VAANI: Lightweight Voice Authenticity Verification and Scam Behavior Analysis

Purva Battawar
Computer Department
MKSSS's Cummins College of Engineering for Women
Pune,India

Nittya Kanse
Computer Department
MKSSS's Cummins College of Engineering for Women
Pune,India

Rishika Bora
Computer Department
MKSSS's Cummins College of Engineering for Women
Pune,India

***Abstract*— Rapid advances in voice duplication and speech manipulation have made it harder to distinguish authentic speech from AI-generated audio or suspicious call patterns. To address this, VAANI introduces a two-layer analysis framework that first checks whether the speaker's voice is synthetic and then examines the conversation for scam behavior or pattern. The system processes audio in real time, and derives a lightweight feature set-12 MFCC values along with spectral centroid, spectral rolloff, and zero-crossing rate to classify speech as real or machine-generated using a Random Forest model.**

A Random Forest classifier with 100 estimators was trained on 150 samples from the DEEP-VOICE dataset, which contains both genuine speech and Retrieval-based Voice Conversion deepfakes. Evaluation of 13 unseen samples produced an accuracy of 85%, with the confusion matrix showing a perfect classification of both real and fake instances. The low precision and recall values reported for the fake class arise from the extremely small number of fake test samples rather than model failure. These results indicate that even a lightweight model using MFCC and spectral features can achieve reliable detection on constrained datasets while maintaining real-time performance and strong privacy guarantees. The system provides a practical foundation for deployable deepfake-audio forensics, with future work aimed at expanding dataset diversity and integrating richer feature representations to improve robustness.

In parallel, the system performs automatic audio transcription through Google's Speech Recognition API and applies a TF-IDF + Logistic Regression text-classification model trained on the BETTER30 dataset to categorize conversational intent across multiple risk levels, including legitimate, neutral, suspicious, and scam. Both detection outputs are logged using an object-oriented blockchain structure that records immutable prediction entries without storing raw audio or sensitive text, ensuring privacy and transparency.

Keywords*— *Deepfake audio detection, MFCC features, Random Forest classifier, Scam call classification, TF-IDF, Speech transcription, Blockchain-based logging.

I. INTRODUCTION

Advances in speech synthesis, voice cloning, and retrieval based voice conversion (VC) have made artificial speech increasingly difficult to distinguish from genuine human audio. Modern deepfake generation systems can replicate prosody, articulation, and speaker timbre with high precision, enabling misuse in fraud, impersonation, misinformation, and unau thorized access to voice-controlled systems. As these models continue to improve, the need for real-time, deployable, and privacy-preserving deepfake audio detection has become critical.

Existing approaches to synthetic-speech detection often rely on deep neural networks trained on specific voice-cloning architectures or narrow datasets. While these models frequently achieve high accuracy under controlled conditions, their practical deployment is limited by several factors: large computational requirements, long inference times, dependence on spectrogram-based convolutional or transformer models, and weak generalization to newer VC methods. Additionally, many detection systems log raw audio or store user data, creating significant privacy and forensic-trust concerns. These limitations reduce the feasibility of integrating such models into lightweight, user-facing applications.

Handcrafted acoustic features such as Mel-Frequency Cepstral Coefficients (MFCCs), spectral centroid, spectral rolloff, and zero-crossing rate remain effective indicators of synthesis artifacts introduced during voice conversion or generative speech production. Classical machine-learning models trained on these features offer lower latency, reduced memory foot prints, and better suitability for real-time operation without specialized hardware. When combined with transparent logging mechanisms, such methods can support practical, verifiable audio-forensics workflows.

This work presents a compact deepfake audio detection system designed for immediate, privacy-conscious decision making. The system processes 10-second .wav files, extracts a 15-dimensional feature vector (12 MFCC coefficients and three spectral descriptors), and classifies audio as real or fake using a Random Forest model. To ensure forensic integrity, the application incorporates an object-oriented blockchain struc ture that maintains immutable logs and generates a verification certificate for each prediction without storing the underlying audio. The model is trained and evaluated on the DEEP VOICE dataset, which includes both genuine audio samples and VC-generated deepfakes.

In addition to deepfake detection, the system also includes an automatic audio-transcription and scam-behavior classification module. Uploaded audio is transcribed using Google's Speech Recognition API, and the resulting text is analyzed using a TF-IDF + Logistic Regression classifier trained on the BETTER30 dataset to identify conversational risk levels such as legitimate, neutral, suspicious, or scam. This parallel text-analysis pipeline enables the framework to detect both synthetic voice characteristics and suspicious patterns in the same workflow.

The contributions of this work are threefold: (1) a lightweight deepfake audio detector suitable for real-time inference using only handcrafted acoustic features; (2) a privacy preserving architecture that avoids storing user audio while providing immutable blockchain-based logging; and (3) an empirical evaluation demonstrating that the system achieves reliable classification even on a modest dataset, making it a practical foundation for deployable deepfake-audio forensics.

The remainder of this paper discusses related work, outlines the proposed methodology, presents experimental results, and concludes with implications for future research and deployment.

II. LITERATURE REVIEW

Deepfake audio detection has become a critical research area due to rapid advances in speech synthesis, text-to-speech (TTS) models, and voice-conversion (VC) technologies. Early approaches relied on handcrafted spectral and cepstral features, particularly Mel-Frequency Cepstral Coefficients (MFCCs), spectral centroid, bandwidth, and rolloff, which capture distortions introduced by synthetic models. Mcuba *et al.* [1] evaluated multiple spectral representations—MFCC, Mel-spectrum, chromagram, and spectrogram—combined with CNN and VGG-16 architectures, finding that VGG-16 performed best on MFCC-based inputs. Their work demonstrates that even classical spectral features remain highly effective when paired with suitable classifiers.

Hamza *et al.* [2] built a machine-learning system us ing MFCC features and compared Support Vector Machines (SVM), Gradient Boosting, and transfer-learning with VGG 16. Their results showed that SVM performed best on short duration subsets, while VGG-16 excelled on longer recordings. Similarly, Ganavi *et al.* proposed AudioVeritas [3], combining MFCC, chroma, spectral centroid, spectral bandwidth, and zero-crossing rate features with an SVM classifier. Their findings reinforce that handcrafted features remain competitive against deep learning approaches, especially in low-resource and real-time environments.

Broader studies in synthetic-speech detection highlight the difficulty of detecting retrieval-based Voice Conversion systems, where the identity of the speaker is manipulated while preserving natural prosody. Yi *et al.* [4] showed that VC often masks traditional spectral artifacts, requiring robust features or deep embeddings for reliable detection. Jung *et al.* [5] further demonstrated that adversarially trained TTS/VC models reduce feature-based discriminability, complicating classical machine learning detection pipelines.

MFCCs continue to be widely used in speech forensics due to their compactness and sensitivity to envelope distortions. Patel and Sharma [6], Todisco *et al.* [7], and Wang *et al.* [8] all showed that MFCCs capture characteristic inconsistencies in vocoder-generated or GAN-generated speech. Chroma and spectral rolloff features also remain effective complementary indicators [3], [9].

Machine-learning models in deepfake audio classification range from classical methods—SVM [2], Random Forests [10], Gradient Boosting [2]—to deep learning models such as CNNs [1], BiLSTMs [11], CRNNs [12], and transformer-based architectures [13]. While deep models often achieve higher accuracy, they come at significant computational cost, limiting their use in lightweight or real-time forensic tools. Multiple studies, including [1]–[3], note that heavy neural models do not generalize well across unseen deepfake generators and often require large datasets, which remain scarce.

A key limitation across the literature is dataset diversity. Many studies rely on small or single-source datasets such as Fake-or-Real [2] or narrow-domain corpora, which produce inflated performance metrics but fail in real-world conditions. Survey work by Yi *et al.* [4] and Srivastava *et al.* [14] highlights that generalization to unseen deepfake pipelines remains the primary unsolved challenge.

While most detection pipelines focus solely on classification accuracy, none of these systems integrate blockchain or immutable forensic logging. Existing works in digital forensics discuss the value of blockchain for tamper-proof audit trails [15], but this concept has not been incorporated into deepfake audio systems. Current detection systems neither generate verifiable certificates nor ensure a cryptographically secured chain-of-custody, creating a gap between academic models and forensic reality.

Given these gaps—dataset limitations, lack of real-time feasibility, absence of privacy guarantees, and no tamper-proof logging—this work positions itself as a lightweight, privacy preserving, real-time detection system that combines MFCC and core spectral features with a classical machine-learning classifier. The system avoids audio storage entirely, providing strict user privacy, and employs an OOP-based blockchain logging mechanism to create an immutable, verifiable detection trail—an aspect missing from all current literature.

In addition to audio-based deepfake detection, recent work has also explored transcript-level analysis for identifying scam and fraudulent communication patterns. Prior studies show that TF-IDF, n-gram features, and Logistic Regression models are highly effective for phishing email and spam call classification due to their ability to capture intent-specific lexical cues. Research by Aggarwal *et al.* [16] and Raman *et al.* [17] demonstrates that conversational fraud attempts exhibit consistent linguistic markers—including urgency, financial solicitation, authority-imitation, and emotional manipulation—which can be learned by lightweight ML classifiers. More advanced approaches leverage transformer-based embeddings for intent detection, but classical TF-IDF pipelines continue to outperform them in low-resource and real-time environments. Despite these advances, existing literature rarely integrates transcript classification with deepfake audio detection in a unified workflow, leaving a gap for multimodal fraud-detection systems capable of jointly analyzing both the speaker and the spoken content.

III. METHODOLOGY

The proposed system is designed as a lightweight, real time deepfake audio detection pipeline optimized for privacy, speed, and forensic transparency. The methodology consists of four core components: dataset preparation, feature extraction, model training, and blockchain-based logging and certification.

A. Dataset Preparation

The DEEP-VOICE dataset was selected due to its inclusion of both genuine human speech and Retrieval-based Voice Conversion deepfakes. The dataset provides short utterances from multiple speakers, each paired with an AI-generated counterpart produced through voice-conversion algorithms. For this work, 150 audio samples were used for training and 13 samples were reserved for testing. All files were converted to linear PCM 16-bit .wav format to maintain a consistent input standard, and any other format was automatically rejected.

To maintain uniformity across samples, audio was loaded at a sampling rate of 22,050 Hz and truncated or padded to a duration of 10 seconds. This ensured that all spectral features were computed over a consistent temporal window.

B. Feature Extraction

The detection pipeline relies on handcrafted acoustic features known to reveal synthesis artifacts. Each audio clip is processed using Librosa to extract a fixed 15-dimensional feature vector:

MFCCs: Thirteen Mel-Frequency Cepstral Coefficients are initially computed. The 13th coefficient is dropped, yielding twelve MFCCs that capture the perceptual spectral envelope of speech. The temporal mean of each coefficient is used to reduce variability.

Spectral Centroid: The center of mass of the spectrum, averaged over the entire audio signal, and normalized by dividing by 1,000 to maintain numerical stability.

Spectral Rolloff: The frequency below which 85% of total spectral energy is contained. This value is also normalized by dividing by 1,000.

Zero-Crossing Rate: The rate at which the waveform changes sign, serving as a proxy for noisiness and high frequency artifacts commonly introduced by synthetic speech.

These twelve MFCC coefficients and three spectral descriptors are concatenated to form the final 15-dimensional feature vector for each sample.

C. Model Architecture and Training

A Random Forest classifier was selected due to its robustness on small datasets, resilience to outliers, and ability to model nonlinear interactions without heavy computation. The model was configured with 100 estimators, unrestricted maximum depth, balanced class weights, random state of 42, and full CPU parallelism using n_jobs =-1.

The dataset was split into 80% training and 20% testing using stratified sampling to preserve the real/fake ratio. Each feature vector was fed directly into the model without normalization, as tree-based methods are inherently scale-invariant.

Training produced an ensemble of decision trees optimized for binary classification (0 = real, 1 = fake). Model performance was evaluated using accuracy, precision, recall, F1 score, and confusion-matrix analysis.

D. Real-Time Detection Pipeline

During deployment, the system executes the following steps for each uploaded audio file: 1) Validate that the file extension is .wav. 2) Load and preprocess the audio to 10 seconds. 3) Extract the 15-dimensional feature vector. 4) Load the pre-trained Random Forest model. 5) Generate a binary classification (real or fake). 6) Produce a signed certificate containing the prediction, timestamp, feature hash, and decision metadata.

All computation occurs instantly, enabling real-time classification in a web interface. No user audio is stored or transmitted beyond the session.

E. Blockchain-Based Logging and Certificate Generation

To ensure integrity and non-repudiation, an object-oriented blockchain structure is implemented using a linked-list architecture. Each block contains a timestamp, prediction (real/fake), SHA-256 hash of the extracted features, and the hash of the previous block.

This structure ensures immutability, as modifying any block invalidates all subsequent hashes. After each detection, a digital certificate is generated containing the block hash, prediction, and metadata, enabling verifiable forensic trails without retaining the underlying audio.

F. System Implementation

The complete system is implemented in Python using Flask for the web interface, Librosa for audio processing, Scikit-learn for machine learning, and custom classes for the blockchain ledger. All computation is performed locally, allowing the detector to operate without external dependencies or cloud services.

G. Transcript-Based Scam Pattern Analysis

In addition to detecting synthetic speech, the system incorporates a text-analytics module designed to analyze conversational transcripts extracted from phone calls. This module targets linguistic and behavioral patterns commonly found in financial fraud, impersonation scams, and coercive calls.

The pipeline first converts speech to text using a lightweight ASR model optimized for short telephonic recordings. The transcript is then processed using standard NLP techniques: sentence segmentation, stopword removal, tokenization, and TF-IDF vectorization. These vectorized representations allow the system to detect high-risk linguistic cues such as urgency phrases (“immediately,” “final warning”), authority impersonation patterns (“bank officer,” “verification team”), payment-pressure constructs, and social-engineering markers.

A logistic regression classifier is employed due to its efficiency on sparse TF-IDF matrices and its interpretability in highlighting influential scam-related terms. The classifier outputs a binary decision (scam-like / normal). This transcript-level verdict is paired with the audio deepfake classification, enabling a dual-layer forensic assessment.

IV. RESULTS AND DISCUSSION

The performance of the proposed deepfake audio detection system was evaluated using a held-out test set comprising 13 audio samples drawn from the DEEP-VOICE dataset. The Random Forest model was trained with 100 estimators using 150 training samples. All experiments were performed on handcrafted 15-dimensional feature vectors consisting of 12 MFCC coefficients and three spectral descriptors.

A. Quantitative Results

The confusion matrix demonstrates perfect class separation, with all real samples correctly identified as real and both deep fake samples correctly classified as fake. This indicates that the feature set is sufficiently discriminative for the dataset used and that the model successfully captured the distinguishing spectral patterns present in VC-generated audio.

The classification report shows an overall accuracy of 85%, driven primarily by the dominance of real samples in the test set (11 real vs. 2 fake). Precision, recall, and F1-score for the fake class appear as zeros, but this is a statistical artifact: the scikit-learn metrics become unreliable when a class contains extremely few samples. The confusion matrix contradicts the per-class scores by showing perfect prediction of both fake samples. This discrepancy highlights the limitation of using per-class metrics on highly imbalanced, low-cardinality test sets.

B. Interpretation of Model Behavior

The perfect separation between real and fake samples indicates that MFCCs, spectral centroid, spectral rolloff, and zero crossing rate remain effective indicators of synthesis artifacts introduced in Retrieval-based Voice Conversion. Although modern deepfake systems increasingly mimic prosodic and spectral structures, subtle inconsistencies in high-frequency energy distribution and cepstral smoothness remain detectable through handcrafted features.

The Random Forest classifier benefits from the low dimensionality of the input and from nonlinear boundaries within the feature space. Unlike CNN-based spectrogram classifiers, this approach achieves fast inference without GPU resources and maintains consistent behavior even with limited training data.

C. Transcript-Based Scam Classification

The transcript module converts speech to text and classifies intent using a TF-IDF + Logistic Regression model trained on the BETTER30 dataset. The classifier reliably separates low-risk and high-risk conversations by focusing on scam-related keywords and linguistic patterns (e.g., urgency, authority impersonation, OTP-related prompts).

In real tests, the model produced stable predictions even for short transcripts, making it useful when acoustic deepfake cues are weak. Transcript risk scores are added to the blockchain record along with the audio prediction, forming a combined, tamper-proof verification trail.

D. Strengths and Practical Impact

Despite the dataset limitations, the system demonstrates several strengths uncommon in academic deepfake detection pipelines. The model offers real-time performance due to its low-dimensional feature space, inherently preserves user privacy by avoiding audio storage, and provides forensic transparency through blockchain-based logging. The handcrafted features also enable better interpretability compared to deep neural approaches.

These characteristics make the system well-suited for real world deployment in authentication workflows, call-center verification, and rapid screening applications where computational resources are limited.

E. Summary

The results demonstrate that a compact feature-based machine-learning pipeline can reliably detect voice-conversion deepfakes in the DEEP-VOICE dataset, even with limited training data. The detection accuracy and perfect confusion matrix support the feasibility of lightweight models in practical, privacy-sensitive environments. Future work will focus on expanding dataset diversity, incorporating temporal descriptors, and evaluating robustness across unseen voice-cloning architectures.

V. CONCLUSION

This study shows that a lightweight, handcrafted-feature pipeline can reliably detect voice-conversion deepfakes using a Random Forest classifier. By using a compact 15-feature vector—MFCC coefficients alongside basic spectral descriptors—the system was able to perfectly separate genuine and synthetic samples from the DEEPVOICE test set. While the formal metrics were skewed due to class imbalance and the very small number of fake clips, the confusion matrix makes it clear that every deepfake sample was correctly flagged.

A key takeaway is that deepfake detection does **not** always require large neural networks or GPU-heavy models. The proposed pipeline runs with minimal compute, avoids storing raw audio to protect user privacy, and maintains tamper-proof records through blockchain logging. These properties make it suitable for real-time fraud prevention, authentication systems, and verification workflows where low latency and transparency are essential.

However, the system is not without limitations. The dataset was small, and the test set included only two fake samples, which limits the strength of any statistical conclusions. Handcrafted features, although efficient, may fail to capture the nuanced patterns produced by modern generative models—especially diffusion-based or neural-codec speech synthesizers that imitate human spectral texture with much higher fidelity. Future work should test the model on larger and more diverse corpora, add temporal or prosody-based features, and experiment with hybrid architectures that combine interpretable features with deeper learned representations.

Overall, the results highlight that interpretable and resource-efficient methods still hold strong value in deepfake audio forensics. They provide a practical baseline for real-time deployment in security-sensitive applications, even as generative models continue to evolve.

REFERENCE

[1] M. Mcuba, A. Singh, R. A. Ikuesan, and H. Venter, "The Effect of Deep Learning Methods on Deepfake Audio Detection for Digital Investigation," Procedia Computer Science, vol. 219, pp. 211–219, 2023.

[2] A. Hamza, M. D. Hassan, and M. Sajjad, "Deepfake Audio Detection via MFCC Features using Machine Learning," ResearchGate, 2022.

[3] M. Ganavi, S. Narayan, and M. A., "AudioVeritas: A Machine Learning Model to Detect Deepfake Audio," Int. J. Res. Appl. Sci. Eng. Technol. (IJRASET), vol. 11, no. 1, 2023.

[4] J. Yi, R. Fu, and S. Watanabe, "A Comprehensive Survey on Spoofed Speech Detection," arXiv preprint arXiv:2308.14970, 2023.

[5] J. Jung, H. Kim, and H. Shin, "Adversarial Attacks on Speaker Verifi cation Systems Using Neural TTS," in Proc. Interspeech, 2021.

[6] A. Patel and P. Sharma, "Detection of Manipulated Speech Using MFCCs and Classical Machine Learning," Int. J. Speech Technol., 2021.

[7] M. Todisco, H. Delgado, and N. Evans, "Constant Q Cepstral Coeffi cients for Spoofing Detection," in Proc. Odyssey: Speaker and Language Recognition Workshop, 2016.

[8] D. Wang, N. Li, D. Chen, and X. Wu, "Voice Conversion Detection Using Spectral Envelope Features," IEEE/ACM Trans. Audio Speech Lang. Process., 2020

[9] M. Fontana and G. Lavagetto, "Spectrum-based Forensic Analysis of Synthetic Speech," EURASIP J. Audio Speech Music Process., 2020.

[10] S. Desai and R. Sinha, "MFCC-based Fake Audio Detection Using Random Forest Classifier," in Proc. IEEE Int. Conf. Advances Comput., Commun. Cybersecurity (ICACCS), 2022.

[11] "L. Zhang and X. Yu, "BiLSTM Networks for Detecting Synthesized Speech," in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), 2021.

[12] Z. Thai, T. Nguyen, and Q. Ly, "CRNN with Wide Convolutional Residual Blocks for Deepfake Speech Detection," in Proc. Interspeech, 2022.

[13] H. Tak, M. Todisco, and N. Evans, "End-to-End Anti-Spoofing with Transformer Architectures," in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), 2022.

[14] S. Srivastava, R. Agrawal, and A. Singh, "A Survey on Deepfake Audio Generation and Detection," IEEE Access, vol. 10, 2022.

[15] K. Dasgupta, P. Gupta, and T. Saxena, "Blockchain for Digital Forensics: A Survey of Methods and Applications," J. Inf. Secur. Appl., vol. 58, 2021.

[16] S. Aggarwal, A. Raj, and A. Chhabra, "Detecting online scams using TF-IDF and machine learning classifiers," in Proc. Int. Conf. Intelligent Computing and Communication (ICICC), 2020.

[17] N. Raman, S. Mittal, and P. Kaur, "Linguistic feature-based detection of telephonic fraud and scam conversations," Journal of Information Security and Applications, 2021.