END-TO-END PRONUNCIATION DETECTION USING NON-AUTOREGRESSIVE NEURAL MODELS

DR. SUJEET MORE, Trinity College Of Engineering And Research, Pune-48, Maharashtra, India
ANITA MARVAR, Trinity College Of Engineering And Research, Pune-48, Maharashtra, India
SAKSHI MAVALKAR, Trinity College Of Engineering And Research, Pune-48, Maharashtra, India
UTKARSHA MOHITE, Trinity College Of Engineering And Research, Pune-48, Maharashtra, India
SAURABH JAGTAP Trinity College Of Engineering And Research, Pune-48, Maharashtra, India

❖ ABSTRACT

Automatic Pronunciation Error Detection (APED) serves as a vital component in language learning systems by offering users immediate feedback on how accurately they pronounce words. Traditional systems often depend on autoregressive models and manual feature engineering, which introduces latency and limits scalability. This paper proposes a non-autoregressive, end-to-end deep learning framework designed to improve efficiency by removing dependencies on sequential output generation. The system incorporates powerful audio feature extractors such as Mel-Frequency Cepstral Coefficients (MFCCs) and Wav2Vec, and employs a Transformer-based encoder-decoder model to identify and evaluate mispronunciations. The architecture supports real-time audio processing and classifies user input as either correctly or incorrectly pronounced. Comparative analysis reveals that this design reduces processing delays while maintaining a high level of accuracy, making it suitable for deployment in scalable language education platforms.

Keywords:
Non-Autoregressive Networks, Pronunciation Evaluation, Speech Processing, Transformer Architecture, MFCC, Wav2Vec.

❖ I. INTRODUCTION

Recent advances in deep learning have significantly enhanced the performance of speech-based technologies, especially in the domain of language learning. A critical element of these systems is Automatic Pronunciation Error Detection (APED), which assists learners by offering immediate, precise feedback on how well they articulate words. Historically, many such systems have relied on autoregressive models that process inputs sequentially, where each output is conditioned on prior predictions. While these models are effective, their sequential nature results in slower response times and increased computational demands—challenges that hinder real-time deployment.

To address these shortcomings, this project presents a non-autoregressive, end-to-end neural network for detecting pronunciation errors. By removing the dependency on sequential processing, the model supports parallelization, leading to faster inference and greater efficiency. The architecture leverages robust feature extraction methods like MFCCs and Wav2Vec and uses a Transformer-based encoder-decoder setup to learn acoustic and linguistic patterns necessary for detecting mispronunciations accurately.

❖ II. PROBLEM STATEMENT

Most existing pronunciation assessment tools are based on autoregressive models that predict outputs in a step-by-step manner, where each prediction depends on its predecessors. This design leads to slow processing speeds, which makes delivering real-time pronunciation feedback a challenge—especially in interactive learning environments. Moreover, these systems often rely on manual feature extraction, which increases complexity and resource usage.

This project aims to overcome these issues by developing a real-time pronunciation error detection model that eliminates sequential dependency and automates feature extraction using modern deep learning techniques. The ultimate goal is to improve user experience by providing fast and accurate feedback without compromising model efficiency or scalability.

❖ II. OBJECTIVE

The primary objectives of this project are:

- To design and implement a non-autoregressive, end-to-end neural network for accurate pronunciation error detection.
- To incorporate automatic and advanced audio feature extraction techniques such as MFCCs and Wav2Vec to minimize manual preprocessing.
- To utilize a Transformer-based encoder-decoder architecture for effectively modeling acoustic and linguistic patterns in speech.
- To enable real-time classification of spoken words into correct or incorrect pronunciation categories.
- To benchmark the system's performance against traditional autoregressive models, focusing on speed, accuracy, and scalability.

To develop an end-to-end system for automatic pronunciation evaluation without relying on traditional phoneme alignment or rule-based methods.

To utilize non-autoregressive neural models (e.g., Transformer-based architectures) for faster and more efficient pronunciation detection compared to autoregressive models.

To detect and classify pronunciation errors such as insertions, deletions, and substitutions at the phoneme or word level.

To eliminate the need for forced alignment or manual feature engineering by leveraging direct acoustic-to-evaluation mappings.

To improve inference speed and reduce latency in pronunciation scoring systems using non-autoregressive decoding techniques.

To enhance the robustness of pronunciation detection across speakers of different accents, ages, and fluency levels.

To integrate contextual and temporal modeling without depending on sequential (left-to-right) decoding as in traditional models.

To evaluate model performance on standard pronunciation assessment datasets using metrics like accuracy, precision, recall, and phoneme error rate (PER).

To explore data augmentation techniques for better generalization of pronunciation models on low-resource and multilingual datasets.

To contribute to computer-assisted language learning (CALL) systems by providing real-time, interpretable feedback on spoken pronunciation quality.

This study highlights the potential of non-autoregressive neural models in transforming pronunciation detection by offering faster, scalable, and alignment-free solutions. The proposed approach not only streamlines the evaluation process but also makes pronunciation feedback more accessible and real-time, paving the way for enhanced language learning tools and broader multilingual support.

❖ III. LITERATURE SURVEY

1. Leung, W.K.; Liu, X.; Meng, H. — CNN-RNN

CTC Based End-to-End Mispronunciation Detection

This research explores a deep learning-based end-to-end system for mispronunciation detection using a combination of CNN, RNN, and CTC. CNNs are employed to extract audio features, RNNs capture the sequential aspects of speech, and CTC aligns the predicted outputs with the actual transcriptions. The model effectively handles both spatial and temporal dynamics of speech, improving detection accuracy. However, it requires a large and diverse labeled dataset, which is challenging to compile, especially across various accents and dialects.

2. Wu, M.; Li, K.; Leung, W.K.; Meng, H. Transformer-Based End-to-End Mispronunciation Detection and Diagnosis

This paper introduces two Transformer-based models for detecting and diagnosing mispronunciations. One is built on the standard Transformer architecture, while the other leverages wav2vec 2.0. These models utilize self-attention mechanisms to better understand phonetic contexts and classify pronunciation errors by analyzing correct and incorrect speech samples. The strength of this approach lies in its ability to capture long-range dependencies in speech. Nevertheless, the method depends heavily on the availability of large, annotated speech datasets, which may not always be feasible in multilingual environments.

3. Korzekwa, D. et al. — Computer-Assisted Pronunciation Training (CAPT)

The study focuses on CAPT systems that enable language learners to independently practice pronunciation through digital tools. These systems rely on automatic speech recognition (ASR) to provide immediate feedback, fostering improvement through repeated, self-guided learning sessions. CAPT offers a low-pressure environment conducive to learning but may not offer the same depth of correction and nuance that a human instructor can provide, which could leave some pronunciation issues unaddressed.

4. Higuchi, Y. et al. — Mask CTC: Non-Autoregressive End-to-End ASR (2020)

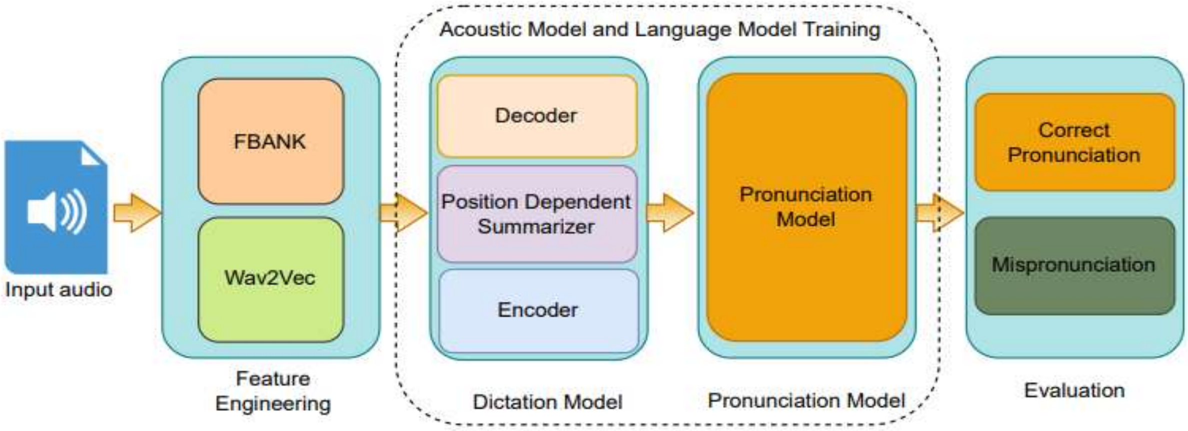
This research proposes a non-autoregressive speech recognition system called Mask CTC, which combines Connectionist Temporal Classification (CTC) with masked language modeling. The model first aligns speech and text using CTC, then refines predictions by filling in masked parts, allowing for simultaneous output generation. While this significantly reduces inference time and supports real-time applications, its decoding strategy may lower accuracy when processing complex or lengthy speech inputs.

5. Sudhakara, S. et al. — Improved GoP for DNN-HMM Based Pronunciation Evaluation

This paper enhances the Goodness of Pronunciation (GoP) metric by integrating Hidden Markov Model (HMM) transition probabilities into the Deep Neural Network (DNN)-HMM framework. The improved GoP score more accurately evaluates pronunciation quality by considering dynamic speech transitions. Though the method improves evaluation reliability, it remains dependent on the DNN-HMM system's alignment quality, which means errors in model alignment can still yield misleading feedback.

❖ IV - PROPOSED METHODOLOGY

➢ SYSTEM ARCHITECTURE

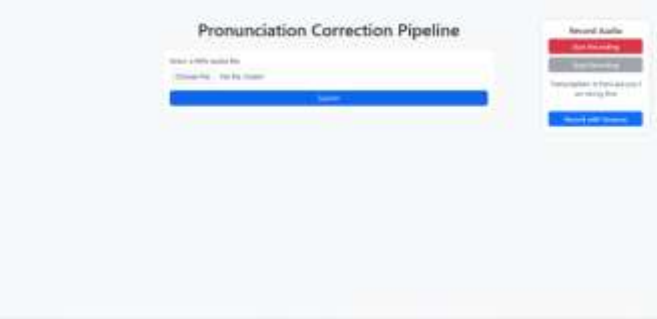


The proposed pronunciation error detection system begins with an audio input, where users either record or upload their speech sample. From this input, key features are extracted using FBANK and Wav2Vec. FBANK helps capture frequency-specific energy patterns, essential for analyzing speech, while Wav2Vec generates dense, context-rich vectors from raw audio, making it suitable for machine learning tasks.

These extracted features are then passed into a dictation model. The encoder first converts the features into high-level representations. A position-based summarizer then emphasizes the location-sensitive aspects of speech to help interpret context accurately. The decoder reconstructs the output, allowing for comparison with the expected pronunciation.

Next, the system performs pronunciation assessment by aligning the encoded features with known correct pronunciations using a phonetic alignment method. This step checks whether the input speech adheres to standard pronunciation norms. Finally, the model classifies the result as either a correct pronunciation or a mispronunciation and provides real-time feedback to the user for improvement.

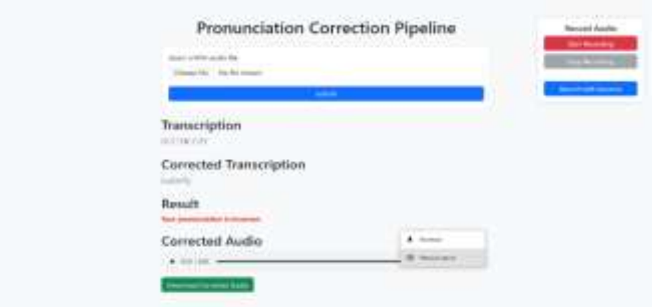
❖ V. RESULTS



Pronunciation Correction Pipeline

Fig. 1 Result of Transcription

The Pronunciation Correction Pipeline allows users to upload a WAV audio file or record speech for pronunciation evaluation. Upon uploading, the system processes the file, transcribes the speech, and checks for errors, providing a corrected transcription and downloadable corrected audio if needed. Users can also record their pronunciation in real-time using the built-in Start Recording button, which captures speech, transcribes it, and displays the text. Additionally, the "Record with Vocaroo" button redirects users to an external recording platform for alternative audio input. A Machine Learning model processes the speech, compares it with the correct pronunciation, classifies it as correct or incorrect, and provides instant feedback. The interface is simple and user-friendly, ensuring ease of use while offering multiple ways to improve pronunciation.



Pronunciation Correction Pipeline

Transcription

Corrected Transcription

Result

Corrected Audio

Fig. 2 Result of Pronunciation Detection.

The Pronunciation Correction Pipeline processes an uploaded WAV audio file or recorded speech, transcribes it, and evaluates pronunciation accuracy. In this case, the transcription result ("BUTTER FLEE") differs from the expected pronunciation ("butterfly"), leading to an incorrect pronunciation assessment, indicated in red text. The system then generates a corrected audio version, allowing the user to listen to the accurate pronunciation. A download option enables users to save the corrected audio for reference. The recording section on the right allows users to record their speech directly or use Vocaroo for external recording. This interface ensures real-time feedback and an easy-to-use correction system for pronunciation improvement.

❖ VI. CHALLENGES

Building and deploying a machinelearning-based system for automatic pronunciation error detection (APED) comes with several challenges. These challenges can be grouped into three main areas: technical hurdles, resource limitations, and project scope constraints.

Technical Challenges

1. Quality and Availability of Data

o Problem: The effectiveness of the model depends heavily on having high-quality, annotated datasets that include both correct and incorrect pronunciations. Acquiring such datasets—especially those covering diverse speech patterns—is a major hurdle.

o Example: A system trained mainly on American English might perform poorly when analyzing accents like British or Indian English.

2. Optimizing for Real-Time Feedback

o Problem: To offer near-instantaneous pronunciation feedback, the system must process audio inputs rapidly. Achieving this in a cloud-hosted setup requires a highly optimized backend.

o Example: Delivering responses within seconds—even under high user load—necessitates intelligent scaling and efficient model execution.

3. Feature Extraction Complexity

o Problem: Deriving useful audio features (like MFCCs or Wav2Vec embeddings) from raw audio signals requires significant computation. This can slow the system, especially on low-end devices.

o Example: Processing on entry-level smartphones may lag, affecting the feasibility of providing real-time insights.

4. Accent and Dialect Diversity Problem: Speakers use varied accents and dialects, making it difficult for the system to distinguish between legitimate variations and genuine pronunciation mistakes.

o Example: The model must learn to accept natural accent differences (e.g., UK vs. US English) without mislabeling them as errors.

4. Balancing Accuracy with Speed

o Problem: Sophisticated models typically offer better accuracy but are slower, whereas lightweight models run faster but may miss subtle pronunciation issues.

o Example: Tuning the model to minimize both false positives and false negatives is essential for delivering reliable feedback.

5. Accent and Dialect Diversity Problem: Speakers use varied accents and dialects, making it difficult for the system to distinguish between legitimate variations and genuine pronunciation mistakes.

o Example: The model must learn to accept natural accent differences (e.g., UK vs. US English) without mislabeling them as errors.

6. Balancing Accuracy with Speed

o Problem: Sophisticated models typically offer better accuracy but are slower, whereas lightweight models run faster but may miss subtle pronunciation issues.

o Example: Tuning the model to minimize both false positives and false negatives is essential for delivering reliable feedback.

Resource Challenges

1. High Computational Demands

o Problem: Training deep learning models like Wav2Vec and Transformers requires significant computing power. Real-time inference also demands consistent GPU/CPU resources, which can be expensive.

o Example: Although cloud platforms like AWS and GCP offer scalable compute options, continuous usage may lead to high operational costs.

2. Managing Storage Efficiently

o Problem: The system must store large volumes of audio files, extracted features, and user-specific progress data, which requires a well-structured storage strategy.

o Example: If the backend databases are not optimized, they can become slow and overloaded, affecting performance.

Project Scope Challenges

1. Generalizing Across Languages

o Problem: Initially focusing on a single language or accent simplifies development, but scaling to multilingual and multidialect support adds complexity in terms of model structure and training data.

o Example: Expanding from English to include other languages would necessitate significant adjustments in both datasets and algorithms.

2. Managing User Expectations

o Problem: Users may expect flawless performance from the system, especially in early versions. It's crucial to communicate that the tool is an aid—not a perfect evaluator.

o Example: Some misclassifications are inevitable, and users should be informed that the system is continually learning and improving.

❖ VII. CONCLUSION

➢ This project presents a machine learning-driven system for automatic pronunciation error detection, aimed at enhancing the language learning experience by offering immediate, accurate feedback. Utilizing state-of-the-art models like **Wav2Vec 2.0** and **Transformer-based architectures**, the system successfully identifies pronunciation mistakes and provides corrective suggestions. Key milestones, including the definition of clear objectives, thorough analysis of both user and system needs, and the development of a scalable architecture (featuring feature extraction, backend APIs, and cloud integration), have been completed — setting a strong foundation for implementation, testing, and deployment in practical environments.

➢ Looking toward future developments, the system holds potential for **multilingual and multi-dialect support**, making it accessible to a broader, global user base. Future upgrades may involve **personalized learning features**, where AI tailors feedback to each user's specific pronunciation habits. Moreover, integration with platforms such as **Duolingo, Coursera**, or virtual assistants like **Google Assistant** could elevate interactive learning experiences. Continued improvements in processing speed, adaptability to various accents, and the precision of feedback will further establish the system as a powerful and reliable tool for language learners around the world.

❖ VIII. REFERENCES

[1] Schneider, S., Baevski, A., Collobert, R., & Auli, M. (2019). *Wav2Vec: A technique for unsupervised learning in speech recognition*. Presented at Interspeech.

[2] Baevski, A., Zhou, H., Mohamed, A., & Auli, M. (2020). *Wav2Vec 2.0: A self-supervised architecture for learning speech representations*. In Proceedings of NeurIPS.

[3] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., et al. (2017). *"Attention is All You Need": Introducing the Transformer model*. Published in NeurIPS proceedings.

[4] Mohamed, A., Hinton, G., & colleagues. (2011). *Utilizing Deep Belief Networks for acoustic modeling in speech systems*. Published in IEEE Transactions on Audio, Speech, and Language Processing.

[5] Kingma, D. P., & Ba, J. (2015). *Adam: An optimization algorithm for stochastic gradient descent*. Preprint available on arXiv:1412.6980.

[6] Graves, A., Mohamed, A., & Hinton, G. (2013). *Enhancing speech recognition using Deep Recurrent Neural Networks*. Featured in the ICASSP conference proceedings.

[7] Chorowski, J., Bahdanau, D., Serdyuk, D., Cho, K., & Bengio, Y. (2015). *Incorporating attention mechanisms in speech recognition models*. NeurIPS conference paper.