Development of an Acoustic Regional Accent Recognition System for Palestinian Accents

Department of Electrical and Computer Engineering, Birzeit University

Spoken Language Processing

Saja Shareef[1], *Shereen Ibdah*[2]

1200901@*student.birzeit.edu*[1], 1200373@*student.birzeit.edu*[2]

***Abstract*—This project explores the development of an acoustic regional accent recognition system. It's made to identify four Palestinian accents. They are Jerusalem, Nablus, Hebron, and Ramallah. The study uses acoustic features from speech, like Mel Frequency Cepstral Coefficients (MFCCs). It uses both traditional machine learning and advanced neural networks. They classify short speech segments by regional dialects. The research aims to help the field of spoken language processing. It does this by addressing the challenges of recognizing accents. These challenges arise in a diverse region.**

Keywords Spoken language identification, Accent recognition, Speech recognition, acoustic modeling, Acoustic features.

I. Introduction

The sound wave contains important information by which the age, gender, and accent of the speaker can be determined. In spoken language processing, accent recognition is crucial. It's important in many applications. These include speech recognition, speaker verification, and linguistic research. Our project focuses on developing a model. It will recognize Palestinian regional accents from short speech segments. Palestinian accents vary across regions, specifically Jerusalem, Nablus, Hebron, and Ramallah. This project aims to create an accent recognition system. It can distinguish between these four accents using extracted acoustic features.

Arabic is considered a common language and is usually the mother tongue in many countries, especially the Levant, Palestine, Syria, Jordan, and Lebanon. But at the level of Palestine alone, the Arabic language differs in its pronunciation based on the society and environment in which the human grows up, and the difference in accents comes from the use of words or phrases that are unique to a community. For example, the accent of speakers from the city of Salfit clearly differs from that of speakers from Hebron. Hence the effect of different accents on the performance of automatic speech recognition (ASR). Accented pronunciation variability is one of the key elements that deteriorates the accuracy of automatic speech recognition (ASR).

This paper shows how to distinguish individual Arabic accents. They are within four regionals: Jerusalem, Hebron, Nablus, and rural Ramallah. An accuracy of 70% is obtained using a support vector machine-based system. The remainder of this paper is organized as follows: First, the data set used for analysis is described and then the methods of, SVMs and RF, KNN are briefly presented. Next, experiments and results are presented and discussed. Finally, conclusions and future directions are included.

II. Background/Related Work

Over the years, researchers have explored several approaches for developing speech-based systems. Researchers have divided the best accent recognition approaches into two major classes. These are: phonotactic-based and acoustic-based.

In an acoustic-based approach (ALID), we extract speech features from the raw signal. We rely on sound-based characteristics. These techniques include Linear Predictive Coding (LPC). They also include Linear Prediction Cepstral Coefficients (LPCC) and Perceptual Linear Predictive (PLP). They also include Mel-Frequency Cepstral Coefficients (MFCC) and others. They can produce them. Afterward, we extract x-vectors or i-vectors. In contrast, PLID systems are phonetic-based. They use the phonetic information in the speech signal. Phonetic methods use regular phone recognizers. They tokenize running speech into a sequence of sound units, mostly phones. The sounds are then modeled using language models, like n-grams and RNNLMs. Two phonetic LID approaches exist. The first is phoneme recognition followed by language modeling (PRLM). The second is parallel phone recognition followed by language modeling (PPRLM). PPRLM typically performs better than PRLM. This is because of the extra robustness of having many sets of phonotactic models. [6]

Several approaches can be used to extract features from acoustic signals. These include probabilistic, spectral, model-based, transform-based, and pattern recognition methods. Among the above features, extraction approaches and pattern

recognition methods are adopted widely. We've decided to use the Mel Frequency Cepstral Coefficient (MFCC) for feature extraction.

Several methods are adopted for accent-based speech recognition in ASR. Researchers used conventional statistical methods like HMM and GMM and few researchers recently relied on artificial neural networks and deep neural networks for DID to get better results. It is also noted that very few researchers used unsupervised learning techniques by using autoencoders for dialect-based speech recognition. [4]

An effective approach to distinguishing the accent of an utterance is training a GMM for each accent. We compare the likelihood achieved by each GMM for an utterance. We find out the accent uses training data from each accent to build a GMM for that accent. So, we will have five GMMs for five Farsi accents. Then, for a test utterance, we select the accent whose GMM makes the maximum likelihood. [1]

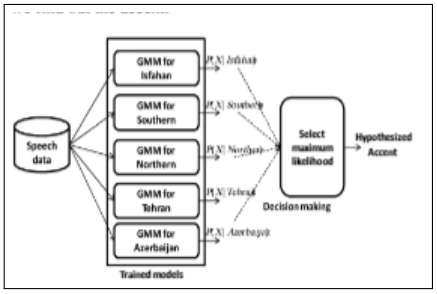

Fig. 1: The block diagram of using GMM based acoustic models followed by ML algorithm

III. Methodology (System Description)

Data Collection The dataset comprised a training set with 10 audio samples for each of the four regions: Hebron, Jerusalem, Nablus, and Ramallah Reef, with varying sizes. The testing dataset consisted of 5 audio samples per region, categorized similarly into the same four regions, with the audio files located in the voices test directory.

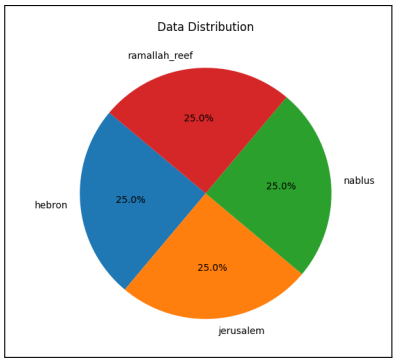

Fig. 2: Pie Charts diagram for data distributions

Features Extraction The "librosa" library extracts acoustic features from audio files. It captures both the spectral and temporal traits of the speech signal. The process computes 24 Mel-frequency Cepstral Coefficients (MFCCs). It also calculates their first-order and second-order delta coefficients. It also calculates RMS energy and its deltas. We calculate these features over audio signal frames with a hop length of 512 and an FFT size of 2048. The mean values of these coefficients are then joined into a single feature vector. This results in a 75-dimensional representation for each audio file. The set has many features. It includes MFCCs, deltas, and RMS energy. They represent the audio well for classification.

Data Preparation The features and labels are prepared as follows: First, we've encoded the labels (region names) as numbers using Label Encoder. Then, we've standardized the features using Standard Scaler to have zero mean and unit variance.

SVM Model The first model used was the Support Vector Machine (SVM). It's a powerful and common machine-learning algorithm. It had the hyperparameter Kernel set to Linear. Class Weight: Balanced to handle any. The SVM aims to find the hyperplane that maximizes the margin between classes. We choose the linear kernel for its simplicity. It accurately processes data that a single line can separate. The 'balanced' class weight parameter helps the model handle imbalanced datasets. The general steps for an SVM with a linear kernel are to find the optimal values of $\mathbf{w}$, b, and α_i, and use the decision function $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$ to classify new data points.

Random Forest Model The second model employed was the Random Forest, a popular ensemble learning technique in machine learning. Random Forest combines multiple decision trees to create a robust and accurate predictive model. It mitigates overfitting and provides feature importance scores, making it suitable for various classifications, The number of estimators and n-estimators is set to 100, specifying the count of trees in the forest. More trees generally improve the model by reducing variance. The random state is set to 42, ensuring reproducible results by producing the same data split across different runs.

KNN Model K-Nearest Neighbors (K-NN) is an alternative classifier. It is used alongside Support Vector Machine (SVM) and Random Forest. It relies on the distances between feature vectors to classify new audio. This is based on their similarity to existing labeled samples. The model uses three nearest neighbors and it's trained on the training data. However, it has low accuracy.

IV. Experiments and Results

The project aimed to identify the origin of audio samples (Hebron, Jerusalem, Nablus, Ramallah Reef) using machine learning. We explored three classifiers: Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Random Forest. All classifiers used MFCCs, their changes (deltas), and RMS energy as features. We've trained each model on a designated dataset and evaluated performance on a separate test set, focusing on metrics like accuracy, precision,

recall, and F1-score to assess their effectiveness in audio classification.

We used to measure or evaluate our results through several evaluation metrics. The **Classification Report** includes precision, recall, F1-score, and support for each class (region). **Precision** is the ratio of correctly predicted positive observations to the total predicted positives. **Recall** is the ratio of correctly predicted positive observations to all observations in the actual class. The **F1-Score** is the weighted average of precision and recall, providing a balance between the two. **Support** refers to the number of actual occurrences of each class in the test set. The **Accuracy Score** is the ratio of correctly predicted samples to the total number of samples, providing an overall indication of the model's performance. A **Confusion Matrix** is a tool that shows how well a classification model performs. It compares true labels to predicted labels. It shows the number of correct and incorrect predictions for each class, helping to identify where the model is making errors [5].

Also, we visualized our classification model's performance. We used a confusion matrix plot. This plot helps in identifying the areas where the model performs well and where it needs improvement. The confusion matrix compares the true and predicted labels. It shows a detailed breakdown of correct and incorrect predictions for each class.

Our first model uses Mel-frequency cepstral coefficients (MFCC) for feature extraction and a Support Vector Machine (SVM) for classification. This approach involves loading the audio data. Then, we extract features that capture the audio's frequency and spectral traits. Figure 3 illustrates this process.

```
Processing region: hebron
* * * * * * * * * *

Processing region: jerusalem
* * * * * * * * * *

Processing region: nablus
* * * * * * * * * *

Processing region: ramallah_reef
* * * * * * * * * *

Processing region: hebron
* * * * *

Processing region: jerusalem
* * * * *

Processing region: nablus
* * * * *

Processing region: ramallah_reef
* * * * *
```

Fig. 3: loaded Data And Extraction

After extracting features, the model trains to classify the audio data. The resulting classification report, presented in Figure 4, details the performance metrics, providing insights into the model's effectiveness. In this case, it's achieved an accuracy of 70%.

```
Classification Report:
                precision    recall  f1-score   support

       hebron       1.00      0.80      0.89         5
    jerusalem       0.67      0.80      0.73         5
       nablus       0.60      0.60      0.60         5
ramallah_reef       0.60      0.60      0.60         5

     accuracy                           0.70        20
    macro avg       0.72      0.70      0.70        20
 weighted avg       0.72      0.70      0.70        20

Accuracy: 70.00%
```

Fig. 4: Classification Report (SVM classifier)

The confusion matrix:

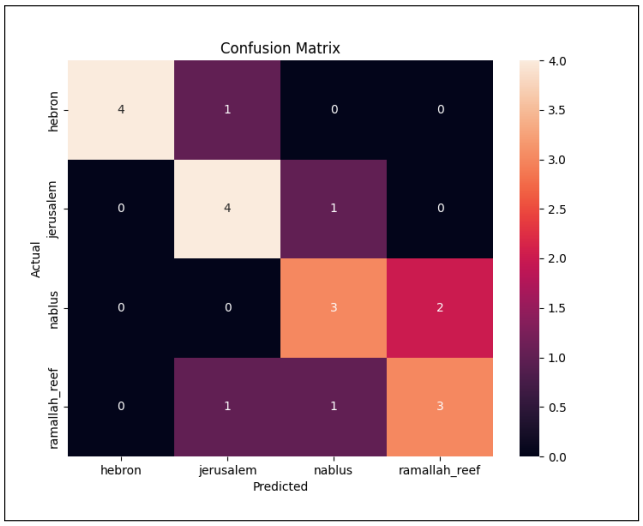

Fig. 5: confusion matrix (SVM classifier)

The second model was explored in this experiment. It uses a K-Nearest Neighbors (KNN) classifier for audio classification. KNN is a non-parametric machine learning approach that classifies data points based on their proximity to labeled data points within the training set [3]. In this context, the audio features extracted using MFCCs serve as the data points. The KNN classifier calculates the distances between its features and the features of the labeled training data when a new audio sample is introduced. The new sample is then assigned the most common label among its K closest neighbors in the training data.

Evalution for KNN
Classification Report:

	precision	recall	f1-score	support
hebron	1.00	1.00	1.00	5
jerusalem	0.33	0.60	0.43	5
nablus	0.40	0.40	0.40	5
ramallah_reef	0.00	0.00	0.00	5
accuracy			0.50	20
macro avg	0.43	0.50	0.46	20
weighted avg	0.43	0.50	0.46	20

Accuracy: 50.00%

Fig. 6: Classification Report (Knn classifier)

The confusion matrix:

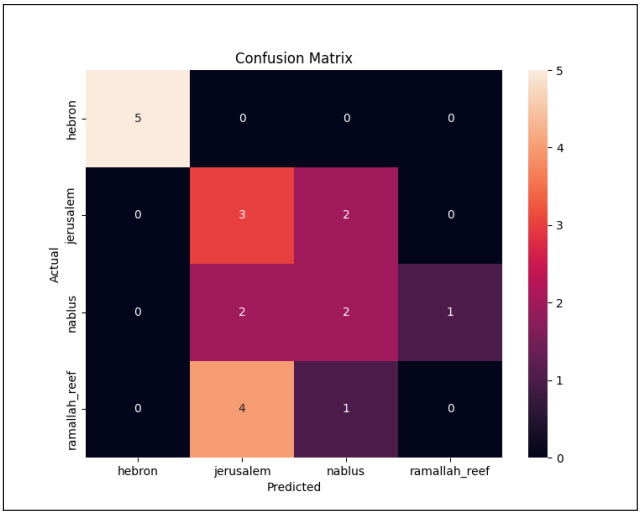

Fig. 7: confusion matrix (Knn classifier)

In this experiment, the KNN model achieved an accuracy of 50% on the test data set. This accuracy suggests that KNN might not be the best choice for this audio task.

The third model explored in this experiment employs a Random Forest classifier for audio data classification. Random Forest is an ensemble learning technique that combines multiple decision trees to enhance overall classification performance [2]. Similar to the previous models, the features used for training the Random Forest classifier are extracted from the audio data using Mel-Frequency Cepstral Coefficients (MFCCs). Each decision tree within the forest is trained on a random subset of these features and data points drawn from the training set. When a new audio sample is introduced, it's passed through all the trees in the forest, and the most frequent prediction among the trees is assigned as the final classification.

Evalution for Random Forest
Classification Report:

	precision	recall	f1-score	support
hebron	0.33	0.20	0.25	5
jerusalem	0.67	0.40	0.50	5
nablus	0.25	0.40	0.31	5
ramallah_reef	0.50	0.60	0.55	5
accuracy			0.40	20
macro avg	0.44	0.40	0.40	20
weighted avg	0.44	0.40	0.40	20

Accuracy: 40.00%

Random Forest Model Accuracy: 40.00%

Fig. 8: Classification Report (Random Forest classifier)

In this experiment, the Random Forest model achieved an accuracy of 40% on the test data set.
The confusion matrix:

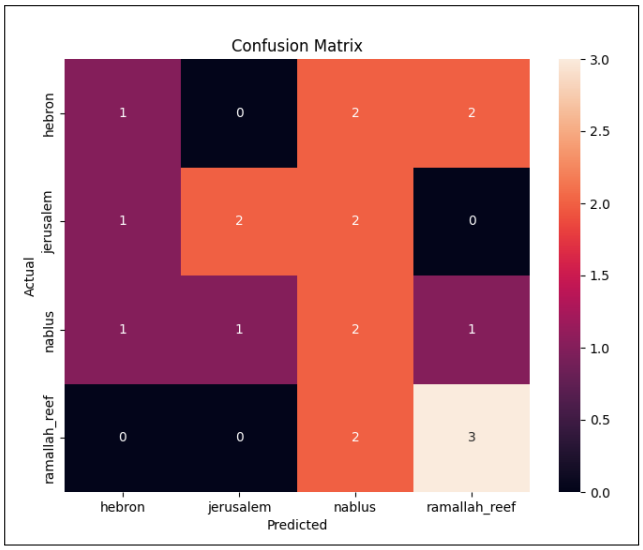

Fig. 9: confusion matrix (Random Forest classifier)

The experiment used three models. They all used Mel-frequency cepstral coefficients (MFCCs) to extract features from the audio data (Figure 3). The first model used a Support Vector Machine (SVM) classifier. It achieved an accuracy of 70% (Figure 4). This suggests a strong ability to distinguish between audio classes. The second model used a K-Nearest Neighbors (KNN) classifier. It had a 50% accuracy (Figure 6). This indicates a lower effectiveness compared to the SVM model. Finally, the third model implemented a Random Forest classifier, reaching an accuracy of 40% (Figure 8). While offering an ensemble approach, Random Forest performed similarly to KNN and might not be the optimal choice for this specific audio classification task.

V. Conclusion and Future Work

The project explored using machine learning to classify audio by their origin in Palestine. We investigated three

classification algorithms. They are Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Random Forest. All models used MFCCs and their deltas. They also used RMS energy as an audio feature. The experiment achieved promising results. The SVM model had the highest accuracy, at 70%, on the test dataset. But, further exploration is necessary to improve performance and address limitations.

In the future, we plan to address the data shortage by adding more speakers to the database. Additionally, we aim to explore how Palestinian accents and Arabic dialects affect Arabic ASR.

VI. Partners participation tasks

Everyone on the team worked hard on this project, and that's why it turned out well.

References

[1] Alizadeh, M., Pishro-Nik, H., and Bayesteh, A. A classifier combination approach for farsi accents recognition. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*.

[2] Analytics Vidhya. Understanding random forest.

[3] IBM. k-nearest neighbors (knn). IBM.

[4] Patil, H. A., and Kumar, A. Accent based speech recognition: A critical overview. *International Journal of Speech Technology*.

[5] Prasanna, C. Classification report explained — precision, recall, accuracy, macro average, and weighted average.

[6] Shakeri, S. M. H., Gholami, O., and Blaabjerg, F. Performance analysis of grid-forming inverters with different control strategies in weak grids. *Computers & Electrical Engineering 106*, 108569.

VII. Appendix

https://drive.google.com/drive/folders/1HQBJxt5dyU4sNkVazqaoFTck-_kQMNw1?usp=sharing