Palestinian Regional Accent Recognition Using Machine Learning Models

Spoken Language Processing - ENCS5344

Electrical and Computer Engineering Department
Birzeit University
Ramallah, Palestine
Amal Butmah [1], *Layan Shoukri* [2], *Yazeed Hamdan* [3]
1200623@student.birzeit.edu, 1201225@student.birzeit.edu, 1201133@student.birzeit.edu

Abstract

There are many accents and sounds in Palestinian societies, as they were divided in our study into distinct accents from four different Palestinian regions, i.e. Jerusalem, Hebron, Nablus, and Ramallah. In this study, we seek to develop a system that automatically recognizes and identifies the accents of the speakers among these four regions. We implemented three techniques and adopted several sounds features to achieve this goal. namely, Gaussian Mixture Model – Support Vector Machines (GMM-SVM), Support Vector Machines (SVM) both with Mel-frequency cepstral coefficients (MFCCs) and contrast features, and K-Nearest Neighbors (KNN) with (MFCCs) feature. These systems were trained and tested on small dataset containing speech from about 10 speakers for each accent. Then, the system performance measures were evaluated, which results in both SVM and GMM-SVM systems with accuracies 75.0% outperform the KNN system with accuracy 65.0%. there are other performance measures were evaluated like Precision, Recall, F1-score, and the confusion matrix for each model.

1. Introduction

Accents, which represent social, cultural, and geographic origins, are essential components of linguistic identity. The Palestinian people speak the Palestinian dialect, which is a subset of the informal Arabic spoken in the southern Levantine region. Several Palestinian distinct dialects differ from one city to another and from various geographic regions i.e. Ramallah, Nablus, Jerusalem, and Hebron, which enhances the linguistic diversity of the nation. The goal of this project is to build and evaluate a simple-to-use system that can identify regional Palestinian accents. The main objective is to correctly identify a speaker's accent from a brief speech file while differentiating between the four main Palestinian accents that were previously mentioned. Gaining an understanding of these dialects is crucial to expressing the distinctive speech patterns of each area. [1]

Recognizing accents can be challenging due to the subtle acoustic variants that separate one accent from another. Accent recognition calls for a nuanced approach to capture the distinctive qualities of every accent, in contrast to language identification, where distinctions are easier to recognize. In order to create a system that can precisely distinguish these differences, this project makes use of acoustic feature extraction techniques. [1]

Our approach involves using existing dataset of speech samples from speakers of each accent. Next, from these samples, we will extract relevant acoustic features that change with accent, like MFCC, energy, contrast and Mel-spectrogram properties. Then, classification model intended to identify the accent of any given speech segment will be trained and tested using these features.

By the end of this project, we aim to provide a comprehensive evaluation of our accent recognition system. The evaluation will point out the system's performance measures, like accuracy, precision, F1-score, and recall dependability, and identify any areas that require work before it can be improved upon in subsequent iterations.

By providing insights into the subtleties of accent recognition, this work makes a significant contribution to the field of speech processing. It also has useful applications across a range of fields. For example, by improving speech recognition systems' ability to accommodate various accents, it can improve user experience. Additionally, by offering in-depth analyses of regional speech patterns, it can support linguistic research. Furthermore, through digitally recording and identifying the various Palestinian dialects, this project contributes to the preservation of the cultural past.

The remaining sections of this paper are organized as follows: Section 2 introduces the overall background and prior works; Section 3 presents the features extracted and the models SVM, GMM-SVM, and KNN applied; Section 4 describes the experiments and results analysis; and Section 5 concludes the paper.

2. Background/Related Work

Various efforts have been made in developing different methods to identify dialects of different regions over the past years. Much research has been done in this area, exploring different models based on acoustic features, which later also included deep neural networks.

These studies began long ago, with Reynolds et al beginning in 2000 [2] using a Gaussian mixture-based model to build a dialect-independent model called UBM (Universal Background Model) and then using adaptive MAP (Maximum A Postiori) to adapt UBM parameters for each target dialect. Then, in 2005 [3], Yanli Cheng and his team investigated speech recognition based on different Chinese dialects. They combined MLR (Maximum Linear Regression) with MAP, and optimized and adapted them individually for each speaker in the test. Then in 2006 [4], Konstantin Markov and Satoshi Nakamura used HMM (Hidden Markov Model) with a Bayesian network for dialect-based speech recognition, and in the same year SVM (Support Vector Machines) [5] were used to classify labels in a hypervector field. This system is referred to as GMM-SVM which is improved by incorporating the concepts of JFA (Joint Factor Analysis) to reduce the variance between sessions within a single dialect. In 2011 [6], A.N. Mishra and his team in India studied the park frequency and regression vertical frequency coefficients to describe robust

features for number recognition in a noisy and clean environment based on a HMM and used their toolkit for MFCC, where MFCC was chosen for feature extraction because it is the most commonly used and most efficient, while the other features are extracted by Matlab and saved in HTK (Hidden Markov Model Toolkit) format, which was later confirmed to be accurate compared to Matlab, with HTK giving an accuracy of 99-100% for clean data, which is 5-6% better than Matlab. In a noisy environment, HTK outperforms in terms of accuracy by 89-94% compared to Matlab. In the same year, Biadsy study [7] was published, which pointed out the diversity of methods that rely on and take advantage of acoustic features in building dialect recognition systems. The best study was tested on several Arabic and English languages, and the method of this study showed an error level ranging between 4-14.6 % between different languages and dialects. In 2013 [8], Gaikwad et al focused-on pronunciation between Marathi and Arabic by extracting acoustic features such as energy, pitch, and formant frequency and using them in the experiment. It was found that the formula frequency feature gives a better result for the Marathi language, unlike the Arabic language, which had better results with the energy feature. In another study in 2016 [9], Pham et al. combined the Mel Frequency Cepstrum Model and the F0 of the GMM (Gaussian Mixture Model) in Vietnamese Dialect Recognition It has been confirmed that combining the formulas and bandwidths with the natural F0 results in enhancing the identification of the basic dialect of the language from 58.6% to 72.2%.
In 2016 [10], Jiao et al. 2017 [11] Astrid et al. they began to explore methods based on DNNs (Deep Neural Networks). It was found that the performance of these methods outperforms the latest vector methods in many fields.

3. Methodology

After understanding the idea of the project and reviewing the four accents that we need to distinguish. It can be noticed that the methodology of our work on this project consists of several steps. Each step represents an important part in reaching the goal and the required evaluation. The following block diagram shown in Figure 1 summarizes the working methodology.

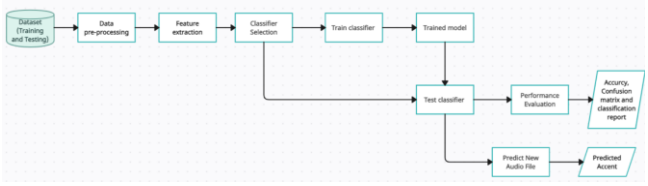

Figure 1: Block diagram for the Methology

In the data pre-processing part, the files containing the audio files were decompressed and arranged. The Ramallah Reef files names were different from each other. So, they have been renamed, and then the files were read and ensured that they were not empty. In the next step, acoustic features were extracted for both training files and testing files, the work relied mainly on extracting the MFCC feature in each algorithm.

MFCC

MFCCs are useful in machine learning models, uniquely emphasize the Mel-scale, which mirrors human auditory sensitivity to different pitches, and they provide a decorrelated feature set that simplifies machine learning models tasks. MFCCs simplify the features into a form that is easier for algorithms to handle, it often leads to more accurate and efficient models.

Spectral contrast

Spectral contrast was extracted and used in GMM-SVM, Spectral contrast measures the difference in amplitude (dB) between peaks and valleys in the sound spectrum, highlighting the presence of strong harmonic content versus noise. This helps models detect subtle nuances in sounds, making it particularly useful for distinguishing between different types of audio environments or sound textures.

Mel-spectrogram

Mel spectrogram features were extracted and used in SVM model, a Mel-spectrogram represents audio signals in the Mel scale frequency, emphasizing human auditory sensitivities to pitch and loudness, which enhances feature representation for vocal and environmental sound distinctions, it provides a more perceptually relevant display, so, it is useful for speech analysis.

Energy

In machine learning, especially for audio tasks, energy features are useful because they measure how loud or intense a sound is. This helps models figure out whether there's sound or silence, making it easier to tell different sounds apart or detect when there's no sound at all. By integrating energy as a feature, the algorithms can improve their accuracy in identifying and classifying sounds, especially in diverse acoustic conditions. Energy was extracted and used in SVM model.

For the stage of choosing a suitable learning machine model, three models were selected and they are Gmm-Svm, SVM, and KNN.

GMM-SVM

Mix of two techniques Gaussian Mixture Models (GMM) and Support Vector Machines (SVM) were used to identify different accents in audio samples. First, the GMM learns the unique patterns and variations in accents by analyzing the data. It then creates new features called posteriors, which are basically the probabilities showing how likely it is that a piece of data belongs to a certain group or class in the model. These posteriors were added to the original data features to make the information richer and more detailed. Next, we train the SVM on this enhanced set of features. The SVM is great at sorting and classifying data, so it takes this detailed information and accurately determines the accent of each new audio sample.

SVM

Support Vector Machine (SVM) was used for classifying accents from audio data. SVM is a powerful machine learning algorithm that works by finding the best boundary that separates data points of different classes. By using a linear kernel in this context, the SVM attempts to differentiate between the four Palestinian accents by analyzing the features extracted from the audio, like MFCCs and Mel-spectrograms.

KNN

the K-Nearest Neighbors (KNN) algorithm was used to classify audio samples based on their accents. KNN is a straightforward machine learning algorithm that classifies data points based on the most common class among the nearest k neighbors. By analyzing the features extracted from audio files, such as MFCCs, KNN looks at the 'k' closest training

examples and predicts the accent based on the majority vote among these examples.

These models showed a clear superiority in the results over the rest of the other models, all gave good accuracy appropriate to the given data set, after that, the extracted feature was trained in the selected models. After testing the classifiers, the Accuracy, Precision, Recall, F1-score, and confusion matrix were evaluated for each model.
Finally, the prediction part was added to the models, the code allows the user to enter any external voice from among the four existing accents, the model can predict some of them correctly and others incorrectly, and this is logical because the accuracy rate and other evaluations are not so high values.

4. Experiments and Results

In this section, the data set used, the experiments performed, how they were accomplished, and the results obtained will be explained.

4.1. Dataset

Table I below shows the details of the dataset more clearly, the number of sound files and the total duration in minutes in the training and testing data files for each accent, it appears that it is a relatively small data set in terms of the number of speakers or in terms of the total duration of the sounds.

Table I: Speaker Distribution in the Dataset

	NO. of Train Files	***NO. of Test Files***	***Duration of Train Set (min)***	***Duration of Test Set (min)***
Jerusalem	10	5	9.51	22.53
Hebron	10	5	17.20	4.68
Ramallh-R	10	5	7.34	13.28
Nablus	10	5	50.42	9.56
Total	40	20	84.47	50.05

It also clearly appears that there is a large difference in the total duration in minutes for each class of accents, whether in training data or in testing data. Which makes obtaining so high-performance evaluations almost difficult.

4.2. Feature Extractions

First, for the MFCC, it has been found that increasing the number of parameters leads to improving the readings in the performance metric in the algorithms used. A small number does not give the desired results

For example, in the GMM-SVM, when the number of MFCCs is increasing from 40 to 80 with existence of the Spectral contrast feature, the Accuracy was increased from 70% to 75%, second example, in the SVM algorithm, when using the feature without any other features when using a number of MFCCs equal to 40, the Accuracy will be 70%, but when raised the number 80, the Accuracy will be 75%
and here the strong influence of the number of MFCCs on the model results is clearly evident.

Second, in SVM1 code, Only MFCC with 40 parameters were used, and the code gives Accuracy 70% but when Mel-spectrogram and energy features added to the extraction process "SVM2 code", the Accuracy increased to 75%, this change shows the important and the advantages of adding other features with MFCC like Mel-spectrogram and energy features.

4.3. Machine Learning Models

4.3.1. GMM-SVM

In this model, when using the kernel type in the SVM to be Linear, it will give the highest values in the measurements. However, when using other types, such as Polynomial or RBF, this will lead to a significant decrease in accuracy, so the Linear type was adopted. When using the number of Gaussians to be 4 with Diagonal covariance type in this code accuracy will be 75%. When the number increases from 4 to 8, the Accuracy will still the same.

4.3.2. SVM

Linear kernel was used in SVM to simplifies the decision boundary, making it efficient for high-dimensional data. It tends to perform well when the data is linearly separable or has a large number of features. Additionally, linear kernels are computationally less expensive compared to other kernels, making them suitable for large datasets.

4.3.3. KNN

In this model, when using the number of neighbors to be 4, it will give an Accuracy equal to 65%, it was noticed that if this number increase to eight neighbors, the Accuracy will be the same. However, if the increase is greater to 12, this will sabotage the calculations and make the accuracy equal to 50%.

4.4. Results

In this section, the results obtained for each model such as the Evolution of the performance, Accuracy, precession, Recall, F1-Score and the Confusion Matrix will be presented, and a comparison between each algorithm and the other in terms of the apparent results. Figure 2 shows the difference in accuracy between the selected models in the project and the other models that were tested also.

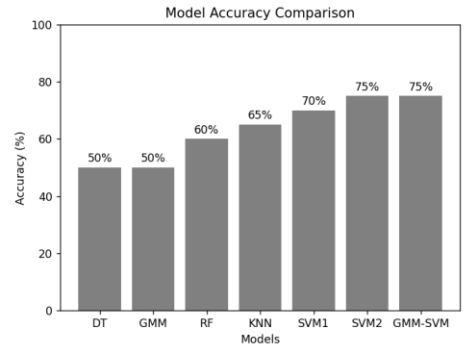

Figure 2: Accuracy Comparison

The graph above compares how well different models predict accents, measured by accuracy percentages. The models SVM2, and GMM-SVM perform best result with 75% accuracy. The Decision Tree (DT) and Gaussian Mixture Model (GMM) are the least accurate, both scoring only 50%.

Table II contains the Performance Evaluations for selected models.

Table II: Performance Evaluations for the models

	Acc.%	*Precision%*	*Recall%*	*F1%*
GMM-SVM	75	80	75	74
SVM2	75	75	75	74
SVM1	70	79	70	67
KNN	65	80	65	63

The above table compares the selected models used to predict accents using four measures: accuracy, precision, recall, and F1 score. The GMM-SVM model has the best balance with 75% accuracy, 75% recall, 74% F1 score, and 80% precision. The KNN model scores the lowest across all metrics, indicating it might not perform as well in recognizing the correct accents compared to other models.

The following confusion matrix shown in Figure 3 shows how well a model predicts accents from four different accents. The numbers along the diagonal tell us how many times the model correctly identified each city's accent. The other numbers show how often the model got it wrong. For example, it often mixed-up Nablus with Ramallah but did well with Hebron and Jerusalem.

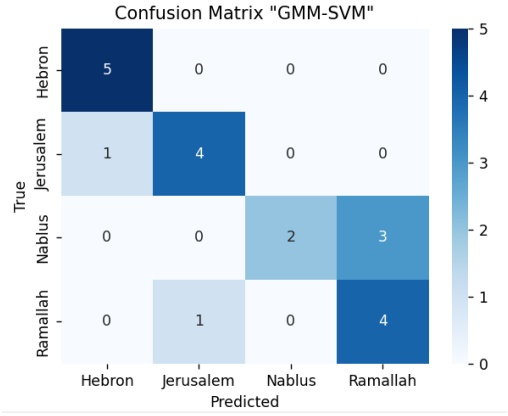

Figure 3: Gmm-Svm Confusion Matrix

Figure 4 shows confusion matrices for two models, SVM 1 and SVM 2. The numbers on the diagonal show correct predictions (like SVM 1 and SVM 2 both correctly identified all 5 Hebron accents). The other numbers show mistakes. SVM2 shows superiority in Nablus over the SVM1.

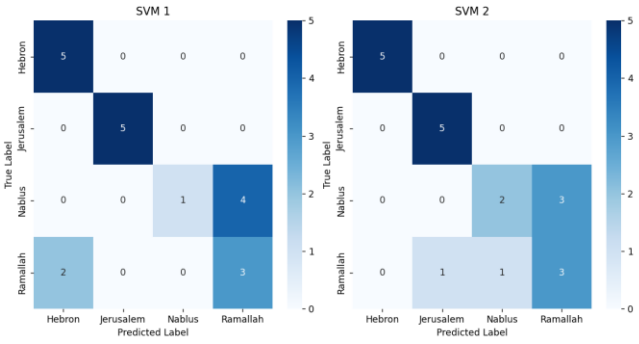

Figure 4: SVM1 and SVM2 Confusion Matrix

The confusion matrix shown in Figure 5 shows the results of a KNN model. The model did well with Jerusalem, correctly identifying all five accents, and fairly well with Hebron and Ramallah, with a few misclassifications. However, it struggled with Nablus, where it only correctly identified one target and misclassified the others as Hebron or Ramallah. This summary helps understand the model's strengths and weaknesses in accent recognition.

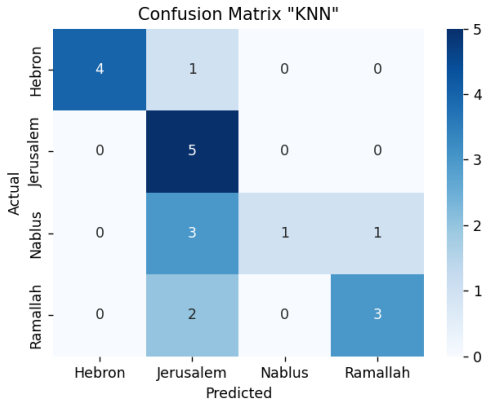

Figure 5: KNN Confusion Matrix

5. Conclusion and future work

In conclusion, in this paper, three models were studied and used for the process of distinguishing between the dialects of four Palestinian cities. These models are, in order of superiority, SVM_GMM, SVM, and KNN.
Accent recognition is a pre-processing step in speech recognition that can help fine-tune speech recognition systems to better detect distinct speech. To do this, we used several features that helped increase the accuracy of the models, as we were able to distinguish between them and know which feature was useful for which model. Among them, the most used feature was MFCC, which gave a clear and excellent difference in the results.

In general, the results were somewhat satisfactory, as the accuracy reached 75% in the SVM_GMM model, but on the other hand, we could have obtained better evaluations if we had a larger and more detailed data set.
As a final step, and in order to make our experience in this project more challenging and enjoyable, we collected a voice from each city and examined the voice in each model to see if the model would recognize this voice as belonging to any city. The performance of the models was excellent in terms of identifying at least 3 votes out of 4. For one model.

It is strongly advised to expand the speech data and incorporate regions beyond those already identified for future research. An even distribution of participants across age groups, genders, and languages can be taken into account to enhance the accuracy of the accent recognition model for speakers.

6. Partners participation tasks

In this paper, the work was largely collaborative, but the writing was divided as follows: Amal wrote the introduction and abstract, while Layan wrote the background and related works, along with the conclusion. As for Yazeed, he wrote Methodology with Experiments and Results.

7.References

[1] 7 11 2021. [Online]. Available: https://gopalestine.org/study-the-palestinian-dialect-in-palestine-in-2022/. [Accessed 9 6 2024].

[2] D. A. Q. T. F. a. D. R. B. Reynolds, 1 2000. [Online]. Available: https://www.researchgate.net/publication/222674333_Speaker_Verification_Using_Adapted_Gaussian_Mixture_Models. [Accessed 9 6 2024].

[3] R. S. L. G. I. S. Z. Y. S. D. J. R. S. S.-Y. Y. Yanli Zheng, 9 2005. [Online]. Available: https://www.researchgate.net/publication/221479462_Accent_detection_and_speech_recognition_for_Shanghai-accented_Mandarin. [Accessed 9 6 2024].

[4] S. N. Konstantin MARKOV, 3 2006. [Online]. Available: https://www.researchgate.net/publication/31157975_Using_Hybrid_HMMBN_Acoustic_Models_Design_and_Implementation_Issues. [Accessed 9 6 2024].

[5] D. E. S. a. D. A. R. William Campbel, 6 2006. [Online]. Available: https://www.researchgate.net/publication/3343440_Support_vector_machines_using_GMM_supervectors_for_speaker_verification. [Accessed 9 6 2024].

[6] M. C. A. B. S. N. S. A. N. Mishra, 2 6 2011. [Online]. Available: https://article.nadiapub.com/IJSIP/vol4_no2/8.pdf. [Accessed 9 6 2024].

[7] 2011. [Online]. Available: https://academiccommons.columbia.edu/doi/10.7916/D8M61S68. [Accessed 9 6 2024].

[8] D. W. G. S. K. G. Karbhari Kale, 2 2013. [Online]. Available: https://www.researchgate.net/publication/258070241_Accent_Recognition_for_Indian_English_using_Acoustic_Feature_Approach. [Accessed 9 6 2024].

[9] L. T. V. P. N. H. Nguyen Hong Quang, 7 2016. [Online]. Available: https://www.researchgate.net/publication/340231347_AUTOMATIC_IDENTIFICATION_OF_VIETNAMESE_DIALECTS. [Accessed 9 6 2024].

[10] J. M. L. Y. J. Ming TuVisar Berisha, 9 2016. [Online]. Available: https://www.researchgate.net/publication/307889236_Accent_Identification_by_Combining_Deep_Neural_Networks_and_Recurrent_Neural_Networks_Trained_on_Long_and_Short_Term_Features. [Accessed 9 6 2024].

[11] V. B. S. P. H. A. E. T. G. Shelby Carleton, 2017. [Online]. Available: https://ojs.aaai.org/index.php/AIIDE/article/view/12968/12816?fbclid=IwZXh0bgNhZW0CMTAAAR1MmUHzzRXzrBu6it4hCc5u1UkuABSsHfXziIMpinGSiGo7RZomezYuvoY_aem_AUNsMR4BkBGcj8gKusUDXfeNryicrDkW7G8_hXMVJg9xG307fEMuNAUA0jDCyE9SsP_aZgVu1RARRt9oM3Dxm-XF. [Accessed 9 6 2024].

Appendix

Code source

https://drive.google.com/drive/folders/1obX7hGYy9N1cVhX1rImysEHFhQecvXwU