dataset provides 10 `.wav` files for each sub-accent, totaling 40 files. Similarly, for testing, the dataset includes 5 `.wav` files for each sub-accent, resulting in a total of 20 files. The sampling rate of the microphone used to collect the data is 44.1kHz.

After feeding the test data to the model for evaluation, the overall accuracy was observed. Then, the precision, recall, and f1-score were studied to evaluate the model's learning. Two important parameters when using MFCC are the frame length `n_fft`, and hop length `hop_length`. First, we tried the default values in MFCC with `n_fft = 2048 samples`, and `hop_length = 512 samples`. This means the frame size equals `46.44ms` and the frame shift equals `11.6ms`. This case showed `70%` accuracy.

Then, the accuracy was investigated for three cases, when:

1) `n_fft = 4096 samples`, and `hop_length = 1024 samples`
2) `n_fft = 8192 samples`, and `hop_length = 2048 samples`
3) `n_fft = 16384 samples`, and `hop_length = 4096 samples`

Using the third case where the frame length `n_fft = 16384 samples` which is approximately equals `371.5 ms`, and a hop length `hop_length = 4096 samples`, which is approximately `92.9 ms`, gave the best accuracy of **`85%`**. This is most likely due to the nature of the dataset, where using a wider frame gives better frequency resolution and hence, better results. Having a wider frame means that the sound was divided to larger parts (words or parts of words). This impacts the coefficients resulting from the MFCC.

Figure 3 shows the accuracy results. Since using a frame size equals 16384 samples gives the best accuracy, we moved on investigating its results.

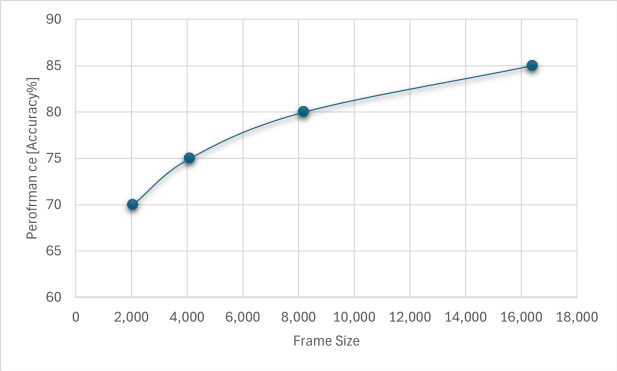

Fig. 3. Frame Size vs. Accuracy

The following performance metrics were investigated:

- **Precision**: measures how accurate the model's positive predictions are.
- **Recall**: measures how well the model identifies all the relevant cases.
- **F1-score**: a harmonic mean that combines both precision and recall, providing a balanced view of the model's performance.

Figure 4 shows the classification report that contains calculating the performance metrics for each region.

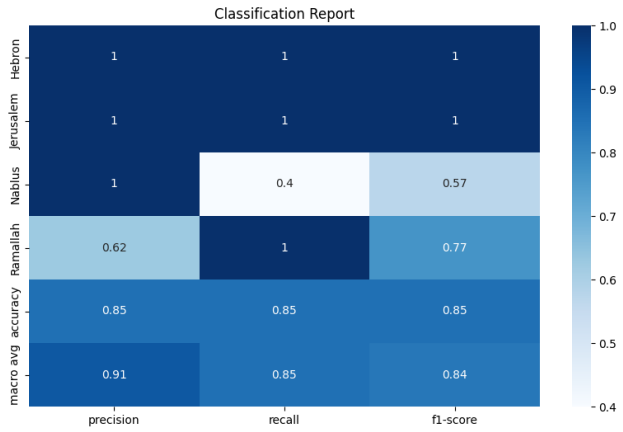

Fig. 4. Classification Report

There are a few observations made based on figure 4 results. First, the recall values show that the model succeeded to identify Ramallah-Reef, Hebron and Jerusalem test data. However, the lowest recall value is for Nablus, indicating that Nablus testing data was classified to other regions.

Looking at the precision, we see that none of the data was wrongly predicted as Hebron, Jerusalem or Nablus, since the precision is 1. On the other hand, Ramallah prediction was made for some of Hebron, Jerusalem, or Nablus data.

Overall, the accuracy of the performance metrics was 85% each, which is acceptable given the small data size.

V. Conclusion and future work

This project successfully developed an acoustic-based Palestinian regional accent recognition system targeting four regions: Jerusalem, Nablus, Hebron, and Ramallah, achieving an accuracy of 85% using MFCCs and a Random Forest classifier. Despite the promising results, future work should focus on expanding the dataset, adding more accents, and employing advanced machine learning techniques like CNNs or RNNs to enhance performance. User feedback integration is also crucial for further improvements and broader application. These enhancements will contribute significantly to the fields of speech processing and linguistic research.

VI. Partners participation tasks

The project was divided equally on all group members. For writing the report, Tariq was responsible for the Abstract and the Background parts, Tala was responsible for the Introduction and the Conclusion, while Shahd was responsible for the Methodology. The Experiments and Results part was done by all group members.

References

[1] Douglas A. Reynolds, Thomas F. Quatieri, and Robert B. Dunn. Speaker verification using adapted gaussian mixture models. *Digital Signal Processing*, 10(1):19–41, 2000.

[2] M.A. Zissman, T.P. Gleason, D.M. Rekart, and B.L. Losiewicz. Automatic dialect identification of extemporaneous conversational, latin american spanish speech. In *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, volume 2, pages 777–780 vol. 2, 1996.

[3] Julia Hirschberg and Fadi Biadsy. Automatic dialect and accent recognition and its application to speech recognition. 2011.

[4] lorita Damayanti and Amalia Zahra. Accent detection task to classify accented and non-accented speech. *International Journal of Recent Technology and Engineering (IJRTE)*, 8:8597–8600, 09 2019.

[5] Stephen Taylor, Abualsoud Hanani, Hanna Basha, and yasmeen Sharaf. Palestinian arabic regional accent recognition. 10 2015.

[6] William Campbell, Douglas Sturim, and Douglas Reynolds. Support vector machines using gmm supervectors for speaker verification. *Signal Processing Letters, IEEE*, 13:308 – 311, 06 2006.

[7] Abualsoud Hanani, Martin Russell, and Michael Carey. Computer and human recognition of regional accents of british english. pages 729–732, 08 2011.

[8] M. Najafian. *Automatic Speech Recognition of Regional Accents*. PhD thesis, University of Birmingham, 2016.

VII. Appendix

Project Code on github