WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Automatic Pronunciation Mistake Detector

Aditi Tiwari[1], Abdul Sameer[2], Dr. Gousiya Begum[3]

Department of CSE, MGIT (A), Gandipet ,Hyderabad ,500075 ,Telangana ,India
Email:atiwari_cse210566@mgit.ac.in,asameer_cse210565@mgit.ac.in, gousiyabegum_cse@mgit.ac.in

A B S T R A C T

Traditional English Education System emphasizes on writing and grammar instead of vocal language learning. As a result of which many Chinese students perform well on written exams but only few can speak English successfully in everyday situations. It is critical to develop a scientific and efficient automatic correction system for English pronunciation errors. To help students practice their pronunciation outside of school, here as a project an English Pronunciation Tool is presented. Our Automatic Pronunciation Mistake Detector project is an efficient automatic correction system for English pronunciation errors. It is designed to enable students/users to improve their pronunciation skills. By using Speech Recognition, pyaudio, pyttsx3, the project aims to efficiently diminish the error rate and enhance the accuracy of error detection. For pronunciation detection the user would need to select the word/sentence they want to pronounce and click the button to hear it is correct pronunciation. Along with pronunciation there is also a provision to check the grammar and spelling mistakes in the text. Our approach focuses on extracting key information and generating natural language description about it that are both coherent and accurate.

Keywords: Vocal language learning, efficient, Pronunciation errors, Speech recognition, pyaudio, pyttsx3, error rate, accuracy, spelling mistakes

1. Introduction

Machine learning allows computers to learn to make intelligent choices by accessing the learning of all of the examples rather than being programmed step by step, in a facilitated way. Systems like TensorFlow and PyTorch enable developers to create these learning systems easier now than before. While the world is more interconnected and English become a common language for communication, students need to practice speaking in order to develop fluent speaking skills. However, with the limited classroom time, there's less time to get immediate feedback when they mispronounce a word or phrase, particularly if they are just using pictures and recordings to practice enunciation. With the use of technology now being increasingly commonplace, computers are being used in education and act as a companion to support learning in all possible means. While many language learning A.I.T. and software-based systems focus on applying grammar and understanding of words, they do not improve students' speaking abilities, like their pronunciation, and therefore they make little impact on students' speaking skills [1]. Pronunciation is one of the most important parts of speaking in a way that enables clarity of communication and draws confidence from the user. As the world becomes more global, English has become a language of communication. For students to speak fluently, one must practice forming speech sounds (pronunciation) regularly, preferably with quick feedback. However in many classrooms, including China students are often concerned with and will put most of their time into reading and writing, rather than their pronunciation [2]. When practicing pronunciation in isolation, students may not even know if they are speaking correctly, because there is no feedback (real-time) immediately after speaking. For this reason, the need for pronunciation detection tools, and pronunciation correction tools, that can help learners and practice pronunciation detection and correction automatically is increasing. As technology continues to advance, learning with computers has become pretty common. Lots of language software helps with grammar and comprehension, but not many tools focus on improving speaking skills. This project introduces a smart system that uses machine learning—specifically a better version of the Random Forest algorithm—to spot and correct pronunciation mistakes. It uses special sound features called MFCC and simplifies data complexity with PCA to boost accuracy. The system provides clear, straightforward feedback, helping learners pronounce words more clearly and correctly. The goal of this project is to help people improve their pronunciation using a speech recognition system. It listens to what you say, compares it to the correct pronunciation, and gives feedback on how well you spoke. If there are errors, it suggests fixes—not just for the words, but also for pitch and tone. By checking both what was said and how it was said, the system can offer more accurate and helpful suggestions [3]. Then, it rates your pronunciation as "Exactly correct," "Nearly correct," "Slightly correct," or "Wrong." This tool is especially useful for language learners, speech therapists, or anyone wanting to get better at speaking English. As more people around the world try to pick up new languages, online learning tools have really taken off. One of the most helpful tools is Computer-Assisted Pronunciation Training (CAPT), which is designed to help learners say words more accurately. A big part of CAPT is Automatic Pronunciation Error Detection (APED) — a system that spots and emphasizes mistakes in how people pronounce words. In the past, traditional APED systems relied on complex speech recognition setups, including dictionaries and models [5].

1.1 Problem Statement

In an increasingly globalized world, English has become the primary medium for communication across academic, professional, and social domains. Despite its importance, mastering spoken English remains a significant challenge, particularly for non-native speakers. Traditional language education focuses heavily on grammar, vocabulary, and written skills, often neglecting pronunciation and spoken fluency. This imbalance leaves learners proficient in theoretical aspects of the language but ill-equipped to communicate effectively in real-world scenarios. Conventional methods, such as listening to audio recordings and mimicking pronunciation, fail to provide personalized feedback, making it difficult for learners to identify and correct their mistakes. Furthermore, these approaches are generic and do not cater to individual linguistic challenges or diverse accents, creating a gap in the learning process.

1.2 Existing System

Existing technological solutions also have limitations. Many rely on simple algorithms that detect errors but lack the ability to offer detailed analyses or actionable suggestions for improvement. Others require extensive datasets to train machine learning models, making them less accessible in resource-constrained regions. Additionally, these tools often fail to account for the nuances of regional accents, reducing their effectiveness for a global audience. To address this gap, an efficient and accessible Automatic Pronunciation Mistake Detector is needed—one that uses artificial intelligence and speech recognition to analyze pronunciation errors, provide real-time feedback, and offer tailored suggestions for improvement. This system aims to empower learners to enhance their spoken English skills independently and effectively, bridging the gap between theoretical knowledge and practical fluency.

1.3 Proposed System

The project aims to develop an Automatic Pronunciation Mistake Detector that identifies pronunciation errors, provides real-time feedback, and offers corrective suggestions. By leveraging advanced technologies such as artificial intelligence, speech-to-text APIs, and phoneme comparison algorithms, the system bridges the gap between spoken and written language proficiency.

How it improves on existing solution:

• **Real-time Feedback with Detailed Analysis**: The system provides learners with immediate, detailed feedback that highlights specific problems such as intonation, stress, or phoneme articulation. This enables them to make corrections right away and advance more quickly.

• **Customization Based on Needs**: Unlike generic solutions, it conforms to the user's native accent and linguistic background to provide tailored feedback that addresses particular pronunciation issues.

• **Comprehensive Error Analysis**: The system helps students concentrate on areas that need the most work by classifying errors, such as mispronounced vowels or consonants, in addition to flagging them.

• **Less Reliance on Big Datasets**: The solution makes use of fine-tuned, pre-trained machine learning models, which eliminates the need for large datasets and increases accessibility in places with limited resources.

• **Accessibility and Cost-Effectiveness**: It operates on common devices, such as PCs, providing an affordable pronunciation training.

2. Literature Review

[1] Vinita Vikram Patil, Pratiksha Mohan Wadkar, Aishwarya Bhagwan Mohite, Ms. Priyanka Rajendra Jadhav –"International Journal of Creative Research-Thoughts (IJCRT)", 2024

This paper presents a method for automatic pronunciation detection that detects and fixes pronunciation errors using machine learning and speech recognition techniques. The study demonstrates how this approach, which offers real-time feedback, enhances language learners' spoken English abilities.

[2] Zhang Shufang – "Design of an Automatic English Pronunciation Error Correction System Based on Radio Magnetic Pronunciation Recording Devices",2021

*I*n this paper the focus is on a system using radio magnetic pronunciation recording devices to correct English pronunciation errors. The proposed solution ensures high accuracy in detecting mispronunciations.

[3] Nevon Projects– "Pronunciation Detection and Correction Tool",2020

In this paper, the correction tool uses deep learning and audio processing to correct the pronunciation of the learner. It is a practical tool for learning pronunciation, although it does not provide much feedback for the pronunciation of phonetic variations, and also requires an extensive dataset that is relatively inaccessible to an average user.

[4] S.M. Witt– "Phone-Level Pronunciation Scoring and Assessment for Interactive-Language-Learning",2020

This paper explores phoneme-level pronunciation scoring, where a system assesses individual sounds in spoken English. While effective for language learning applications, the system's computational requirements are high, and it lacks adaptability to more complex, modern AI-driven systems**.**

[5] Gang Zhang – "Quality Evaluation of English Pronunciation Based on Artificial Emotion Recognition and Gaussian Mixture Model",2021

In this paper, a new research on machine learning component in recognizing English pronunciation by applying emotional recognition and Gaussian mixture model is introduced. Such approach has been shown to greatly improve the system accuracy in scoring pronunciation.

[6] Fengming Jiao, Xin Zhao, Jiao Song – "Spoken English Teaching System Based on Speech Recognition and Machine Learning",2021

This paper presents a system that combines speech recognition and machine learning for spoken English teaching. It offers an efficient way to teach pronunciation but is more focused on general language learning systems rather than detailed pronunciation error correction.

[7] Liang Meng, Jinal Upadhyay, Sumit Kumar et al. – "Nonlinear Network Speech Recognition Structure in a Deep Learning Algorithm",2022

This paper emphasizes on the deep learning models developed for nonlinear speech recognition especially pronunciation detection. This model, despite its novelty, has high computational costs and is hardware dependent hence making it hard to be implemented for general use.

[8] Cai Xiao-Ming – "Automatic Detection of English Pronunciation Errors Based on Statistical Pattern Recognition",2020

In this study, a statistical methodology was introduced to detect English pronunciation mistakes. It is useful for getting mistakes at a beginner level. Still, it has the limitation of not being able to correct real-time mistakes, so the learners cannot improve English speaking quickly.

[9] Yuhua Dai– "An Automatic Pronunciation Error Detection and Correction Mechanism in English Teaching Based on an Improved Random Forest Model",2022

In this paper, we propose an improved random forest model for automatic pronunciation error detection. While it offers more accurate error detection, it struggles with complex pronunciation issues and requires large datasets for effective model training.

[10] Lei Chen, Chenglin Jiang, Yiwei Gu, Yang Liu, Jiahong Yuan – "Automatically Detecting Reduced-Formed English Pronunciations Using Deep-Learning",2022

This research based on deep learning techniques of reduced form English pronunciation detection provides insights into understanding pronunciation, but it is limited to specific English speeches and does not have a broader applicability to a general set of language learners.

3. System Architecture

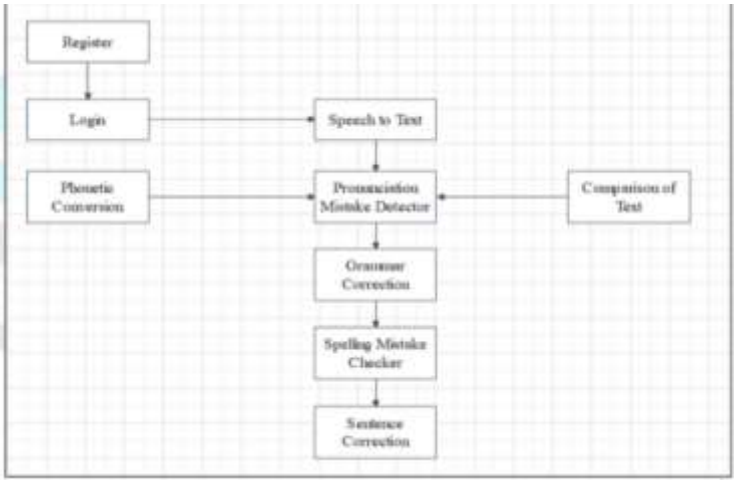

Figure 1 System Architecture that shows the process of a speech-based language learning system

Figure 1 displays a flowchart demonstrating how a speech-based linguistic system works, as it transforms speech to text, checks errors in pronunciation and grammar, and corrects the spellings and sentence structures after user login and phonetic analysis.

There are some important parts of the system. User Authentication is a part that allows new users to access the system by giving their credentials and allow different users to access the system. Once users log in, speech processing begins with a speech recognition component that converts spoken input to text. Phonetic processing then converts the speech into phonemes. In the Error Detection and Correction Pipeline, the pronunciation mistake detector compares the spoken phonemes to correct references, and the comparison of text checks. The text that was read against the content that was spoken. Grammar correction then checks for mistakes in the verb tense, subject-verb agreement, etc. The spelling mistake checker checks for any spelling errors. Finally, sentence correction combines grammar and spelling mistakes to correct the user input.

4. Algorithmic Workflow for Processing Input Data

User Interaction (Frontend - Tkinter):

The user interacts with the graphical interface, primarily by:

1)Entering text into the "Enter Sentence to Practice" field.

2)Clicking the "Check Grammar" button.

3)Entering text into the "Type a word/sentence to hear its pronunciation:" field.

4) Clicking the "Check Pronunciation" button.

5) Interacting with registration and login windows.

Grammar and Spell Check Workflow (Backend - Python, language_tool_python, autocorrect):

When the "Check Grammar" button is clicked:

The text entered in the "Enter Sentence to Practice" field is retrieved by the Python backend.

Spell Checking (autocorrect): The input text is processed word by word to identify and suggest corrections for potential misspellings.

Grammar Checking (language_tool_python): The (potentially spell-corrected) text is analyzed for grammatical errors based on the selected language (currently primarily English).

The identified errors, suggestions, and a corrected sentence are formatted and displayed back to the user in the GUI's result label.

Pronunciation Check Workflow (Backend - Python, pyttsx3, phonemizer):

When the "Check Pronunciation" button is clicked:

1) The text entered in the "Type a word/sentence to hear its pronunciation:" field is retrieved by the Python backend.

2) Text-to-Speech (pyttsx3): The backend initializes the TTS engine and uses the selected language's voice to synthesize and play the audio pronunciation of the entered text. This might happen in a separate thread to prevent UI blocking.

3) Phonetic Transcription (phonemizer**):** If the selected language is English, the backend uses the phonemizer library to generate a phonetic representation of the input text.

4) A new window is created (using Tkinter) to display the original text alongside its phonetic transcription.

User Registration Workflow (Backend - Python, hashlib, MySQL):

When a user interacts with the registration window and clicks "Register":

1) The username and password entered by the user are retrieved.

2) The password is hashed using hashlib (SHA-256) for security.

3) The username and the hashed password, along with other registration details, are sent to the MySQL database to be stored in the users2 table.

4) The application checks if the username already exists in the database to prevent duplicates.

User Login Workflow (Backend - Python, hashlib, MySQL):

When a user interacts with the login window and clicks "Log In":

1) The username and password entered by the user are retrieved.

2) The entered password is hashed using the same algorithm (SHA-256).

3) The application queries the MySQL database to find a user with the entered username.

4) The hashed entered password is compared with the hashed password stored in the database for that user.

5) If the hashes match, the login is successful, and the main application interface is displayed.

Natural Language Processing (NLP): Natural Language Processing (NLP) falls under Artificial Intelligence (AI) domain, focusing on both interpreting and generating human language by machines. This way, it brings different machine functionalities in contact with human language, making it understandable.

How NLP is Used in this Project:

1) For Grammar Checking- LanguageTool uses a rule-based system mainly. This means it has a vast set of predefined rules that describe correct and incorrect grammar for various languages. These rules are usually based on linguistic expertise and analysis of common errors.

How Python (language_tool_python) Detects Errors:

Tokenization: The passage is divided into individual words and punctuation marks.

Part-of-Speech Tagging (POS Tagging): Every token is categorized based on grammar (e.g. nouns, verbs, adjectives, etc.). It often relies on statistics for tagging, based on the context in a sentence, in LanguageTool.

Rule Matching: The sequence of labeled tokens is then checked against a large set of grammar rules. These can describe incorrect patterns. For example, a rule could say that an article must be followed by a consonant. This would be triggered by a rule that says "a apple."

Mistake Noticing: When a sequence of tokens matches an error rule, LanguageTool marks it as a grammatical error.

Suggestion Generation: The rules often include information on how to correct the identified error, leading to the suggestions provided by the library.

2) For Spelling Correction: autocorrect works by using a probability-based model that incorporates n-gram language models and edit distance.

How Python (autocorrect) Detects and Corrects Errors:

Word Tokenization: The input text is split into individual words.

Candidate Generation: Autocorrect will generate a list of possible corrections for each word by looking at words that are a certain number of removals, additions, transpositions, or substitutions away from the original word.

Probability Ranking: The list of possible suggestions gets ranked as per the likelihood of the words appearing in the English language. The probability of a given word is determined by the n-gram language model (a library or list of words and the likelihood of their occurrence) used by autocorrect. Words that are common will have a higher probability ranking.

Correction Selection: The word that has the highest probability of being the correct one is selected and recommended.

3) For Pronunciation Checking: These technologies use advanced speech recognition systems and linguistic models that are trained on extensive audio materials. In general, they incorporate Hidden Markov Models (HMMs) and Deep Learning Networks (DNNs).

How Python (speech_recognition) Detects Spoken Words:

Audio Processing: Voice recording analysis involves retrieving sound properties such as frequency, time, and intensity.

Acoustic Modeling: These characteristics are compared to the acoustic models to determine the most probable collection of phonemes.

Language Modeling: The sequence of phonemes that results is sent to the language model which predicts the most likely sequence of words based on the syntactic and statistical patterns of the target language.

Transcription is the process of converting spoken words or audio into written text.

5. Results

INPUT TABLE

INPUTS	VALUES
User ID	user123
Language	English
Target Text	She don't like going to the beach becaus of the sand, and their are often too many peoples their.
User Audio Input	(Audio data or path to audio file - implementation specific)
Pronuncia-tion Model	en-US_standard
Accuracy Threshold	70

OUTPUT TABLE

OUTPUT FIELD	VALUE
Check Timestamp	4/27/2025, 10:17 PM

Detected Errors	Subject-verb agreement: "don't" should be "doesn't"; Spelling: "becaus" should be "because"; Homophone/Subject-verb: "their are" should likely be "there are"; Pluralization: "peoples" should be "people"; Homophone: "their" should be "there".
Suggested Corrections	"doesn't like", "because", "there are", "people", "there"
Corrected Sentence	She doesn't like going to the beach because of the sand, and there are often too many people there.

Acknowledgements and Reference heading should be left justified, bold, with the first letter capitalized but have no numbers. Text below continues as normal.

6.Conclusion

The system is known as Automated Pronunciation Error Detection systems that assist English learners in developing their speaking and writing skills. Essentially, it collects voice input from the user, processes it and then compares the user's pronunciation of spoken words against a standard dictionary with the goal of identifying pronunciation errors. Consequently, it informs the user if his/her pronunciation is correct or not.

Beside just detecting errors, the software could also give custom advice on common errors of pronunciation helping users identify common mistakes and guiding them to improvement. This is especially useful for users looking to achieve fluency and confidence in speaking English. On a technical level, the grammar and spelling check features depend on tools like language_tool_python and autocorrect to identify and correct grammar and typographical errors in written input. For the pronunciation analysis aspect, it is possible that the system relies on pyttsx3 for text-to-speech synthesis and phonemizer for phonetic transcription to be able to assess the phonological structure of words effectively.

Moreover, having a user registration and login module is a strong indication that the platform is likely to have personalized learning materials for tracking progress over time, storing learners' feedback, and tailoring lessons based on their performance in previous sections. In this case, a MySQL database is utilized to keep these details secure and expedite the learning process with ease.

In general, the system is a complete digital technology for learning that comprises of speech recognition, natural language processing, and intelligent feedback to help users improve their written and spoken English.

7. Future Scope

One of the most effective ways to grow this system will be the inclusion of support for multiple languages other than English. This will require the use of various language-specific grammar checkers and spell-checkers. It will also need various Text-To-Speech (TTS) tools, including phonetic transcription for the languages in question. This will enable a lot more people to be able to use the tool, especially in multicultural learning environments.

Besides extending its coverage of languages, the tool could be improved by including more advanced natural language processing (NLP) features. By implementing such techniques as part-of-speech (POS) tagging, dependency parsing, and semantic reasoning, the app could go beyond merely pointing out errors on the surface. It would be able to identify more complex grammatical mistakes and offer context-based recommendations, taking into consideration the structure, grammar, and meaning of sentences in order to provide higher-quality feedback.

Another good inclusion can be of user profiles and progress tracking. This means that users will need to sign in and they will be able to track their learning journey which can enable them to correct their errors which are commonly made and they can work on improving them. Another great way to improve the platform would be to add user profiles and progress tracking.

When users can create an account and save their learning history, the system can spot patterns in their mistakes, suggest areas to work on, and give personalized exercises or feedback customized just for them. This kind of personalization makes learning more effective and helps users build their skills over time. On top of that, building mobile apps for both Android and iOS would make it super easy for people to access the tool right from their phones or tablets. This means they can practice pronunciation and grammar whenever they have a few minutes, whether they're commuting, at home, or anywhere else. It's all about making learning more flexible and convenient, so users stay engaged and keep improving.

References

Vinita Vikram Patil, Pratiksha Mohan, Aishwarya Bhagwan Mohite, "Automatic Pronunciation Mistake Detector", International Journal of Creative Research

Thoughts (IJCRT), 2024

Yuhua Dai, "An Automatic Pronunciation Error Detection and Correction Mechanism in English Teaching Based on an Improved Random Forest Model",2022

Pushkar Joglakar, Moin Khan, Vedant Mohol, Pranav Modhave, Manas Kadam, "Automatic Pronunciation Mistake Detector Using Python",2024

Long Zhang, Ziping Zhao, Chunmei Ma, Linlin Shan, “End-to-End Automatic Pronunciation Error Detection Based on Improved Hybrid CTC/Attention Architecture”,2020

Shuang Xu, Jie Jiang, Zhenbiao Chen, Bo Xue, Automatic Pronunciation Error Detection Based On Linguistic Knowledge and Pronunciation Space”, 2009

Yassine El Kheir, Ahmed Ali,” Automatic Pronunciation Assessment - A Review”,2023

Renlong Ai,”Automatic Pronunciation Error Detection and Feedback Generation for CALL Applications”,2024

Eric Sanders, Henk Van Den,“ Automatic Pronunciation Mistake Detector in Repitito”,2020

Fengming Jiao,“ A Spoken English Teaching System Based on Speech Recognition and Machine Learning”,2021

ZhangShufang, “Design of an Automatic English Pronunciation Error Correction System Based on Radio Magnetic Pronunciation Recording Device”,2021

Hadeel H. Qassem, Zeena A. Jalal, “The Effect of Spell-Checker Features on Spelling Competence among EFL Learners”, International Journal of English Linguistics, 2022.

Steve Graham, Karen R. Harris, “The Influence of Spell-Checkers on Students’ Ability to Generate Repairs of Spelling Errors”, The Journal of Special Education Technology, 2017.

Ankur Sahni, Anandita Sharma, Sakshi Vats, “Online Grammar Checker Using Neural Networks”, International Journal of Engineering Research & Technology (IJERT), 2021.

Yash Gondaliya, Poojan Kalariya, Brijeshkumar Y. Panchal, Amit Nayak, “A Rule-Based Grammar and Spell Checking”, SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology, 2022.

Madhvi Soni, Jitendra Singh Thakur, “A Systematic Review of Automated Grammar Checking in English Language”, arXiv preprint, 2018.