AI-Powered Pronunciation Mistake Detection Using Gemini 1.5 Flash: A Training-Free Approach

Supritha P O[1]; Omkar Mahale[2]; Shalya Gaonkar[3]; Shetty Aditya Udaya[4]; and Sooraj Devadiga[5];

[1]Assistant Professor, Department of Computer Science & Engineering, Sri Dharmasthala Manjunatheshwara Institute of Technology, Ujire, Karnataka, India
[2,3,4,5] BE Scholar, Department of Computer Science & Engineering, Sri Dharmasthala Manjunatheshwara Institute of Technology, Ujire, Karnataka, India

Correspondence should be addressed to Supritha P O; suprithapoov95@gmail.com

Received: 21 November 2025 Revised: 14 December 2025 Accepted: 28 December 2025

ABSTRACT- Pronunciation accuracy is a fundamental factor in effective language learning; however, many existing systems face difficulties in delivering real-time error analysis without relying on computationally intensive acoustic model training. This paper introduces an AI-driven pronunciation mistake detection system developed using Google Gemini 1.5 Flash, a low-latency multimodal large language model capable of directly processing spoken input. Unlike conventional approaches based on MFCC features or task-specific deep learning pipelines, the proposed system employs prompt-guided reasoning combined with algorithmic scoring methods to detect pronunciation errors at the word, phoneme, and prosodic levels. Learner speech is transmitted to the Gemini API, which generates a structured pronunciation analysis that includes phoneme-level interpretations and word-level discrepancies. These outputs are further processed by a custom scoring framework to evaluate pronunciation quality and produce clear, actionable feedback. Experimental evaluation using diverse English utterances demonstrates the system's effectiveness in identifying vowel–consonant substitutions, omitted syllables, and stress-related errors. The findings underscore the potential of LLM-based audio reasoning as a lightweight, scalable, and real-time solution for automated pronunciation assessment.

KEYWORDS- Pronunciation Error Detection; Gemini 1.5 Flash; Speech Processing; Multimodal Llms; Prompt Engineering; Phoneme Analysis; Real-Time Pronunciation Feedback; Ai-Assisted Learning

I. INTRODUCTION

How clearly someone speaks can shape how well they're understood, how sure they feel when talking, plus how smoothly ideas come across. Still, getting a true measure of speaking clarity is tough work. Old-school tools made to judge speech often fall short and largely depend on acoustic models, manually engineered features, and carefully annotated speech datasets. Most methods hit roadblocks when scaling up, adjusting to different ways people speak, or giving instant responses - this weakens their impact today. Thanks to progress in AI that generates content and models handling multiple data types at once, fresh paths now exist to study spoken words without building custom systems each time. A lean, quick-processing system like Google Gemini 1.5 Flash handles sound effectively across formats, fitting smoothly into live speaking evaluations. Unlike traditional automatic speech recognition (ASR) pipelines, Gemini can process spoken input directly while reasoning over linguistic and phonetic patterns. This capability removes the need to develop, train, or fine-tune conventional ASR or phoneme recognition models. In this work, we propose a complete pronunciation mistake detection framework that combines Gemini's multimodal reasoning with custom-designed scoring algorithms to deliver immediate and meaningful feedback to learners. This paper presents an AI-powered pronunciation mistake detection system that evaluates user speech in real time using Gemini 1.5 Flash. The proposed approach does not involve training any acoustic or phoneme-level models; instead, it relies on carefully designed prompts and algorithmic scoring strategies to guide the model's reasoning process. By comparing Gemini's inferred phoneme sequence with a reference pronunciation, the system detects word-level pronunciation errors, phoneme substitutions, deletions, insertions, and stress-related inaccuracies. The resulting framework provides a scalable, efficient, and interpretable solution for automated pronunciation assessment in language learning applications.

II. RELATED WORK

Algabri et al. [1] introduced a mispronunciation detection and diagnosis framework that integrates phoneme identification with articulatory feature analysis. Their approach formulates pronunciation assessment as a multi-label detection task, allowing the system to identify both phonemes and their associated articulatory properties within a single model. Speech signals are first transformed into spectral image representations, which are then processed using a deep learning–based object detection architecture. This enables the simultaneous recognition of phonemic units and articulatory features, supporting more detailed pronunciation diagnosis and the generation of targeted articulatory feedback for language learners.

Starting off, Zhang, Zhao, and Ma [2] introduced a method that detects mispronunciations automatically through a refined blend of CTC and attention models. This setup weaves pronunciation evaluation straight into speech recognition systems using both Connectionist Temporal Classification and an attention-driven decoder working together. While the CTC part keeps timing in check between spoken sounds and their corresponding units, the attention layer adds adaptability when interpreting sequences. Because these two pieces support each other's strengths, the model spots errors more consistently - skipping extra steps like manual alignment or cleanup afterwards.

Shahin, Epps, and Ahmed [3] presented a phonological-level mispronunciation detection and diagnosis approach built on wav2vec 2.0 representations. Focusing on broader sound structures instead of single speech units helps overcome flaws in standard tools that depend heavily on detailed phonetic labels. These older techniques tend to struggle whenever rare sounds or unusual speaking styles appear. Moving away from strict letter-by-letter analysis allows the new method to adapt better across different speakers. Less need arises for massive sets of narrowly defined markings because patterns emerge more clearly at a wider scale.

A fresh take on spotting mispronunciations comes from El Kheir, Chowdhury, and Ali [4], who blended several angles of the same spoken input. Instead of one fixed sound-based format, their method routes audio through separate streams - some tuned to single languages, others handling many tongues - to gather richer speech details. With training spread across varied tasks at once, it adapts better, catching mistakes more consistently, whether voices differ in accent or native language.

Dong *et al.* [5] investigated improvements to automatic pronunciation assessment (APA) for second-language English learners by incorporating suprasegmental features into a GOPT-style evaluation framework. In addition to conventional segmental pronunciation measures, their approach explicitly models higher-level prosodic characteristics such as stress, rhythm, and intonation. The study demonstrates that integrating suprasegmental information leads to more comprehensive and perceptually aligned pronunciation assessment, particularly for capturing fluency and prosodic quality in L2 speech.

One early tool for spotting mispronunciations came from Jo and team [6], built to help people learning a new language. Instead of relying on intuition, their approach uses sound patterns matched up with individual speech sounds. By lining up each spoken unit against a native speaker's version, differences become visible. Timing shifts, misplaced edges between sounds, and subtle audio mismatches. give clues about errors. Feedback then zeroes in on these deviations, nudging the learner closer to clearer speaking.

One way to look at it starts with Strik et al. [7], who compared several tools meant to catch mispronunciations in language learning software. Instead of just listing results, they tested familiar strategies - like GOP scores, aligning sounds step by step, or judging certainty levels - to see how well each spot mistakes learners make when speaking. What stands out comes from trials using carefully recorded voice samples: every technique shows strengths in catching errors, yet each also stumbles under certain circumstances. Accuracy shifts depending on the person talking; some need more computing power, while none stay ahead, no matter what changes occur during testing.

Starting off, Robertson, Munteanu, and Penn [8] introduced a method to spot mispronunciations tailored for those just starting out learning a new language. Their main concern? Reducing incorrect flags so beginners do not get discouraged by too many mistaken alerts. Instead of simply linking issues together, they focused on precision to protect confidence. The goal wasn't flashy - it was practical: catch errors without overwhelming the user. Though similar systems exist, they paid special attention to early-stage struggles. By narrowing the scope, accuracy improved where it mattered most. Not every mistake needs catching; only the ones worth correcting. Their system integrates acoustic modelling, articulatory feature analysis, and confidence-based scoring to differentiate genuine pronunciation errors from acceptable variations commonly produced by novice speakers. The study highlights the importance of usability and learner-centered design, demonstrating that adaptive decision thresholds and articulatory-informed analysis led to improved detection accuracy while delivering feedback that is more constructive and encouraging.

Strik, Truong, De Wet, and Cucchiarini [9] examined the effectiveness of various machine-learning classifiers for automatic pronunciation error detection in second-language learning contexts. One way to look at it begins with comparing various ways to sort speech patterns - like models built on hidden Markov methods, features drawn from goodness of pronunciation, tree-style sorting systems, along with broader number-driven tactics - to see which best tells right sounds apart from wrong ones. When tested on voices of people speaking a second language, results show accuracy shifts heavily based on what sound traits are used, which particular sounds matter most, and how much speakers differ from one another - making flexibility and smart use of data essential behind any working solution.

Zhu *et al.* [10] proposed a pronunciation error detection framework based on feature fusion, in which multiple sources of information are jointly exploited to improve detection accuracy. Their approach combines acoustic, phonetic, and linguistic features, allowing the system to capture both fine-grained spectral characteristics and higher-level phoneme patterns. By integrating deep neural network representations with selected handcrafted features within an optimised classification scheme, the model achieves more reliable identification of different pronunciation error types, including phoneme substitutions, deletions, and insertions.

Xu *et al.* [11] introduced an automatic pronunciation error detection approach that incorporates linguistic knowledge within a structured pronunciation space to enhance mispronunciation identification. Rather than depending exclusively on acoustic likelihood measures or GOP-based scoring, their method models relationships among canonical phonemes, typical learner pronunciation variants, and phonetic similarity. This representation enables the system to not only detect whether a pronunciation is incorrect, but also to characterise the nature and direction of the learner's deviation from the intended phoneme.

Dai [12] proposed a pronunciation error detection and correction approach for English learners based on an enhanced Random Forest classifier. The method was reported to utilise acoustic features such as pitch information, formant characteristics, and spectral attributes, with modifications to feature weighting and decision thresholds aimed at improving classification performance. A fresh look at how automatic corrections were built using forecasted mistake types was part of the research. Still, the paper got pulled back later, so what it claimed about results and methods needs careful handling now. When old-school machine learning meets speech evaluation, solid proof and open processes matter more than ever.

Hosseini-Kivanani *et al.* [13] examined the performance of ASR-based mispronunciation detection systems for both child and adult learners of English. Their study evaluates the system's ability to identify phoneme-level pronunciation errors using speech recognition models trained on non-native speech data. Multiple acoustic modelling techniques and scoring strategies are compared, including confidence-based measures and likelihood scores obtained through forced alignment. Experimental results indicate that pronunciation error detection is notably more challenging for children due to greater speech variability and developmental factors, whereas adult speech demonstrates more consistent and reliable detection performance.

Zhang, Wang, and Yang [14] proposed an end-to-end mispronunciation detection approach that incorporates a Simulated Error Distance (SED) mechanism to more effectively capture pronunciation deviations. Instead of relying only on hand-labelled mistakes, they add made-up speech changes by tweaking how sounds are encoded while training, which strengthens what the system learns. Because of this, the model can show how close a pronunciation is to correct, not just if it's right or wrong. Using a sound-processing design built on transformers, the method handles small or faint errors better, showing stronger performance where earlier systems might miss them.

Ryu, Kim, and Chung [15] presented a joint multi-task learning (MTL) framework that simultaneously addresses pronunciation assessment, mispronunciation detection, and error diagnosis within a single integrated model. Instead of tackling each job alone, the system uses common sound and speech features tied to separate output sections for different goals. Because tasks share a foundation, insights move between them - so judging pronunciation gets help from tiny sound errors, whereas spotting mispronunciations draws clues from general speaking trends seen in scores. Built through full training on non-native English recordings, this method performs noticeably better than isolated models when measuring correctness and giving useful feedback

III. PROBLEM STATEMENT

Few things matter more in speaking a new language than how words sound, though getting the sounds right often trips people up. Classes packed with students mean less chance to get personal tips on accent, especially when time runs short, and tools fall flat at spotting subtle speech details. Machines that guide pronunciation now exist, but they usually lean on standard voice-recognition tech or measurements tied to how likely a sound matches a known phoneme - like what you see in GOP scores. Still, they struggle when noise is present. Different ways people speak can throw them off, too. They do not handle rhythm or stress well at all. Giving helpful responses turns out to be tricky more often than not. Because of this, live coaching feels out of reach still. Now, newer tools like Gemini 1.5 Flash show promise by using both sound and words together. Listening while reading lets them catch mistakes better. Spotting wrong sounds becomes possible now. Feedback starts sounding less robotic, more natural. Even so, turning such power into something you can actually use isn't straightforward:

- Ensuring reliable capture and preprocessing of learner speech, followed by its conversion into well-structured prompts and the unambiguous interpretation of the model's outputs;
- Identifying pronunciation errors at the word, phoneme, and suprasegmental levels using a model that is not explicitly trained for computer-assisted pronunciation training (CAPT);
- Putting the model's results into a scoring system that stays fair, clear, always works the same way, also makes sense to people who speak different languages;
- Feedback comes through clearly when it connects to how people actually learn a new language. What matters most shows up in moments that guide growth without confusion;
- Clear direction often sticks best if it fits known ways students pick up skills. Thoughtful comments work well when they match learning patterns proven over time. Success hides in details that feel helpful, not overwhelming.

A fresh approach shapes how speech gets checked fast, even when lots of people use it at once. Built to work smoothly whether on phones or browsers, speed stays steady. Structure grows with demand without slowing down. Performance holds firm across devices. Efficiency drives every layer. Quick feedback happens without delay. System handles real-time checks reliably. This time around, the work tackles how to build a tool powered by artificial intelligence that catches mispronunciations automatically. It aims to judge spoken English precisely, spotting sound-level slips along with rhythm and stress issues through deeper analysis. Feedback given to users comes out straightforward, useful, and shaped by insight rather than just matching audio patterns. Instead of leaning on standard speech recognition methods, it pulls strength from Gemini 1.5 Flash's ability to reason about language structure. Speed matters here, so does reliability - the system stays lean without losing performance. Think of it like a guide that listens closely, understands layers in speech, and then scores them clearly. Progress becomes visible because each suggestion ties directly to what was said. Accuracy grows not from guesswork but from structured review across tones, timing, and articulation.

IV. METHODOLOGY

The methodology of the proposed pronunciation mistake detection system is organised into five interconnected stages: data acquisition and preprocessing, prompt formulation with LLM-based pronunciation analysis, error extraction and classification, scoring and evaluation, and feedback generation with user interface integration. Built with a clear purpose, every piece taps into Gemini 1.5 Flash's ability to process different types of information at once. Yet each part stays focused on delivering results that are stable, understandable, and useful for teaching speech sounds. Because clarity matters just as much as function, choices were made with learners in mind - no guesswork needed.

- Data Acquisition and Preprocessing- A fresh start happens when sound enters - either live or from stored clips like WAV or MP3. One path leads through cleanup, where background hum fades into silence. What follows is spotting spoken parts, making sure only voices stay in frame. Encoding adjusts the audio so Gemini can work with it smoothly. Alongside, written words get lined up ready to match what was said. This sequence of preprocessing steps standardises the acoustic input and reduces variability introduced by recording conditions, thereby improving the reliability of subsequent pronunciation analysis.
- Prompt Construction and LLM-Based Pronunciation Analysis- Gemini 1.5 Flash serves as the core reasoning component of the proposed system. As large language models do not explicitly compute phonetic likelihoods in the manner of traditional ASR-based approaches, the system employs carefully designed prompt engineering to obtain structured and precise pronunciation assessments. Each prompt defines the analysis task, embeds the serialised audio input alongside its reference text, and enforces explicit output constraints to ensure consistency. In response, Gemini produces information including identified phonemes, word-level pronunciation accuracy, phoneme substitutions, insertions, and deletions, as well as segmental and suprasegmental observations, accompanied by a textual diagnosis of the learner’s deviations. This LLM-driven analysis functions as a high-level and interpretable evaluation layer, effectively replacing conventional acoustic alignment mechanisms.
- Error Extraction and Classification- From structured data created by Gemini, a set of rules pulls out specific details about how words are spoken. When mistakes show up, they get sorted into types - like swapping one sound for another, such as saying /t/ instead of /θ/. Some errors involve missing sounds or entire syllables. Others come from adding extra bits that do not belong. Changes in rhythm or emphasis also count. Mispronouncing full words fits here too. In addition to categorical labelling, the system computes phonetic distance scores based on feature-level similarities between expected and produced phonemes, allowing the severity of each deviation to be quantified. This approach provides a more fine-grained assessment than binary correct–incorrect classification.
- Scoring and Evaluation Framework- A single score emerges when three distinct checks come together. What matters first is whether words sound right from start to finish - that forms the Word Accuracy Score. Moving deeper, individual sounds get compared one by one, counting how many matches despite missing, swapped, or extra ones - this gives the Phoneme Match Score. Rhythm and emphasis shape the next layer: where stress lands on syllables, shifts in pitch, and timing patterns feed into the Stress Score. Each piece carries a different weight when combined through a structured blend. Out comes the Final Pronunciation Score - a clear reflection of spoken clarity built from separate angles:

$$FPS = 0.5 \cdot WAS + 0.3 \cdot PMS + 0.2 \cdot SS \qquad (1)$$

where:

WAS- Word Accuracy Score.

PMS- Phoneme Match Score.

SS- Stress Score.

Feedback Generation- Leveraging the descriptive reasoning capabilities of Gemini 1.5 Flash, the system produces learner-oriented pronunciation feedback that is both clear and actionable. Wrong sounds get explained by showing how they should form. Tongue moves, voice activation, and air flow - all adjusted based on what went off track. Stress shifts and timing issues are corrected through real-time cues. Words that demonstrate the right pattern pop up when needed. Practice tips appear only where mistakes show. Each fix lines up exactly with the learner's slip. Improvement comes from precise targeting. Guidance stays linked to actual speech output.

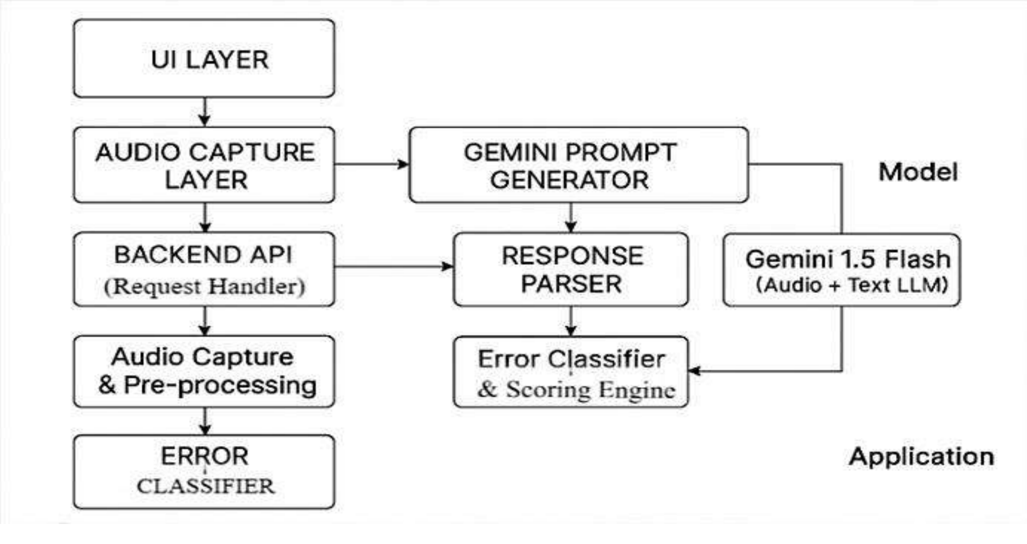

UI LAYER
AUDIO CAPTURE LAYER
GEMINI PROMPT GENERATOR
Model
BACKEND API (Request Handler)
RESPONSE PARSER
Gemini 1.5 Flash (Audio + Text LLM)
Audio Capture & Pre-processing
Error Classifier & Scoring Engine
ERROR CLASSIFIER
Application

Figure 1: System Architecture of an AI-Powered Pronunciation Mistake Detection

V. SYSTEM DESIGN

Figure 1 illustrates a modular, pipeline-oriented system architecture centered on the Gemini 1.5 Flash API. The overall design is organised into six primary functional components, each responsible for a distinct stage of the pronunciation analysis and feedback process:

- Client Layer (Front-end)- The client layer consists of a web- or mobile-based user interface that enables learner interaction with the system. This interface handles audio capture through the device microphone and performs lightweight client-side preprocessing where required. The recorded speech, along with the corresponding reference text, is then transmitted to the backend services via secure API calls for further pronunciation analysis.
- Application & Logic Layer (Backend)- The backend layer is responsible for orchestration, validation, reasoning, and result generation within the pronunciation assessment pipeline. Incoming requests are first processed by a Request Validation module, which verifies audio duration, file format, signal quality, and the presence of reference text. Inputs that are empty, excessively noisy, or malformed are rejected with appropriate error messages to ensure system robustness. Once validated, the Prompt Builder module combines the processed audio input and reference text into a structured prompt. This prompt explicitly instructs Gemini to analyse the learner's speech in comparison with the target pronunciation and to return outputs such as recognised text, phoneme or IPA representations, and pronunciation deviations. The Gemini Client then manages secure HTTP communication with the Gemini 1.5 Flash API, handling audio streaming, authentication, retry logic, and timeout control to ensure reliable inference. The Response Parser converts Gemini's textual response into a structured Python/JSON representation. This step extracts key elements, including recognised words, reference versus produced phoneme sequences, detected pronunciation errors, and any explanatory information provided by the model. These outputs are passed to the Error and Scoring Module, which aligns reference and predicted phonemes and categorises pronunciation deviations at multiple linguistic levels. Mistakes get sorted by sound - like swapping one sound for another, skipping a sound, or adding an extra one. Then there are word-level slips: wrong stress patterns or squashing syllables together too much. Sentence flow matters too; awkward pauses or uneven rhythm count here. From these details comes a score for how close words match the target - the WAS. Another number tracks correct sounds overall - that is the PMS. A third evaluates stress and intonation, giving the SS. Once scores settle, each error finds its place again in the original line. Clear notes then explain what went off track - for easier understanding. Visual cues such as colour coding (e.g., red for incorrect words and amber for partially correct pronunciations) are applied, along with textual guidance (e.g., "The phoneme /θ/ was realised as /t/ in the word *think*"). Meanwhile, stored logs capture usage patterns without personal details. These records help track how well the system works. Over time, they guide tweaks to prompts. Scoring methods also evolve using this data. Even interface choices get shaped by what shows up here. Behind the scenes, analytics turn raw activity into adjustments.
- Model Layer (Gemini 1.5 Flash)- Hidden inside sits Gemini 1.5 Flash - a smart model already trained on both sound and words. It handles spoken input along with written text, blending them into one flow. Instead of teaching it new tricks, we shape what it does by how we ask questions. Responses come out organised: breakdowns, reasons, fixes - all shaped by careful prompting. No extra lessons needed; everything hinges on wording and steering its answers. Because of that approach, things stay light, nimble, ready for different ways people speak. Surprisingly little setup leads to broad usefulness when voices differ.
- Real-Time User Feedback and Iterative Practice- Every time someone speaks, the app shows how well they pronounced things. It takes data from Gemini 1.5 Flash, shaped like JSON, then turns it into something easy to understand. Instead of just numbers, there are short summaries, helpful tips, and sample phrases. Users see where mistakes happened and get ideas on what to fix. After that, they can try again right away - no delay. Each repetition is checked fresh, so the feedback changes based on the new speech. Slowly, speaking gets clearer because errors shrink over attempts. Seeing instant reactions helps people adjust before moving on. Progress builds not from one big leap but from many small tries. What stands out is how fast everything responds - it feels alive.

A. Algorithms used

- Spectral Gating Algorithm- The Spectral Gating algorithm is employed to reduce background noise and enhance the intelligibility of user speech prior to pronunciation analysis. The algorithm operates by converting the time-domain audio signal into the frequency domain using the Short-Time Fourier Transform (STFT). During segments identified as non-speech, the system estimates a noise profile that represents background spectral characteristics. This estimated noise spectrum is then used to attenuate frequency components dominated by noise while preserving speech-relevant frequencies. When weak background sounds get removed through careful filtering in both time and frequency zones, speech details stand out better. This cleanup helps machines judge how words are pronounced with fewer distractions from surrounding noise. Clearer audio means steadier results when checking spoken accuracy.
- Voice Activity Detection- Speech sections get spotted by listening for active parts. Moments of quiet, random sounds, or static are left out on purpose. Old methods check things like volume over small windows. They also watch how often sound flips between positive and negative. Another clue comes from how messy the frequencies look - neat patterns suggest talking. What matters most is telling apart real talk from everything else. More advanced approaches extend these methods by incorporating multi-band energy analysis with adaptive thresholding, allowing the system to adjust to varying noise conditions.

Speech zones get singled out, so extra noise stays out - that sharpens how clearly the system checks each spoken sound later on.

- Levenshtein Edit Distance Algorithm- One way to match spoken sounds with correct ones uses something called the Levenshtein Edit Distance. It figures out how many changes are needed - like swapping, removing, or adding sounds - to turn what someone said into the right version. Instead of just giving a total score, it lines up each sound carefully. Because of that, errors in pronunciation show up clearly, sound by sound. What makes this useful is how it tracks exactly where things differ.
- Needleman–Wunsch Alignment Algorithm- A match stretches from start to finish when timing wobbles, pacing slips, or stress lands differently. Pairing sounds one by one, it lines up what was said with what should be - keeping the original flow intact. When rhythms drift through a long phrase, mistakes like missing pieces, extra bits, or swapped parts show up more clearly. Each sound finds its best fit across the sequence, even if speed changes mid-way.

VI. SYSTEM IMPLEMENTATION AND SNAPSHOTS

This section outlines the implementation details of the proposed AI-powered pronunciation mistake detection system and presents representative snapshots of the deployed application. The complete end-to-end workflow—from speech input acquisition and preprocessing to LLM-based pronunciation analysis and multi-level error classification—is described to illustrate the practical realisation and operational flow of the proposed approach. The objective of this section is to demonstrate how the conceptual methodology described earlier is translated into a fully functional system capable of real-time pronunciation analysis.

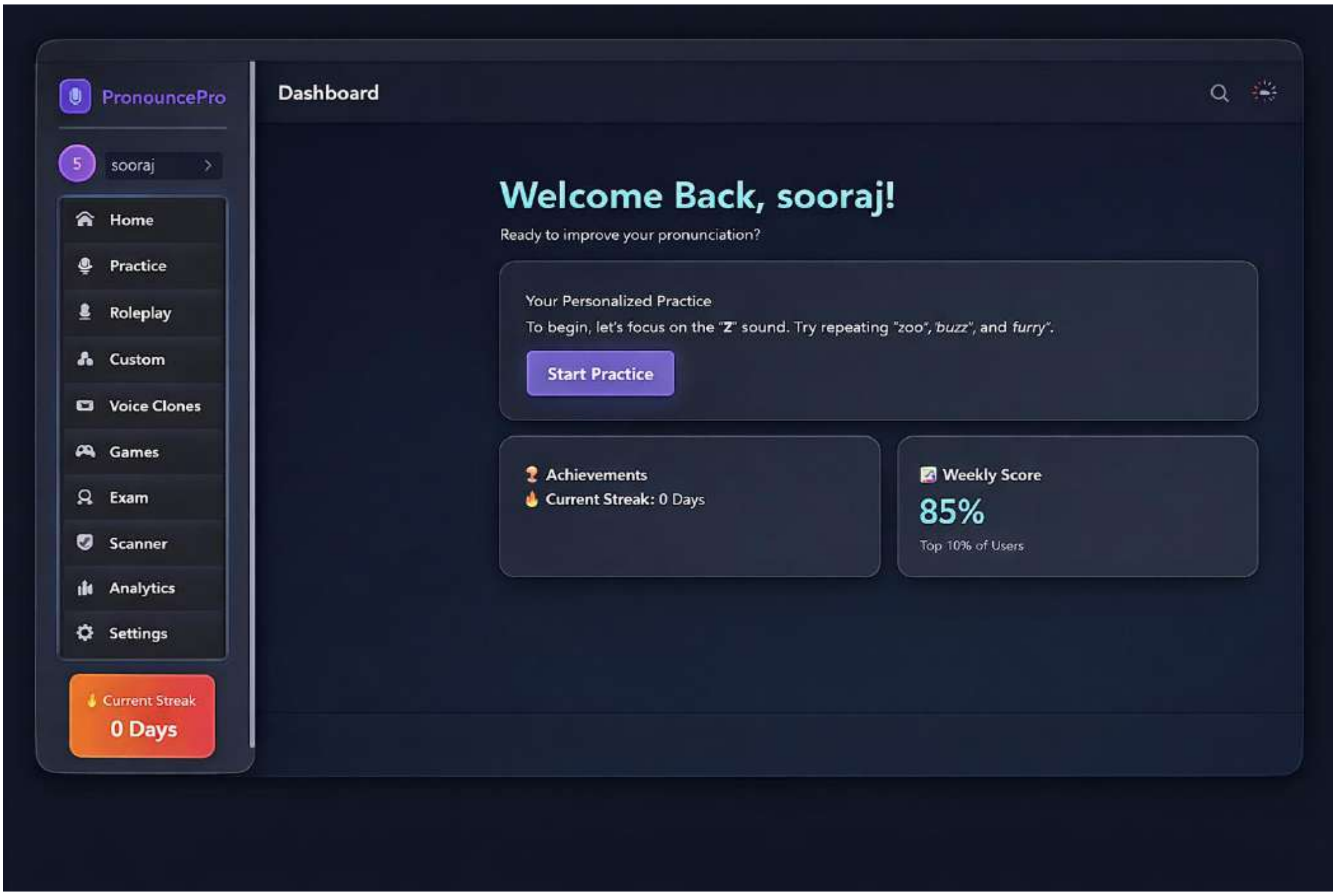
The system features a fully interactive learner dashboard designed to support personalised pronunciation training. Look at Figure 2 - after logging in, people land here first. This screen shows custom practice tips, how far you have come, your score each week, and things like daily streaks or badges earned. From this spot, moving into lessons feels natural, checking past results is quick, and spotting patterns in outcomes gets easier. Simple layout, clear paths through tasks, helps keep focus without confusion. Jumping from speaking drills to simulated talks to tests happens without hiccups because everything lines up just right.

Figure 3 presents the interface of the Practice module, which allows users to select the target language, difficulty level, and reference voice before recording or uploading their speech samples. After the input is processed through the backend services and the Gemini 1.5 Flash analysis pipeline, the system returns a comprehensive pronunciation evaluation. What you see gives one number for speaking clarity, then splits it into how well rhythm and pitch are handled. A picture shows everything laid out simply, so learners grasp where their voice speeds up, slows down, holds steady, shifts tone, or stresses certain parts of words. Another layer appears when diving deeper - mistakes in small sound units show clearly, paired with advice to fix them. How this looks on screen makes clear: the method spots slips in speech, then shares those findings plainly enough for solo practice to improve.

Not just about checking speech sounds, the tool also gives users focused exercises to shape their accent and improve clarity. Shown in Figure 4, the Voice Clone / Accent Training screen lets people pick a mode and choose which accent they want to work on. Here, someone has picked English training with General American as the goal. Once set, lessons are built by artificial intelligence, zeroing in on unique sound patterns like rolled or dropped R's, how "bath" gets pronounced, soft T sounds, plus similar speaking habits. A short note explains each point - how it stands apart from other ways of talking, what changes matter most for sounding closer to the chosen version. A fresh layer slips into place here - shifting focus from spotting wrong sounds toward shaping regional speech habits. What shows up in the image is teaching that sticks close to language rules, yet unfolds in steps a person can follow without hassle.

Picture this: a space where practice feels less like work, more like play. Shown in Figure 5, the Games area pulls learners in with smart pronunciation tasks that respond in real time. Scoring points comes naturally when speaking matches target patterns closely. You see your total score climb, notice how long your win streak lasts, check each session's result, and watch progress unfold. Progress isn't hidden - it shows up clearly, right there on screen. What if you could pick who guides you? That option exists - choose a voice, set the language, shape feedback style. Some want British English, others Australian, maybe even non-native accents for relatability. One size never fits all, especially with speech. Matching tools to individual needs makes sticking with them easier. Learning stays active because choices feel personal. Engagement grows without forcing it. Simple setup, deeper connection. The system also includes an Exam Mode, designed to provide formal pronunciation assessment through structured test items.

Figure 6 presents the exam completion interface, where A screen shows results once the test ends - here, a person gets one number that sums up how well they did on speaking tasks. This particular try lands at 68 per cent, pulled together by checking several spoken answers through automated review. The interface displays both the reference sentence and the learner's spoken output, enabling direct comparison between the expected and produced pronunciations. For each prompt, the system computes an individual item score (e.g., 80% for the highlighted sentence) and provides additional diagnostic feedback in the lower panel, including observations related to articulation accuracy, fluency, and rhythmic control. This exam mode emulates a standardised evaluation setting and facilitates longitudinal progress tracking. Moreover, it serves a dual purpose within the system: offering learners a structured mechanism for measuring improvement while simultaneously generating controlled speech samples that support internal validation of the model's pronunciation scoring consistency. This functionality demonstrates the system's capability to conduct complete pronunciation assessments with transparent and interpretable scoring, extending beyond isolated practice-based evaluation.



Welcome Back, sooraj!

Ready to improve your pronunciation?

Your Personalized Practice

To begin, let's focus on the "Z" sound. Try repeating "zoo", buzz", and furry".

Achievements

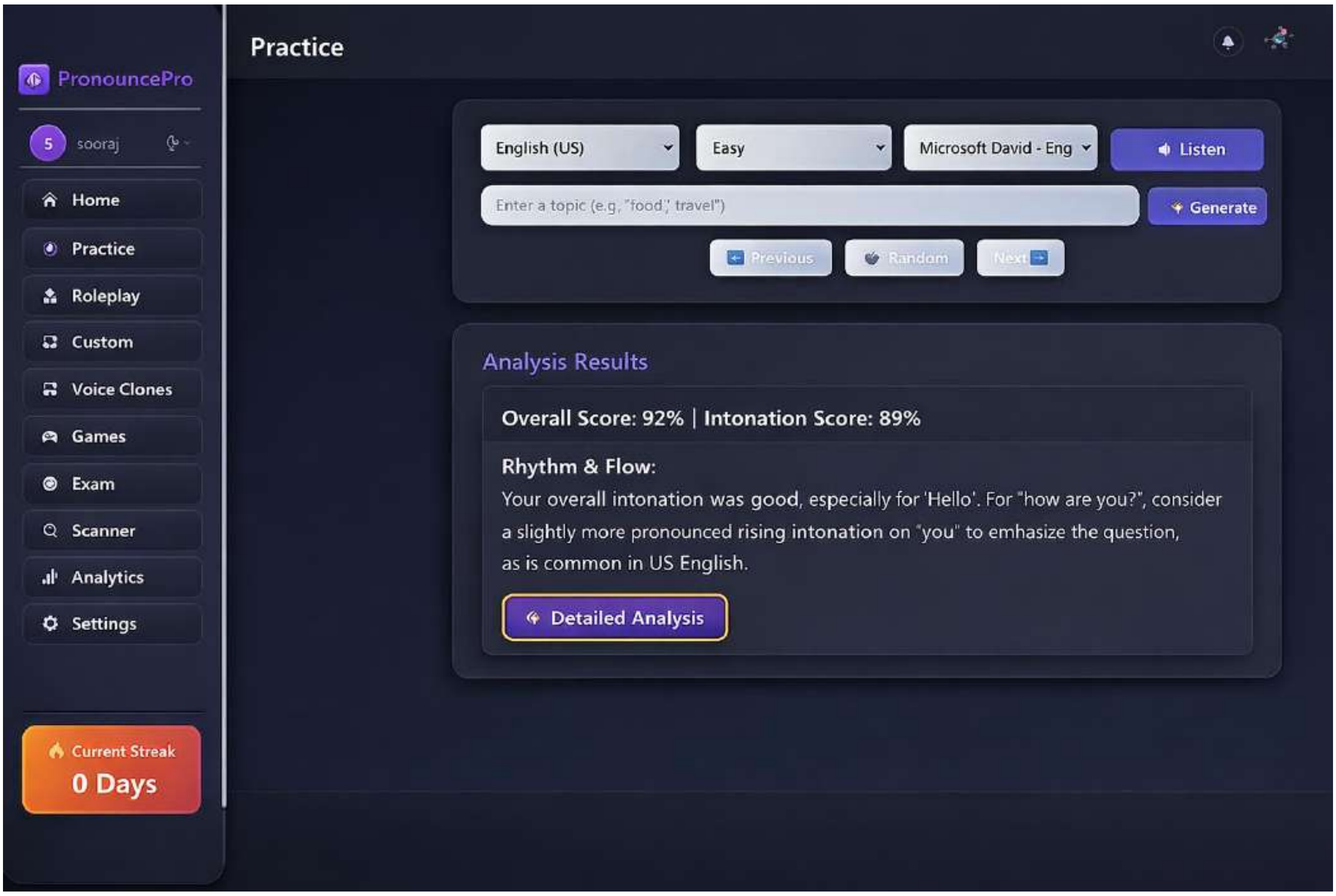
Current Streak: 0 Days

Weekly Score

85%

Top 10% of Users

Figure 2: Dashboard



English (US)

Easy

Microsoft David - Eng

Enter a topic (e.g, "food," travel")

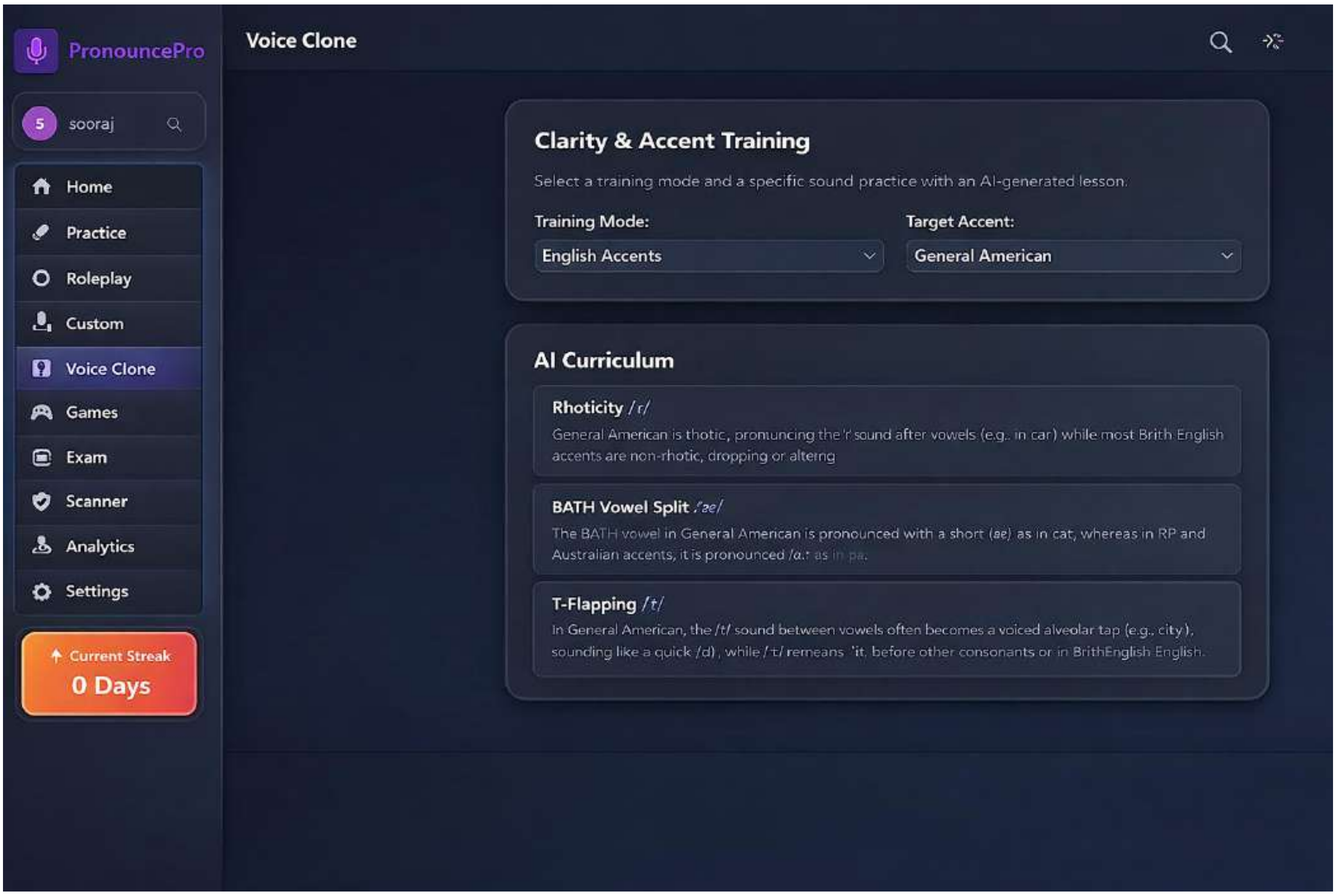
Analysis Results

Overall Score: 92% | Intonation Score: 89%

Rhythm & Flow:

Your overall intonation was good, especially for 'Hello'. For "how are you?", consider a slightly more pronounced rising intonation on "you" to emhasize the question, as is common in US English.

Figure 3: Practice Mode



Voice Clone

Clarity & Accent Training

Select a training mode and a specific sound practice with an AI-generated lesson.

Training Mode: English Accents

Target Accent: General American

AI Curriculum

Rhoticity /r/

General American is thotic, promuncing the 'r sound after vowels (e.g. in car) while most Brith English accents are non-rhotic, dropping or alterng

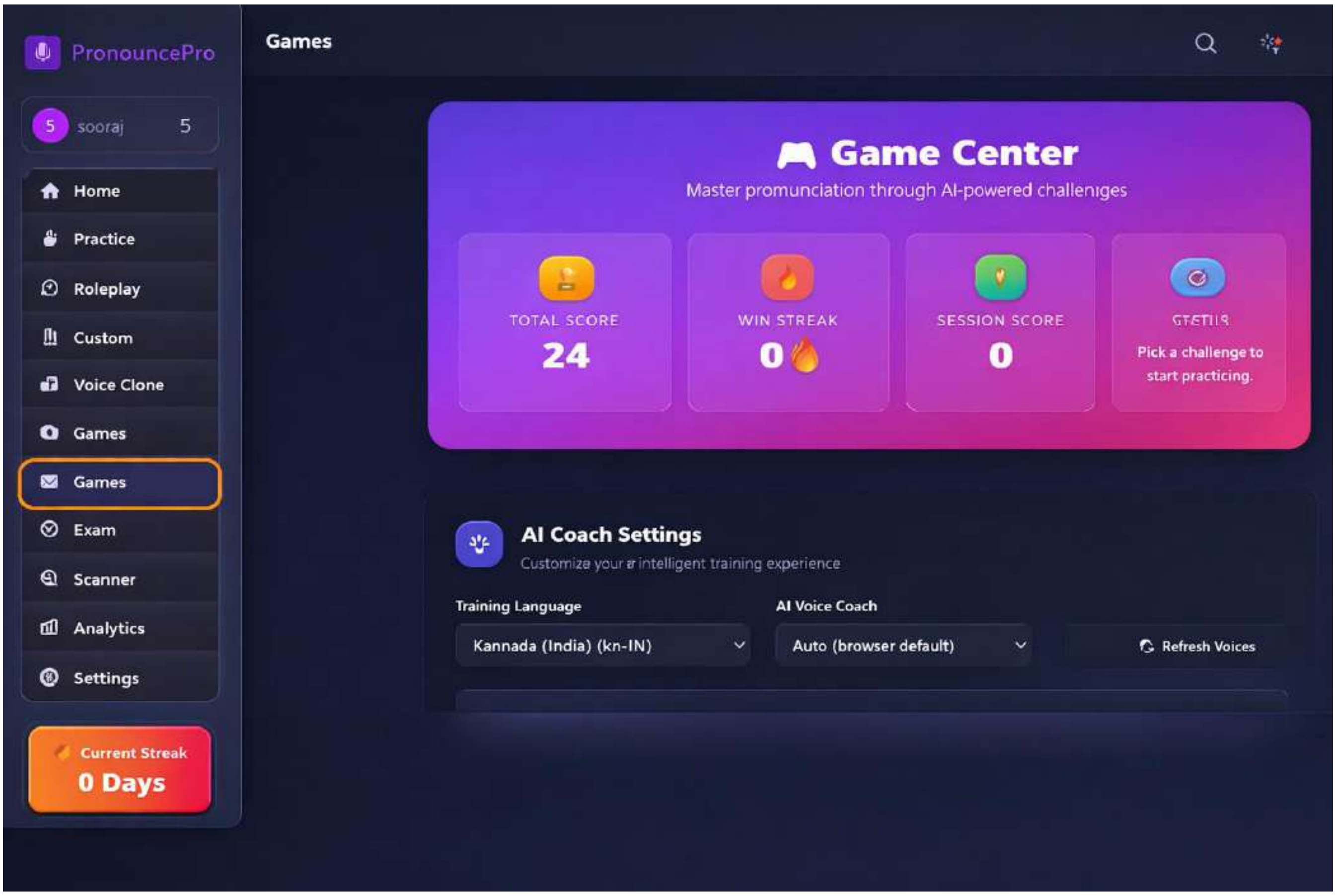
BATH Vowel Split /æ/

The BATH vowel in General American is pronounced with a short (æ) as in cat, whereas in RP and Australian accents, it is pronounced /ɑ.ː as in pa.

T-Flapping /t/

In General American, the /t/ sound between vowels often becomes a voiced alveolar tap (e.g. city), sounding like a quick /d), while /t/ remeans 'it, before other consonants or in BrithEnglish English.

Figure 4: Clarity & Accent Training



Games

Game Center

Master promunciation through AI-powered challeniges

TOTAL SCORE 24

WIN STREAK 0🔥

SESSION SCORE 0

GTETIIS

Pick a challenge to start practicing.

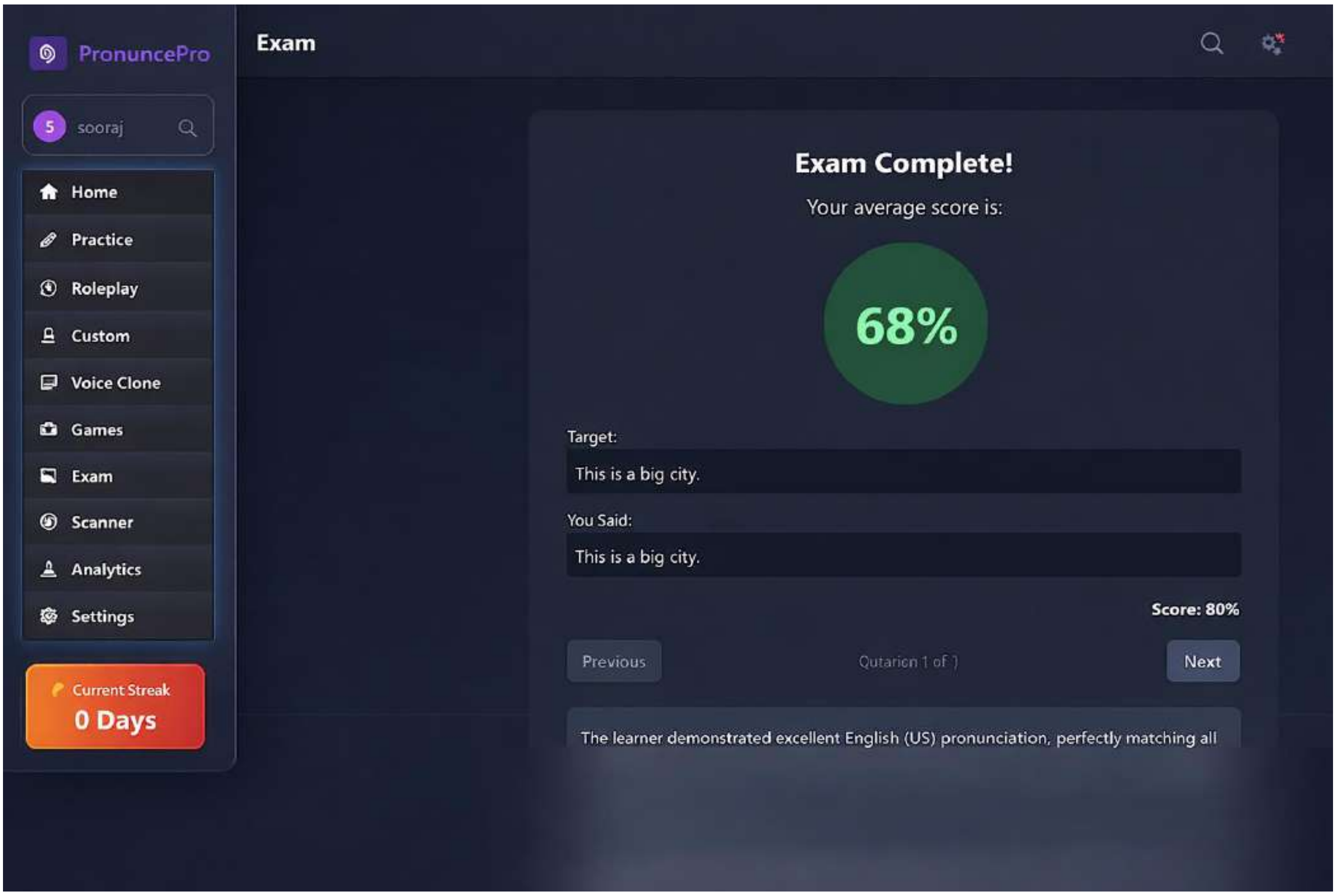
AI Coach Settings

Customize your ᵻ intelligent training experience

Training Language: Kannada (India) (kn-IN)

AI Voice Coach: Auto (browser default)

Figure 5: Game Centre



Exam Complete!

Your average score is:

68%

Target:

This is a big city.

You Said:

This is a big city.

Score: 80%

Qutarion 1 of)

The learner demonstrated excellent English (US) pronunciation, perfectly matching all

Figure 6: Exam Mode

VII. CONCLUSION

This work presents an AI-powered pronunciation mistake detection system built using the Gemini 1.5 Flash multimodal model, combined with a structured prompt engineering pipeline and a rule-based scoring engine. Unlike conventional pronunciation scoring systems that require extensive model training, acoustic modelling, or ASR fine-tuning, the proposed approach requires no training, relying instead on a pretrained LLM for both transcription and phoneme-level reasoning. The system demonstrates Fine results show up in test scores - an 81.66% overall mark sits alongside solid detection at 88.88%. Word errors stay low, just like those on smaller sound units. Confusion charts reveal where slips happen most, which lines up well with how people actually stumble when speaking. Errors of timing and pitch? Caught too, not just wrong sounds. The use of a large language model allows high adaptability, minimal setup, and granular feedback generation without supervised datasets. The built-in scoring engine further enables consistent learner evaluation with transparent, interpretable metrics.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] M. Algabri, H. Mathkour, M. Alsulaiman, and M. A. Bencherif, “Mispronunciation detection and diagnosis with articulatory-level feedback generation for non-native Arabic speech,” *Mathematics*, vol. 10, no. 15, Art. no. 2727, 2022. Available from: https://doi.org/10.3390/math10152727

[2] L. Zhang, Z. Zhao, C. Ma, L. Shan, H. Sun, L. Jiang, S. Deng, and C. Gao, “End-to-end automatic pronunciation error detection based on improved hybrid CTC/attention architecture,” *Sensors*, vol. 20, no. 7, Art. no. 1809, 2020. Available from: https://doi.org/10.3390/s20071809

[3] M. Shahin, J. Epps, and B. Ahmed, “Phonological-level wav2vec2-based mispronunciation detection and diagnosis method,” *arXiv preprint*, arXiv:2311.07037v1, 2023. Available from: https://doi.org/10.48550/arXiv.2311.07037

[4] Y. El Kheir, S. A. Chowdury, and A. Ali, “Multi-view multi-task representation learning for mispronunciation detection,” *arXiv preprint*, arXiv:2306.01845, 2023. Available from: https://doi.org/10.48550/arXiv.2306.01845

[5] W. Dong, C. Cucchiarini, R. van Hout, and H. Strik, “Automatic pronunciation assessment for L2 English by incorporating suprasegmental features and weighted loss function,” in *Proc. 10th Workshop on Speech and Language Technology in Education (SLaTE)*, Netherlands, 2025. Available from: https://doi.org/10.21437/SLaTE.2025-5

[6] C. H. Jo, T. Kawahara, S. Doshita, and M. Dantsuji, “Automatic pronunciation error detection and guidance for foreign language learning,” in *Proc. Int. Conf. Spoken Language Processing (ICSLP)*, 1998, pp. 2639–2642. Available from: https://tinyurl.com/335famka

[7] H. Strik, K. Truong, F. de Wet, and C. Cucchiarini, “Comparing different approaches for automatic pronunciation error detection,” *Speech Communication*, vol. 51, no. 10, pp. 845–852, 2009. Available from:

https://doi.org/10.1016/j.specom.2009.05.007

[8] S. Robertson, C. Munteanu, and G. Penn, "Pronunciation error detection for new language learners," in *Proc. INTERSPEECH*, 2016, pp. 2691–2695. Available from: https://www.researchgate.net/publication/307858007

[9] H. Strik, K. P. Truong, F. D. Wet, and C. Cucchiarini, "Comparing classifiers for pronunciation error detection," 2007. Available from: https://www.researchgate.net/publication/221492116

[10] C. Zhu, A. Wumaier, D. Wei, Z. Fan, J. Yang, H. Yu, and L. Wang, "Pronunciation error detection model based on feature fusion," *Speech Communication*, vol. 156, Art. no. 103009, 2024. Available from: https://www.researchgate.net/publication/375656998

[11] S. Xu, J. Jiang, Z. Chen, and B. Xu, "Automatic pronunciation error detection based on linguistic knowledge and pronunciation space," in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*, 2009, pp. 4841–4844. Available from: https://www.researchgate.net/publication/220731855

[12] Y. Dai, "[Retracted] An automatic pronunciation error detection and correction mechanism in English teaching based on an improved random forest model," *Journal of Electrical and Computer Engineering*, vol. 2022, Art. no. 6011993, 2022.

[13] N. Hosseini-Kivanani, R. Gretter, M. Matassoni, and G. D. Falavigna, "Experiments of ASR-based mispronunciation detection for children and adult English learners," *arXiv preprint*, arXiv:2104.05980, 2021. Available from: https://doi.org/10.48550/arXiv.2104.05980

[14] Z. Zhang, Y. Wang, and J. Yang, "End-to-end mispronunciation detection with simulated error distance," in *Proc. INTERSPEECH*, Incheon, South Korea, 2022. Available from: https://www.isca-archive.org/interspeech_2022/zhang22p_interspeech.pdf

[15] H. Ryu, S. Kim, and M. Chung, "A joint model for pronunciation assessment and mispronunciation detection and diagnosis with multi-task learning," in *Proc. INTERSPEECH*, Dublin, Ireland, 2023. Available from: https://www.researchgate.net/publication/373248779

ABOUT THE AUTHORS

Supritha P. O. received her B.E. in 2017 and M.Tech in 2020 from VTU, Belagavi. She is an Assistant Professor in the Department of Computer Science and Engineering at SDM Institute of Technology, with research interests in artificial intelligence, cloud computing, and computer networks.

Omkar Mahale received his B.E. degree in Computer Science and Engineering from `Visvesvaraya Technological University (VTU), Belagavi. His areas of interest include deep learning, machine learning, and artificial intelligence. He was a student of the Department of Computer Science and Engineering at SDM Institute of Technology, Ujire.

Shalya Gaonkar received his B.E. degree in Computer Science and Engineering from Visvesvaraya Technological University (VTU), Belagavi. His areas of interest include deep learning, machine learning, and artificial intelligence. He was a student of the Department of Computer Science and Engineering at SDM Institute of Technology, Ujire.

Shetty Aditya Udaya received his B.E. degree in Computer Science and Engineering from Visvesvaraya Technological University (VTU), Belagavi. His areas of interest include deep learning, machine learning, and artificial intelligence. He was a student of the Department of Computer Science and Engineering at SDM Institute of Technology, Ujire.

Sooraj Devadiga received his B.E. degree in Computer Science and Engineering from Visvesvaraya Technological University (VTU), Belagavi. His areas of interest include deep learning, machine learning, and artificial intelligence. He was a student of the Department of Computer Science and Engineering at SDM Institute of Technology, Ujire.