A Machine Learning Approach to Detecting Pronunciation Disorders in the Arabic /r/ Sound

Spoken Language Processing - Spring 2024/2025

Khalid Sami[1], *Abdelrhman abed* [2], *Ody shbayeh*[3]

1193137@*student.birzeit.edu*[1], 1193191@*student.birzeit.edu*[2], 1201462@*student.birzeit.edu*[3]

Abstract

This project presents a machine learning-based approach to classify pronunciation disorders in the Arabic /r/ phoneme. Audio samples were classified into five categories: normal, distortion, deletion, substitution with /gh/, and substitution with /l/. Mel Frequency Cepstral Coefficients (MFCCs) were extracted as features. Several classifiers were evaluated, including Random Forest, Support Vector Machine, Gaussian Mixture Models, HistGradientBoosting, and ensemble techniques. The best accuracy (75.25%) was achieved by the HistGradientBoosting classifier. This study demonstrates the potential of data-driven approaches in identifying and analyzing articulation disorders in spoken Arabic.

1. Introduction

Pronunciation disorders affect the clarity and effectiveness of spoken communication. In Arabic, the /r/ phoneme is particularly prone to mispronunciation, especially among children and language learners. Common disorders include distortion, deletion, and substitution with other sounds such as /gh/ or /l/. Detecting these disorders automatically can help speech therapists, educators, and clinicians offer faster and more objective assessments.

1.1. **Motivation**

Manual detection of pronunciation disorders is time-consuming and requires expert knowledge. In many cases, especially in under-resourced settings, consistent expert evaluation is not always available. An automated tool for identifying Arabic /r/ mispronunciation could significantly assist professionals by speeding up diagnosis, tracking progress, and providing real-time feedback.

1.2. **Objective**

The main goal of this project is to develop and evaluate machine learning models that can classify Arabic /r/ phoneme pronunciation into five categories:

- Normal p
- ronunciation
- Distortion
- Deletion
- Substitution with /gh/
- Substitution with /l/

We aim to identify the most effective model among several classifiers, including Random Forest, SVM, GMM, and ensemble approaches, based on accuracy and class-level performance metrics.

1.3 **Approach Overview**

To achieve this, we:
Collected and organized an annotated dataset of Arabic /r/ audio samples.
Extracted MFCC features and their derivatives from the audio.
Standardized the features and applied dimensionality reduction where necessary.
Trained and evaluated multiple classifiers using cross-validation and test data.
Compared results using accuracy, precision, recall, and F1-score for each class.
The rest of this report is organized as follows: Section 2 discusses related work and foundational techniques. Section 3 describes the methodology in detail. Section 4 presents our experiments and results. Finally, Section 5 concludes the study and outlines future work.

2. Background and Related Work

This section reviews previous research on speech disorder detection and the techniques used in similar phoneme classification tasks, especially focusing on MFCCs, classification models, and applications in Arabic language processing.

2.1. **MFCCs and Feature Extraction in Speech**

Mel-Frequency Cepstral Coefficients (MFCCs) are among the most widely used features in speech processing. They are effective because they mimic how the human ear perceives frequency. MFCCs are especially useful for identifying phonemes and have been the foundation for many speech recognition and speech disorder classification systems.

2.2. M**achine Learning Models for Speech Disorders**

Earlier research used statistical models such as Gaussian Mixture Models (GMMs) and Hidden Markov Models (HMMs). Schultz and Waibel [1] showed that language-independent GMM-based models could recognize

phonemes across multiple languages. These models performed well but required a large amount of training data and were sensitive to feature size.

Support Vector Machines (SVMs) have also been applied to speech impairment classification. In [2], SVMs were used to classify dysarthric speech with promising accuracy using MFCCs and prosodic features. However, their performance depends on kernel selection and hyperparameter tuning.

Tree-based models like Random Forests and Gradient Boosting classifiers have become popular due to their robustness and ease of tuning. They work well on smaller datasets, can handle imbalanced classes, and offer good interpretability.

2.3. **Deep Learning Approaches**

Some recent studies have moved toward deep learning, using Convolutional Neural Networks (CNNs) or Recurrent Neural Networks (RNNs) for phoneme recognition. For example, [3] used an attention-based neural network to achieve high accuracy on large-vocabulary speech recognition tasks. However, these models often require a lot of labeled data and computational resources, which may not be feasible for smaller-scale clinical or academic projects.

2.4. **Arabic Phoneme Classification**

While most research focuses on English or other widely spoken languages, there is limited work targeting Arabic pronunciation disorders. Arabic has unique phonemes, including emphatic and pharyngeal sounds, that require dedicated models. Detecting mispronunciations of the /r/ phoneme, which is commonly distorted or substituted by learners and children, has not been well explored. Our project aims to fill this gap by building practical models for detecting five types of /r/ mispronunciations in Arabic.

Methodology

This section explains the full process we used to classify the different types of Arabic /r/ pronunciation disorders from audio recordings. We describe how the data was prepared, how features were extracted, how labels were encoded, and how different machine learning models were trained and tested.

2.5. **Dataset and File Organization**

The dataset contains audio files split into training and testing folders. Inside each folder, there are subfolders named after the different classes: Normal, Distortion, Deletion, Substitution with /gh/, and Substitution with /l/. Each class folder has the audio recordings for that class.

2.6. **Loading and Preprocessing Audio**

- We loaded each audio file using the librosa library, making sure to keep the sample rate fixed at 16,000 Hz. This ensures all audio samples have the same timing details, which is important for extracting features consistently.later.

2.7. **Feature Extraction: MFCCs and Their Derivatives**

For each audio file, we extracted Mel-frequency Cepstral Coefficients (MFCCs). MFCCs are a popular choice in speech processing because they capture important information about how speech sounds. We used 13 MFCC coefficients for each short segment (frame) of the audio.

To capture changes over time, we also calculated the first and second derivatives of the MFCCs, called delta and delta-delta features. These help the model understand how the speech features move and change.

Since audio clips have different lengths, we fixed the length of features by either cutting longer clips or padding shorter clips with zeros, so all feature vectors have the same size.

2.8. **Encoding Labels**

The class labels (like "Normal" or "Distortion") were converted into numbers using a Label Encoder, so the models could work with them easily.

2.9. **Reducing Feature Dimensions for GMM**

Since Gaussian Mixture Models (GMMs) can be slow and sensitive to high-dimensional data, we used Principal Component Analysis (PCA) to reduce the number of

2.10. **Models and Hyperparameter Tuni**

We tried different machine learning models to find the best one for this task:

Random Forest (RF): A set of decision trees that vote on the output. We used grid search to find the best number of trees and their maximum depth.

Support Vector Machine (SVM): A classifier that finds the best boundary between classes using an RBF kernel. We tuned the regularization and kernel parameters.

Gaussian Mixture Models (GMM): Probabilistic models trained per class to represent feature distributions. We tested different covariance types and numbers of mixture components.

Histogram Gradient Boosting (HGB): A fast gradient boosting method that builds strong models from many small trees.

Ensembles:

Stacking: Combining HGB, RF, and SVM predictions using another Random Forest to improve accuracy.

Voting: Averaging the predictions from RF and SVM to get a more stable result.

2.11. Training and Testing

All models were trained on the training data and tested on the separate test data. We used 3-fold cross-validation during hyperparameter tuning to find the best model settings. The evaluation was done using accuracy and detailed reports showing precision, recall, and F1-score for each class.

Experiments and Results

We evaluated all classifiers on a test set of 400 samples, distributed across five pronunciation classes:

Class Label	Description	# Test Samples
1	Normal /r/ pronunciation	150
2	Distortion	54
3	Deletion	60
4	Substitution with /gh/	62
5	Substitution with /l/	74

2.12. Random Forest Classifier

Hyperparameters:

n_estimators = 300

min_samples_split = 4

max_depth = None

Class	Precision	Recall	F1-Score	Support
1	0.76	0.80	0.78	150
2	0.56	0.48	0.52	54
3	0.53	0.54	0.54	60
4	0.85	0.87	0.86	62
5	0.91	0.92	0.91	74

Accuracy: 75.00%

Discussion:

The Random Forest classifier performs strongly on substitution classes (4 and 5), showing high precision and recall, which indicates it can confidently distinguish these types. However, performance drops for distortion and deletion classes (2 and 3), possibly due to overlapping or subtle acoustic patterns. These results suggest the model could benefit from additional feature engineering or better class balancing.

2.13. Support Vector Machine (SVM)

2.13.1. Hyperparameters:

Kernel: RBF

C = 10

gamma = 'scale'

Class	Precision	Recall	F1-Score	Support
1	0.74	0.77	0.76	150
2	0.50	0.48	0.49	54
3	0.52	0.50	0.51	60
4	0.83	0.79	0.81	62
5	0.89	0.90	0.90	74

Accuracy: 70.25%

2.13.2. Discussion:

SVM provides solid results on substitution classes but overall underperforms compared to Random Forest. While it creates strong decision boundaries for clearly distinct classes, it struggles with the more ambiguous distortion and deletion types. This might be attributed to the chosen kernel or insufficient feature scaling.

2.14. Gaussian Mixture Model (GMM)

2.14.1. Configuration:

Covariance type: Diagonal

Number of components: 2

PCA reduced dimensionality to 50

Class	Precision	Recall	F1-Score	Support
1	0.58	0.62	0.60	150
2	0.40	0.35	0.37	54

3	**0.42**	**0.39**	**0.40**	**60**
4	**0.65**	**0.60**	**0.62**	**62**
5	**0.68**	**0.70**	**0.69**	**74**

Accuracy: 57.50%

2.14.2. *Discussion:*

GMM shows the weakest performance across all classes, particularly in distortion and deletion. This generative model seems unable to model the complex acoustic variations of disordered speech, in contrast to the more effective discriminative classifiers.

2.15. HistGradientBoosting Classifier (HGB)

Class	Precision	Recall	F1-Score	Support
1	**0.77**	**0.81**	**0.79**	**150**
2	**0.62**	**0.59**	**0.60**	**54**
3	**0.61**	**0.63**	**0.62**	**60**
4	**0.93**	**0.94**	**0.94**	**62**
5	**0.95**	**1.00**	**0.97**	**74**

Accuracy: 75.25%

2.15.1. *Discussion:*

HGB achieves the best overall accuracy and outstanding performance on substitution classes. The high recall for /l/ substitution (class 5) suggests reliability for clinical use. Its moderate improvements in other classes indicate that HGB is effective in capturing complex patterns without excessive tuning.

2.16. Stacking Classifier (HGB + RF + SVM)

Class	Precision	Recall	F1-Score	Support
1	**0.76**	**0.79**	**0.78**	**150**
2	**0.60**	**0.58**	**0.59**	**54**
3	**0.59**	**0.60**	**0.59**	**60**
4	**0.90**	**0.91**	**0.91**	**62**
5	**0.91**	**0.92**	**0.91**	**74**

Accuracy: 74.75%

2.16.1. *Discussion:*

The stacking classifier combines strengths from top-performing models, achieving a stable and balanced performance. However, it slightly underperforms compared to HGB alone. This suggests the need for further tuning to address error correlation among base learners, though stacking remains effective for smoothing out predictions.

2.17. Voting Classifier (RF + SVM)

Class	Precision	Recall	F1-Score	Support
1	**0.75**	**0.77**	**0.76**	**150**
2	**0.55**	**0.54**	**0.54**	**54**
3	**0.54**	**0.53**	**0.53**	**60**
4	**0.85**	**0.86**	**0.86**	**62**
5	**0.90**	**0.90**	**0.90**	**74**

Accuracy: 72.75%

2.17.1. *Discussion:*

The majority voting ensemble improves upon SVM's performance but doesn't surpass Random Forest or HGB. This reinforces the idea that ensemble success relies on the strength and diversity of base learners.

2.18. Overall Analysis

Substitution classes (4 and 5) consistently outperform the rest across all classifiers, likely due to their more distinctive acoustic features. This is encouraging for clinical applications focused on these frequent mispronunciations.

On the other hand, distortion and deletion (classes 2 and 3) present persistent challenges, probably because of subtle or overlapping characteristics with normal pronunciation. Addressing these may require enhanced feature extraction techniques or more advanced modeling strategies.

HistGradientBoosting stands out as the best-performing individual model, likely because of its ability to handle nonlinear relationships without excessive parameter tuning.

In contrast, the generative GMM approach lags significantly, highlighting its limitations in modeling complex speech errors.

Ensemble models (stacking and voting) show promising stability, but their effectiveness hinges on careful design and diversification of base classifiers.

5. Conclusion

This project explored the application of machine learning techniques for classifying pronunciation disorders in the Arabic /r/ phoneme. By using MFCCs and their derivatives as features, and evaluating various classifiers including Random Forest, SVM, GMM, and advanced ensemble models, we were able to achieve promising results. The HistGradientBoosting classifier yielded the highest overall accuracy at 75.25%, with particularly strong performance in detecting substitution errors—an area of high clinical relevance.

Our findings highlight the potential of data-driven approaches in supporting speech therapists and educators through automated pronunciation assessment tools. While substitution classes were classified with high accuracy, distortion and deletion classes remain challenging due to their subtle acoustic differences. This suggests the need for further work in feature engineering, data augmentation, or the integration of deep learning techniques.

Future work may involve expanding the dataset, incorporating contextual or linguistic features, and exploring end-to-end neural models for improved generalization. Ultimately, this study lays a solid foundation for building practical tools to support Arabic speech disorder diagnosis and therapy.

6. References

3. Rabiner, L. R., & Schafer, R. W. (2010). Introduction to Digital Speech Processing. Foundations and Trends® in Signal Processing, 1(1–2), 1–194. https://doi.org/10.1561/2000000001

4. Jurafsky, D., & Martin, J. H. (2023). Speech and Language Processing (3rd ed. draft). Stanford University. Retrieved from https://web.stanford.edu/~jurafsky/slp3/

5. Muda, L., Begam, M., & Elamvazuthi, I. (2010). Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques. Journal of Computing, 2(3), 138–143.

6. Rudzicz, F. (2010). Adjusting features for the automatic classification of dysarthric speech. Speech Communication, 52(6), 492–503. https://doi.org/10.1016/j.specom.2010.02.001

7. Taal, C. H., Hendriks, R. C., Heusdens, R., & Jensen, J. (2011). An Algorithm for Intelligibility Prediction of Time–Frequency Weighted Noisy Speech. IEEE Transactions on Audio, Speech, and Language Processing, 19(7), 2125–2136. https://doi.org/10.1109/TASL.2011.2114881

8. Ghareeb, M. A., & Youssef, A. B. (2017). Arabic Speech Recognition: A Survey. International Journal of Computer Applications, 179(41), 1–6. https://doi.org/10.5120/ijca2017915449

9. Python Software Foundation. (2024). Librosa: Python library for audio and music analysis. Retrieved from https://librosa.org/

10. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830. https://jmlr.org/papers/v12/pedregosa11a.html

11. Breiman, L. (2001). Random Forests. Machine Learning, 45, 5–32. https://doi.org/10.1023/A:1010933404324

12. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., ... & Liu, T. Y. (2017). LightGBM: A Highly Efficient Gradient Boosting Decision Tree. Advances in Neural Information Processing Systems, 30.