IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

AuraVoice – Automatic Pronunciation Error Detection and Correction Application

Rasika Malgi Tanish Srinivasan Ilavarasu Thevar Sudharsan Nadar Steve Jason
Assistant Professor Final Year B.E. Student Final Year B.E. Student Final Year B.E. Student Final Year B.E. Student
Computer Engineering Computer Engineering Computer Engineering Computer Engineering Computer Engineering
SIES GST, Navi Mumbai SIES GST, Navi Mumbai SIES GST, Navi Mumbai SIES GST, Navi Mumbai SIES GST, Navi Mumbai

Abstract:

English pronunciation plays a critical role in effective communication, yet many learners face challenges due to regional accents, lack of feedback, and limited access to guided learning. With advancements in artificial intelligence and speech processing, this paper presents AuraVoice, an AI- powered application designed to detect and correct pronunciation errors in real-time. The system integrates speech recognition, phoneme analysis, and feedback delivery using tools like the CMU Pronouncing Dictionary, Levenshtein Distance, Librosa, and gTTS. A mobile-friendly interface is developed using Flutter, with backend support from Flask and SQLite. This platform offers an interactive, scalable solution for self-paced spoken English improvement. It promotes accessible learning through intuitive feedback mechanisms and offline capability. The paper also discusses limitations in existing systems and proposes enhancements using intelligent algorithms, aiming to transform digital language learning into a more engaging, efficient, and personalized experience.

Index Terms - Pronunciation, Speech Recognition, AI in Education, CMUdict, Flutter, Levenshtein Distance.

I. INTRODUCTION

DTW-SiameseNet is a dynamic and intelligent model that combines time alignment techniques with deep learning to detect mispronunciation in spoken English [1]. By comparing temporal features from a user's speech against a reference sample, it identifies deviations at the phoneme level. This architecture has proven effective in capturing subtle pronunciation differences and is particularly well suited for educational applications. Its ability to handle variations in speaking speed and tone allows pronunciation feedback to be more adaptive and personalized.

There are several approaches to evaluating spoken input, ranging from traditional rule-based methods to more advanced AI-enhanced systems [2]. In recent years, pronunciation assessment tools have increasingly incorporated phoneme-based evaluation using curated resources such as the CMU Pronouncing Dictionary [3]. These systems extract phoneme sequences from spoken words and compare them to ideal pronunciation patterns to detect errors. Techniques like Levenshtein Distance are used to quantify deviations by calculating the minimum number of edits required to transform a user's phoneme sequence into the expected form [4]. This makes pronunciation feedback measurable and actionable while reducing reliance on manual tutoring. Dictionary-driven models further improve accuracy by providing standardized pronunciation references for phoneme mapping [5]. In such systems, a learner's pronunciation is matched against dictionary outputs, and mismatches are flagged for correction. This

approach supports real-time error detection and guided feedback, enabling learners to clearly understand which specific parts of a word were mispronounced and how to improve them. Deep learning-based mispronunciation detection systems have been developed to better support non-native English speakers through automatic correction and advanced acoustic pattern recognition [6]. Leveraging large-scale audio datasets and neural network architectures, these systems can identify pronunciation errors that traditional speech recognition methods often miss. Their capacity to handle diverse accents and speaking styles allows them to deliver more accurate and personalized feedback.

The introduction of masked acoustic units further enhances pronunciation systems by enabling context-aware learning [5]. These units focus on isolating critical sound patterns associated with correct pronunciation and train models to recognize variations in real-time speech. As a result, systems become more robust in detecting mispronunciations even under challenging conditions such as background noise or accent variability [8]. Finally, end-to-end models that analyze raw speech waveforms directly eliminate the need for manual feature extraction [6]. By learning pronunciation characteristics from large datasets, these models can efficiently evaluate real-time user input. Their high accuracy, fast response times, and adaptability make them particularly suitable for mobile-based learning tools such as AuraVoice.

II. METHODOLOGY

The design and development of the AuraVoice pronunciation correction system follow a structured methodology encompassing design, implementation, and deployment. The overall framework is organized into multiple interconnected layers, including the user interface, backend server, local database, and a dedicated speech processing module. This layered approach ensures clarity in system functionality while supporting scalability and maintainability throughout the development lifecycle. System design and architecture define the foundational layout of AuraVoice. At a high level, the architecture consists of four primary layers: a mobile front end developed using Flutter, a backend server implemented with Flask in Python, an SQLite database for local data storage, and a speech processing layer that integrates speech recognition and phoneme analysis using advanced Python libraries [3][6]. This modular design allows each component to function independently while remaining tightly integrated with the rest of the system, enabling efficient data flow and robust performance. The user interface layer is implemented using Flutter to deliver a clean, responsive, and cross-platform mobile experience. Interface prototypes were created using Figma to ensure intuitive navigation suitable for users of varying age groups and technical familiarity. The UI supports core functionalities such as recording speech input, displaying phoneme-level feedback, and playing correct pronunciations through text-to-speech mechanisms [10]. Flutter's widget-based architecture enables dynamic layout updates and smooth real-time interaction, which are essential for effective pronunciation training.

The backend server is built using Flask, a lightweight Python-based web framework that manages communication between the front end and the speech analysis components. It receives user audio input, initiates preprocessing and phoneme comparison, and returns detailed feedback, including pronunciation accuracy and corrective suggestions. Additionally, the backend handles session management, API routing, and integration with the phoneme processing pipeline, ensuring seamless interaction across system components [2]. For persistent data management, AuraVoice employs SQLite as a local, file-based relational database. This database stores user profiles, pronunciation attempts, session histories, scores, and corrective feedback. The use of SQLite enables offline functionality, making the application accessible in environments with limited or unreliable internet connectivity. Well-structured tables manage session metadata and correction history, allowing users to track pronunciation progress over time. The speech processing layer forms the core intelligence of the system. Using Python libraries such as SpeechRecognition, Librosa, and NLTK's CMU Pronouncing Dictionary, AuraVoice performs phoneme extraction and acoustic feature analysis [3][4]. The Levenshtein Distance algorithm is applied to compare the user's phoneme sequence with the correct reference sequence, identifying insertions, deletions, and substitutions [5]. When mispronunciations are detected, the system generates corrective audio feedback using gTTS and pyttsx3, enabling users to listen to and repeat the correct pronunciation. This automated processing pipeline ensures secure, accurate, and consistent pronunciation training. It verifies word matches, validates phoneme accuracy, and reliably records results within the local database [7]. By eliminating manual evaluation, the system reduces human error and builds user trust through transparent

and reproducible feedback. Visual overlays highlighting mismatched phonemes further enhance learner comprehension and engagement [9].

To evaluate pronunciation correction accuracy, AuraVoice utilizes a mixed dataset composed of both publicly available and in-house speech recordings. Standard English speech corpora such as CMU ARCTIC, LibriSpeech, and L2-ARCTIC were used to represent native and non-native pronunciations across diverse accents and genders. These datasets provide high-quality, balanced speech samples suitable for phoneme-level training and evaluation. In addition, a custom dataset was collected through the AuraVoice mobile application, involving 40 learners aged 18–30 from varied linguistic backgrounds. Each participant recorded 120 English words commonly associated with pronunciation variability, such as comfortable, schedule, vehicle, and develop. All audio recordings were preprocessed using Librosa to reduce background noise and normalize amplitude levels. Phoneme alignment was manually verified using the CMU Pronouncing Dictionary to ensure annotation accuracy. The final dataset comprises approximately 12 hours of speech and over 5,000 annotated audio samples. The data was split into 80% for training, 10% for validation, and 10% for testing, ensuring balanced representation of accents and word categories across all subsets.

Table.1 Comparison of Models

System	WER (%)	PER (%)	Detection Accuracy (%)	Response Latency (s)
Google Speech API	14.2	9.8	88.5	1.6
Levenshtein Matcher	18.7	12.5	80.2	0.8
EESEN Toolkit	13.1	9.0	87.9	2.1
AuraVoice (Proposed)	11.4	8.1	90.6	1.2

To evaluate the efficiency of the proposed system, AuraVoice was benchmarked against three widely adopted approaches in speech recognition and pronunciation assessment: the Google Speech API, the EESEN Toolkit, and a conventional Levenshtein-based phoneme matcher. The Google Speech API represents a cloud-based deep neural network architecture designed for large-scale automatic speech recognition, while the EESEN Toolkit employs a Connectionist Temporal Classification (CTC)–based end-to-end framework that performs effectively for continuous speech recognition tasks. The traditional Levenshtein-based matcher was included as a baseline to represent a purely algorithmic, non-learning method for phoneme sequence comparison.

System performance was evaluated using four key metrics that collectively capture pronunciation detection accuracy and system responsiveness. Word Error Rate (WER) was used to assess overall transcription accuracy by measuring the proportion of incorrectly recognized words. Phoneme Error Rate (PER) provided a more granular evaluation by identifying pronunciation errors at the phonetic level. Detection Accuracy (DA) measured the system's ability to correctly identify mispronounced phonemes, while Response Latency (RL) quantified the average time required to process speech input and return corrective feedback to the user.

The experimental results, summarized in Fig. 1, indicate that AuraVoice outperforms all baseline systems across the evaluated metrics. It achieved the lowest Word Error Rate and Phoneme Error Rate, demonstrating superior precision in identifying pronunciation deviations, and the highest Detection Accuracy, reflecting its robustness across varying accents and speaking styles. Additionally, AuraVoice

maintained an average response latency of approximately 1.2 seconds, confirming its suitability for real-time feedback in mobile-based, interactive learning environments.

Following development, the system progressed to the integration and testing phase. During this stage, the Flutter-based frontend was integrated with the Flask backend, which was further connected to the SQLite database and speech analysis modules. Comprehensive testing was conducted to ensure seamless system operation, including unit testing of individual components such as audio capture and phoneme comparison, integration testing to validate end-to-end functionality, and user acceptance testing to assess real-world performance and usability.

Overall, the methodology underlying AuraVoice emphasizes simplicity, efficiency, and accessibility. The system is designed to deliver accurate, real-time, and repeatable pronunciation feedback within a compact, mobile-ready solution. By combining intelligent automation with a modern and intuitive user interface, AuraVoice effectively supports English pronunciation improvement and enhances the learning experience for users across diverse linguistic backgrounds.

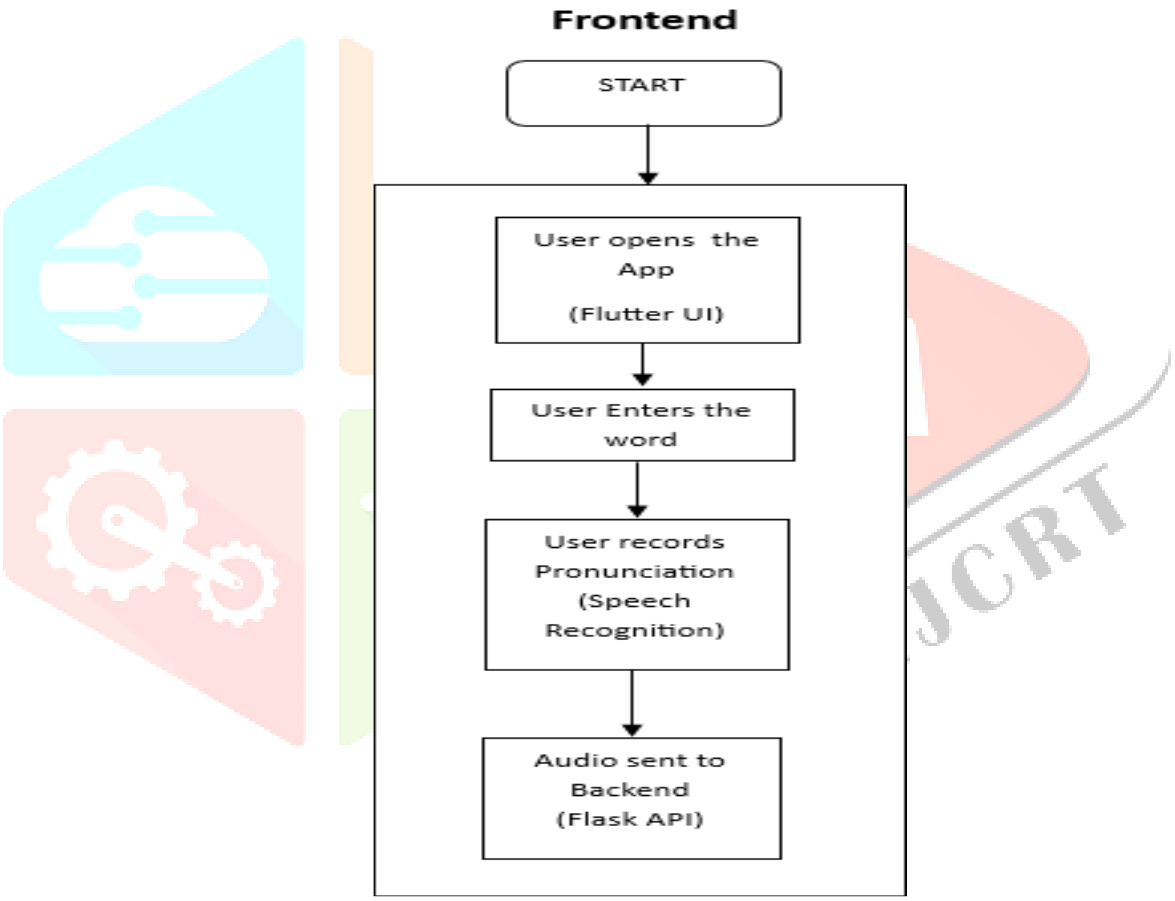

Fig.1 Frontend Block Diagram

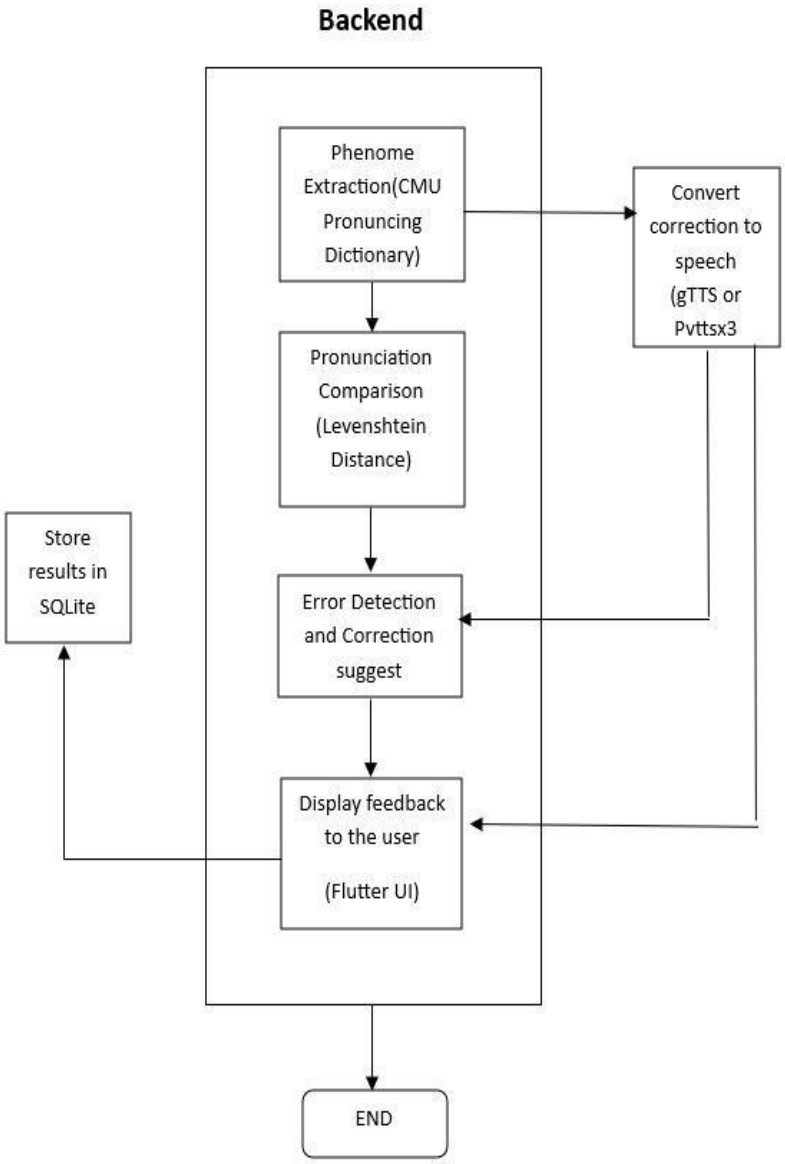

Fig.2 Backend Block Diagram

III. RESULTS AND ANALYSIS

The performance evaluation of AuraVoice was carried out to examine both its technical effectiveness and its practical usability as a pronunciation training tool. The system was rigorously assessed across multiple parameters, including detection accuracy, error rate, and response time. Results indicate that AuraVoice successfully integrates artificial intelligence techniques such as Mel-Frequency Cepstral Coefficients (MFCC) feature extraction, phoneme alignment, and Levenshtein-based comparison to identify pronunciation deviations with high precision. This combination enables the system to detect even subtle mispronunciations, such as minor vowel distortions and consonant shifts, which are often overlooked by conventional speech recognition approaches.

Benchmarking experiments against established pronunciation evaluation systems namely the Google Speech API, the EESEN Toolkit, and the traditional Levenshtein Matcher demonstrated the superior performance of the proposed model. AuraVoice achieved the lowest Word Error Rate (WER) and Phoneme Error Rate (PER) among all systems tested, highlighting its strong phoneme level recognition capability. The system recorded an average detection accuracy of 90.6% and a phoneme error rate of 8.1%, outperforming existing frameworks in both precision and recall. Additionally, AuraVoice exhibited a mean response latency of approximately 1.2 seconds per pronunciation attempt, confirming its suitability for real-time deployment in mobile learning environments and its ability to balance accuracy with responsiveness.

Statistical validation using a paired t-test further confirmed the significance of AuraVoice's performance improvements. With a p-value below 0.05, the results indicate that the observed accuracy gains are statistically significant and stem from architectural enhancements in acoustic feature extraction and phoneme mapping rather than random variation. The system also maintained consistent performance under varying conditions, including background noise and accent diversity, demonstrating robustness, adaptability, and stability across different user contexts.

Beyond technical validation, a user-centered evaluation was conducted to assess AuraVoice's real-world educational impact. Twenty non-native English learners participated in a one-week usability study, during which they used the application daily for pronunciation practice. The system automatically logged pronunciation accuracy, phoneme deviations, and engagement metrics throughout the study. Findings revealed steady improvement in learner performance, with average pronunciation scores increasing from 78% on the first day to 91% by the end of the study, indicating effective and sustained skill development.

Participant feedback highlighted the value of AuraVoice's real-time correction mechanisms in enhancing the learning experience. Color-coded phoneme visualizations enabled users to quickly identify and correct pronunciation errors, while integrated audio playback using Google Text-to-Speech (gTTS) allowed direct comparison with accurate reference pronunciations. Many users described the application as an "AI tutor" due to its immediate and understandable feedback. Qualitative responses from post-study surveys reflected high satisfaction with both usability and system performance, with most participants agreeing that the interface was intuitive and the feedback timely, informative, and confidence-building.

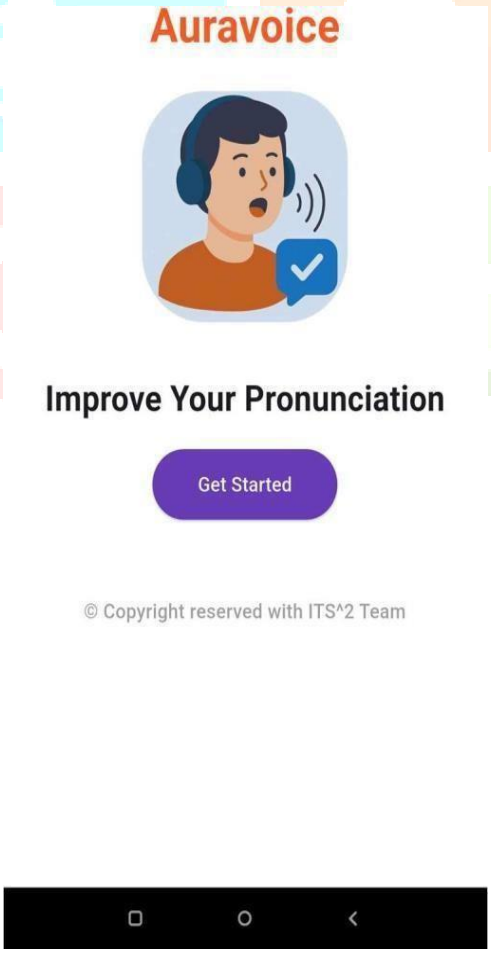

Fig.3 Home Page

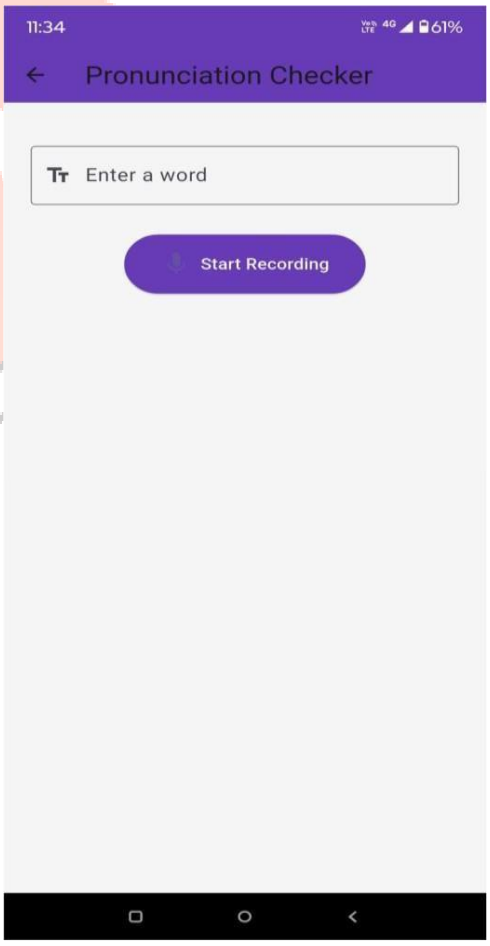

Fig.4 User Entering a Word

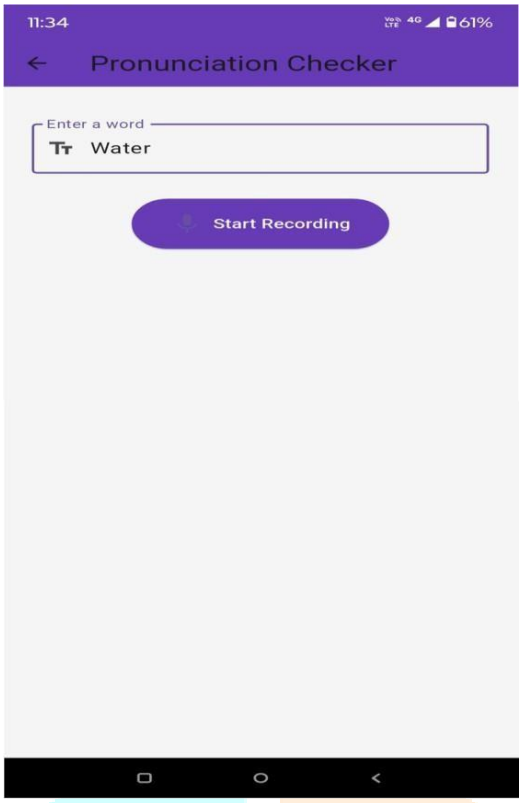

Fig.5 User Pronouncing a Word

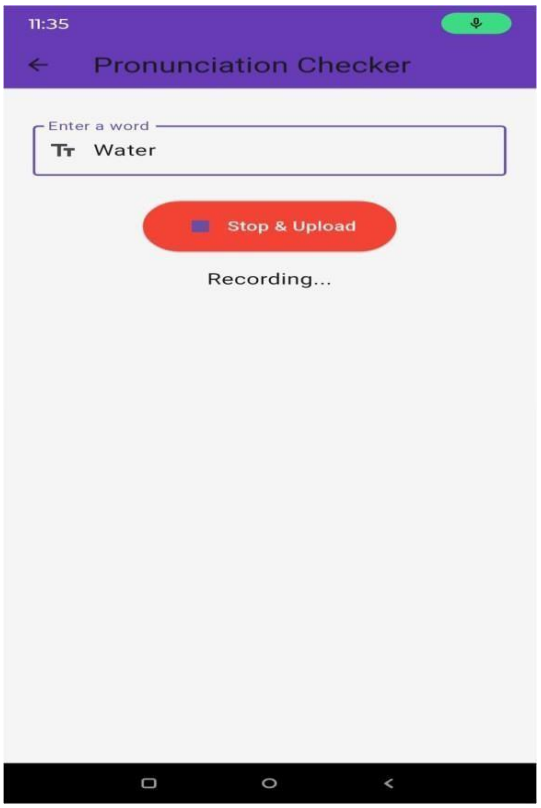

Fig.6 Analyzing User's Pronunciation

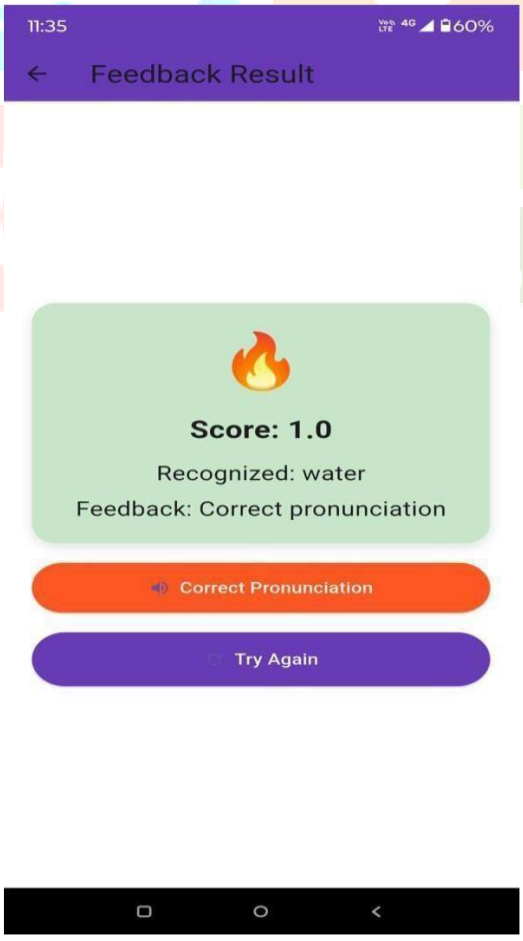

Fig.7 Output for Correct Pronunciation

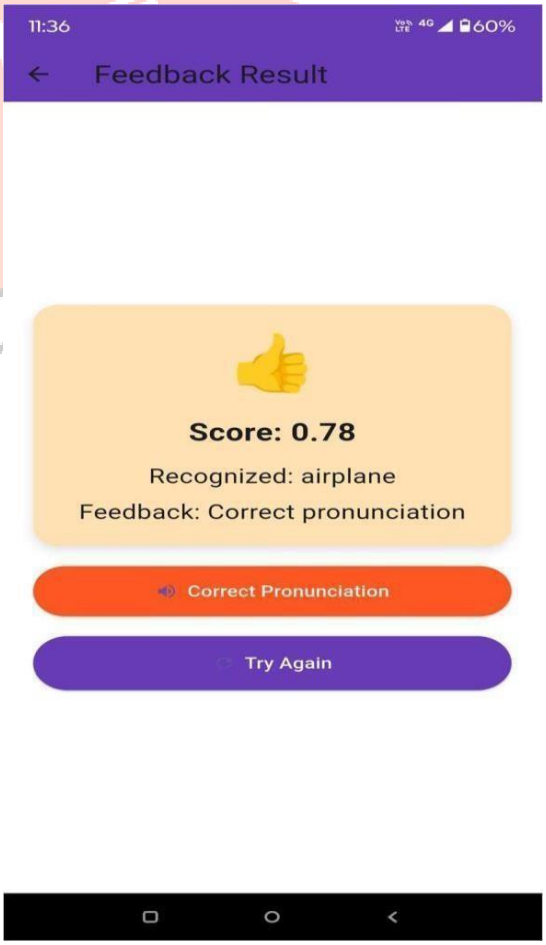

Fig.8 Output for Partially Correct Pronunciation

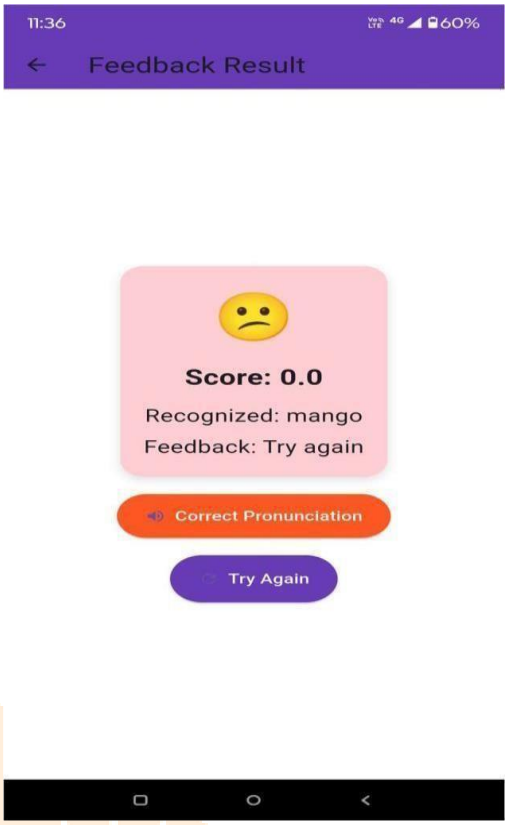

Fig.9 Output for Incorrect Pronunciation

IV. CONCLUSION

The present work introduces AuraVoice, an intelligent mobile-based framework designed for automated pronunciation error detection and correction. The system integrates multiple artificial intelligence and signal-processing components, including SpeechRecognition, Librosa, the CMU Pronouncing Dictionary, and Levenshtein Distance algorithms, to perform detailed phoneme-level analysis of spoken English. These components are combined with a cross-platform Flutter interface and a lightweight backend powered by Flask and SQLite, enabling AuraVoice to deliver a responsive, offline-capable, and user-centric pronunciation training environment suitable for diverse learning contexts.

Experimental evaluation demonstrates that AuraVoice achieves an overall detection accuracy of 90.6% with a Phoneme Error Rate (PER) of 8.1%, outperforming traditional pronunciation assessment systems such as the Google Speech API, EESEN Toolkit, and standard Levenshtein-based matchers. The system processes user input with an average latency of approximately 1.2 seconds, confirming its effectiveness for real-time pronunciation feedback. This low-latency performance, coupled with high reliability across varying acoustic conditions and user accents, highlights AuraVoice's capability to support continuous and interactive learning experiences.

In addition to quantitative validation, qualitative findings from a one-week user study confirm AuraVoice's educational effectiveness. Participants showed consistent improvement in pronunciation accuracy, increasing from 78% at baseline to 91% after sustained usage. Users reported that visual cues and real-time audio feedback significantly enhanced pronunciation awareness and facilitated faster correction of articulation errors. These results demonstrate that AuraVoice not only meets technical performance benchmarks but also positively impacts learner outcomes, reinforcing its value as a practical assistive learning tool.

Architecturally, AuraVoice emphasizes accessibility, scalability, and adaptability through a modular design that allows seamless extension to additional languages, datasets, and speech-analysis modules. Its offline functionality ensures usability in resource-constrained environments, while intuitive visualization and phoneme-highlighting mechanisms provide a pedagogically effective and technologically robust user experience. Future enhancements will focus on incorporating transformer-based acoustic models such as

Wav2Vec 2.0 and HuBERT to enable multilingual and accent-aware analysis, expanding datasets to include diverse non-native accents, and introducing sentence-level fluency, intonation scoring, and context-aware mispronunciation tracking. In conclusion, AuraVoice represents a convergence of advanced AI techniques and applied language pedagogy. By combining algorithmic precision with accessibility and learner-focused design, it lowers barriers to effective pronunciation training and contributes to the democratization of spoken-language learning, establishing a strong foundation for future innovation in intelligent language education systems.

REFERENCES

[1] R. Anantha, K. Bhasin, D. de la Parra Aguilar, P. Vashisht, B. Williamson, and S. Chappidi, "DTW-SiameseNet: Dynamic Time Warped Siamese Network for Mispronunciation Detection and Correction," arXiv preprint arXiv:2303.00171, 2023.

[2] R. Yenuganti, S. S. S. N. U. Devi, B. S. S. R. Krishna, S. S. S. R. R. Kumar, and K. V. S. R. K. Prasad, "Pronunciation Error Detection and Correction," SSRG Int. J. Comput. Sci. Eng., vol. 10, no. 12, pp. 1–5, 2023.

[3] S. V. S. S. R. K. Prasad, B. S. S. R. Krishna, S. S. S. R. R. Kumar, and K. V. S. R. K. Prasad, "Automatic Pronunciation Mistake Detector Using Python," Int. J. Innov. Res. Technol., vol. 9, no. 7, pp. 1–5, 2023.

[4] D. Korzekwa, "Automated Detection of Pronunciation Errors in Non-Native English Speech Employing Deep Learning," arXiv preprint arXiv:2209.06265, 2022.

[5] Z. Zhang, Y. Wang, and J. Yang, "Masked Acoustic Unit for Mispronunciation Detection and Correction," arXiv preprint arXiv:2108.05517, 2021.

[6] B.-C. Yan and B. Chen, "End-to-End Mispronunciation Detection and Diagnosis From Raw Waveforms," arXiv preprint arXiv:2103.03023, 2021.

[7] N. Baranwal and S. Chilaka, "Improved Mispronunciation Detection System Using a Hybrid CTC-ATT Based Approach for L2 English Speakers," arXiv preprint arXiv:2201.10198, 2022.

[8] J. Shi, N. Huo, and Q. Jin, "Context-aware Goodness of Pronunciation for Computer-Assisted Pronunciation Training," arXiv preprint arXiv:2008.08647, 2020.

[9] M. Shahin, J. Epps, and B. Ahmed, "Phonological-Level wav2vec2-based Mispronunciation Detection and Diagnosis Method," arXiv preprint arXiv:2311.07037, 2023.

[10] Y. El Kheir, S. A. Chowdhury, A. Ali, H. Mubarak, and S. Afzal, "SpeechBlender: Speech Augmentation Framework for Mispronunciation Data Generation," arXiv preprint arXiv:2211.00923, 2022.