Journal of Information Systems Engineering and Management
2025, 10(57s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Audio Deepfake Detection with Stacking Ensemble: Performance vs. Generalization

Zeltni Kamel[12], Habbati Billel[1]
[1]*University of frères Mentouri Constantine 1, Algeria*
[2]*LSIACIO laboratory, Constantine, Algeria*

ARTICLE INFO

Received: 30 Dec 2024

Revised: 12 Feb 2025

Accepted: 26 Feb 2025

ABSTRACT

The rapid progress in generative speech technologies has yielded highly realistic audio deepfakes, posing substantial challenges to security, trust, and the credibility of media. This research proposes a stacking ensemble method for audio deepfake detection. The method utilizes XGBoost and Random Forest algorithms as base learners, and a Multilayer Perceptron (MLP) as a meta-learner. The performance of the model, assessed via ten-fold cross-validation, is noteworthy, with accuracy and Matthews Correlation Coefficient (MCC) exceeding 94% and 0.88 respectively. However, when tested on an unseen dataset, the model suffers a drastic performance drop, indicating limited generalization. To investigate the cause, we conducted two targeted experiments: (1) extensive data augmentation to regularize training and (2) intentional underfitting by reducing model capacity. The lack of improvement in test performance with both strategies rules out overfitting as the primary problem. Instead, our findings point to a deeper problem of distributional shift between training and deployment domains. However, the findings indicate the practical feasibility of established machine learning methods in scenarios with limited resources or real-time constraints, given their efficiency and comparable performance. This work underscores the need for domain-robust feature representations, cross-dataset validation, and scalable solutions for real-world audio deepfake detection.

Keywords: audio deepfake detection, ensemble learning, generalization, classical machine learning.

INTRODUCTION

The rise of generative artificial intelligence has enabled the creation of highly realistic synthetic media, including speech that mimics the voice and speaking style of real individuals. Known as audio deepfakes, these manipulated or entirely fabricated speech signals are increasingly being used in harmful scenarios, from social engineering attacks and misinformation campaigns to biometric spoofing. The growing availability of tools for voice cloning, text-to-speech (TTS), and voice conversion (VC) makes this threat both accessible and scalable.

Existing detection methods, especially those based on deep learning, are often computationally intensive, data-hungry (Yi et al. 2023), and suffer from limited interpretability, which is crucial in forensic and legal contexts where the rationale behind a detection must be clearly explained. These challenges engender a renewed emphasis on statistical techniques and traditional machine learning (ML) methodologies (Rana and Bansal 2024) that offer lightweight and interpretable alternatives, particularly suitable for real-time detection and deployment on resource-constrained systems such as mobile phones, embedded systems, and edge devices, where rapid and reliable decision-making is crucial.

In this work, we propose an audio deepfake detection framework that leverages statistical acoustic features and a stacking ensemble learning strategy. We combine XGBoost (Chen and Guestrin 2016) and Random Forest (Breiman 2001) as base learners with a Multilayer Perceptron (MLP) meta-learner. This hybrid approach leverages the complementary strengths of decision tree ensembles and neural models, aiming to provide a robust and efficient alternative to deep learning-based detectors. The feature set includes Mel-frequency cepstral coefficients (MFCCs) (Abdul and Al-Talabani 2022) and statistical summaries of several low-level descriptors: chromagram (mean), root mean square (RMS) energy, spectral centroid, spectral bandwidth, spectral rolloff, and zero crossing rate (ZCR).

Journal of Information Systems Engineering and Management
2025, 10(57s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

These features are selected for their ability to capture both spectral and temporal characteristics of speech that may be distorted during synthetic generation.

On 10-fold cross-validation, the model demonstrates excellent detection performance, achieving over 94% accuracy and a Matthews Correlation Coefficient (MCC) of 0.88, validating its effectiveness in in-domain scenarios.

However, when evaluated on an unseen test set, the model's performance drops sharply to approximately 60% accuracy and a 0.19 MCC, revealing a dramatic loss of generalization. To investigate this degradation, we conducted two targeted experiments. First, we applied massive data augmentation using a variety of perturbations (e.g., noise, gain, filtering, time masking), aiming to regularize training and reduce overfitting. Second, we intentionally underfit the model by reducing its training capacity. Surprisingly, neither approach improved test performance, thereby refuting overfitting as the primary cause of the generalization failure.

These findings point instead to a more fundamental issue: a distributional mismatch between the training and testing data. This may stem from differences in voice synthesis techniques, speaker variability, or recording conditions not captured in the training distribution. Importantly, our results also suggest that classical machine learning methods, despite their limitations, can still serve as practical, lightweight solutions in low-resource or real-time scenarios, offering fast inference and interpretability.

RELATED WORKS

In terms of classes of generation techniques, text-to-speech (TTS) and and voice conversion (VC) are two prominent classes. Current TTS systems, like Tacotron 2 (Shen et al. 2018), FastSpeech (Ren et al. 2019) and VITS (Kim, Kong, and Son 2021) synthesize speech from text while maintaining the identity of a speaker. VC methods transfer the voice of a source speaker to that of a target speaker while keeping the spoken words the same. Examples of such models include AutoVC (Qian et al. 2019) and models based on VAEs (Long et al. 2022), and GANs (Kameoka et al. 2018), (Kaneko et al. 2020), (Das et al. 2023). These applications are being used more and more for ethically correct and ethically incorrect purposes, making the task of determining the authenticity of audio more difficult than ever.

On the other hand, there have been a number of works dedicated to the detection of audio deepfakes. Recently, deep learning has seen heavy use due to the model's ability to model complex representations of audio signals. CNNs and RNNs have been applied on spectrograms or raw waveforms from the audio to learn spatial and temporal patterns indicative of synthesis artifacts. Models have been applied on benchmark datasets like ASVspoof (Wang et al. 2025) and Fake-or-Real (Reimao and Tzerpos 2019). These include RawNet2 (Tak et al. 2021), SE-ResNet (Yue et al. 2022), graph neural networks (GNNs) (Jung et al. 2021), and transformers (Modak, Das, and Naskar 2025) which have been pretrained on speech data. These models are typically computationally intensive, data greedy and tend to overfit in low-resource or cross-corpus settings.

Before the dominance of deep learning, classical machine learning techniques were widely used for spoof detection. These methods typically involve extracting handcrafted features, such as Mel-frequency cepstral coefficients (MFCCs)(Hamza et al. 2022), linear predictive coding (LPC) (Xiao et al. 2015), or spectral flux, followed by classifiers like Support Vector Machines (SVMs) (Agarwal, Singh, and D 2021), Random Forests (Hamza et al. 2022), or Gaussian Mixture Models (GMMs) (Toda, Black, and Tokuda 2007). Although often less accurate than deep models in high-data regimes, classical methods have advantages in interpretability, computational efficiency, and robustness in low-resource environments.

PROPOSED ADD FRAMEWORK

Our proposed approach for ADD involves the application of classical machine learning classifiers in conjunction with traditional feature engineering techniques. In this study, we utilise the Fake or Real (FoR) dataset (Reimao and Tzerpos 2019), a publicly accessible corpus developed for deepfake audio detection, particularly the for-2sec dataset in which all files are truncated at two seconds. This dataset includes. FoR is splited into training, validating and testing sets. The training set (77.73%) used to train models. The validation set (15.58%) used to assess model performance on unseen data during training. The generalization test set (6.68%) includes only synthetic and real voices from previously unseen sources to evaluate the ability of model to generalize.

Journal of Information Systems Engineering and Management
2025, 10(57s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Features extraction

The primary features selected comprise Mel-Frequency Cepstral Coefficients (MFCCs), commonly used audio features for audio deepfakes detection (ADD). They are compact representations of the spectral characteristics of audio, inspired by how the human ear perceives sound and emphasizing frequencies that are most relevant to human hearing. To extract MFCCs, we compute the first twenty MFCC coefficients per frame. Next, for each coefficient, we compute statistical properties (mean, standard deviation, median, peak-to-peak range, percentile 25th and 75th) to summarize its behavior over time. These statistics help capture overall patterns, variability, and potential anomalies introduced by synthetic speech. This framework also involves six other acoustic features, including the mean of chromagram, root mean square (RMS) energy, spectral centroid, spectral bandwidth, spectral rolloff, and zero crossing rate (ZCR).

Classification models: Stacking ensemble model

Stacking ensemble learning encompasses two distinct phases: the training phase for base-learners and the training phase for the meta-learner. The two base learners selected are Extreme Gradient Boosting (XGB) and Random Forests (RF); each model has been chosen to represent different algorithmic paradigms. Subsequently, a Multi-Layer Perceptron (MLP) is selected as the meta-model.

Base learner 1: Extreme gradient boosting (XGB)

Extreme gradient boosting (XGB) (Chen and Guestrin 2016), is an advanced supervised algorithm based on gradient-enhanced decision trees. The algorithm creates a “strong” learner by combining predictions of “weak” learners using additive training strategies. XGB uses a second-order Taylor expansion of the loss function with a regularisation term, effectively avoiding overfitting and speeding up convergence. It improves prediction accuracy by forming new decision trees to fit residuals of previous predictions, continuously reducing the discrepancy between predicted and true values. Due to its notable speed advantage, XGB is chosen as one of the base models for the ensemble in this study.

Base learner 2: Random Forests (RF)

Random Forests (RF) are ensemble learning methods that integrate decision trees for both classification and regression tasks (Breiman 2001). As a widely recognized machine learning technique, Random Forest (RF) algorithm is capable of managing substantial volumes of data, while the phenomenon known as the “dimensionality disaster” frequently undermines the performance of alternative models in the context of big data. Additionally, RF exhibits an error rate that is comparable to that of other methodologies across the majority of learning tasks, alongside a diminished tendency toward overfitting. As one of the most distinguished algorithms, it is selected as a base model for the ensemble framework implemented in this study.

Meta-Learner: Multi Layer Perceptron (MLP)

In our stacking ensemble architecture, the final prediction is generated by a meta-learner trained on the outputs of base classifiers. For this purpose, we selected a Multi-Layer Perceptron (MLP) as the meta-learner due to its capacity to learn complex nonlinear mappings and its demonstrated effectiveness in high-dimensional learning tasks. We employ an MLP composed of a simple feedforward neural network architecture with one hidden layer. The input layer size corresponds to the number of base models multiplied by the number of class probability outputs per model. The hidden layer consists of 32 units with ReLU activation, followed by a dropout layer (dropout rate = 0.5) to reduce overfitting. The output layer is single unit with sigmoid activation function to produce final class probabilities. This configuration allows the MLP to model non-linear relationships and interactions between the output of base learners.

Training the Stacking ensemble Model

Each base learner model, XGB and RF, is independently trained using the same set of extracted features, thereby enabling the ensemble to leverage diverse inductive biases. The training set is performed in 10-fold cross-validation, which partitions the dataset into 10 segments, ensuring balanced class representation across folds. In each iteration, one segment is used for validation while the remaining segments are employed for training the model and generating predictions for that specific segment.

Journal of Information Systems Engineering and Management
2025, 10(57s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

XGBoost is trained as a binary classifier using the binary:logistic objective function, optimized to distinguish between real and fake audio samples. Model performance is monitored using the negative log-likelihood as the evaluation metric. To prevent overfitting and reduce training time, early stopping is employed, halting training after 10 consecutive rounds without improvement in validation logloss.

The Random Forest classifier is trained using the RandomForestClassifier implementation from the scikit-learn library to serve as a robust, tree-based base learner in the stacking ensemble. The model is optimized for binary classification using the Gini impurity criterion.

The meta-learner Multilayer Perceptron (MLP) is trained to combine the outputs of the base learners (XGBoost and Random Forest). Its input consists of a vector of probability scores, one from each base model, representing the confidence that an audio sample is a deepfake. The MLP learns to map these probabilities to the final binary prediction (real or fake) through non-linear transformations. Training is performed using the Adam optimizer with a learning rate of 0.001 and binary cross-entropy loss. Early stopping was applied based on validation loss, with a patience of 10 epochs. The model was trained for up to 50 epochs, with batch size = 32.

The experiments is conducted on a personal workstation equipped with an Intel Core i7 processor 8th generation and 64 GB of RAM.

RESULTS

To evaluate the performance of the proposed stacking ensemble for audio deepfake detection, we employed a comprehensive set of evaluation metrics that reflect both classification accuracy and robustness under class imbalance. The primary metrics include accuracy, Matthews Correlation Coefficient (MCC), F1-score, precision, recall, and Average Inference Time.

Accuracy provides a general measure of the overall correctness of the model, but it alone may be misleading. Therefore, Matthews Correlation Coefficient (MCC) serves as a robust indicator of reliability and capacity of the model's generalization. MCC is used to assess the correlation between predicted and true labels, especially with imbalanced class distributions, acommon scenario in audio deepfake detection. Finally, average inference time is reported to assess the practical efficiency of the model. In applications like real-time monitoring or forensic triage, low-latency predictions are essential. By measuring inference time, we provide a realistic view of the deployability and responsiveness of the model.

Framework Performance and evaluation

The proposed framework is assessed through stratified 10-fold cross-validation on the training dataset. Furthermore, to evaluate its capacity for generalization, an additional experiment is conducted using the generalization test set to simulate a real-world deployment scenario. The experimental results, including the average outcomes of the 10-fold cross-validation and testing on an unseen dataset, are presented in Table 1.

Table1 Results of stacking ensemble model

Metric	Cross Validation	Testing
Accuracy	0.9421	0.5974
Precision	0.9227	0.6242
Recall	0.9650	0.5358
F1-Score	0.9434	0.5767
MCC	0.8851	0.1991
Inference Time	0.0469	0.0017

These results demonstrate that, across all folds, the model maintained a consistently high level of performance, as indicated by the average value in Table 1. They also exemplify the strong capability of classical machine learning approaches to effectively model the training distribution. Furthermore, the high MCC value corroborates the robustness of the predictions. Conversely, the results obtained from the test dataset indicate a significant decline in performance on entirely unseen data (accuracy approximately 59%), revealing a substantial deterioration in

Journal of Information Systems Engineering and Management
2025, 10(57s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

generalization capabilities. Additionally, these findings illustrate that classical machine learning methods can achieve rapid inference times of 0.0469 ms, rendering them a practical and competitive alternative to deep learning, particularly in low-resource or real-time deployment environments.

These outcomes warrant a comprehensive investigation into the fundamental causes of generalization failure, whether primarily due to overfitting during training or reflecting the intrinsic limitations of classical machine learning models in generalizing to out-of-distribution data.

Investigating the Limits of Generalization: Beyond Overfitting

To critically assess whether overfitting was the predominant factor contributing to the decline in performance on the unseen test set, we employed two complementary strategies: extensive data augmentation and the deliberate induction of underfitting. The augmentation strategy involved expanding the training set through various acoustic perturbations to enhance the diversity of the training data. The augmentation process increased the volume of training data approximately threefold relative to the original. We implemented a series of probabilistic audio augmentation techniques, each applied independently according to the probabilities specified in Table 2.

- Gain perturbation: Randomly adjusts the signal amplitude to simulate recording volume variation.
- Additive noise: Adds background noise to the signal to simulate real-world acoustic environments.
- Filtering: Applies random bandpass and notch filters to simulate microphone and channel characteristics.
- Time masking: Randomly masks short temporal segments in the waveform or spectrogram, inspired by SpecAugment.
- Polarity inversion: Inverts the signal waveform polarity, which is perceptually harmless but useful for model invariance.
- Clipping distortion: Simulates nonlinear distortion by artificially clipping waveform amplitudes.

Table 2 Probabilities applied in data augmentation

Augmentation technique	**Probability**
Gain perturbation	0.6
Additive noise	0.6
Filtering	0.5
Time masking	0.4
Polarity inversion	0.1
Clipping distortion	0.3

After training the stacking ensemble (XGBoost + Random Forest + MLP meta-learner) on the augmented dataset, we evaluated its performance on the same unseen test set used in earlier experiments. The results are summarized in Table 3:

Table 3 Results with Data augmentation

Metric	Cross Validation	Testing
Accuracy %	0.9642	0.6259
Precision	0.955	0.6145
Recall	0.9743	0.6756
F1-Score	0.9645	0.6436
MCC	0.9285	0.1991

The results show negligible improvements over the baseline results obtained without augmentation. However, the performance on the unseen test set remained poor, with an approximately accuracy of 63% and an MCC of 0.2 indicating that regularization via augmentation was insufficient to address the generalization failure.

Journal of Information Systems Engineering and Management
2025, 10(57s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

In contrast, to further investigate the generalization failure observed on the unseen test dataset, we conducted a second set of experiments aimed at intentionally inducing underfitting to limit model's training capability. The aim was to reduce the training–test performance gap by preventing the model from memorizing training-specific patterns. To enforce underfitting, we simplified the model by limiting the number of training epochs and applied early stopping. These changes resulted in a significant drop in performance on the training folds, confirming that the model was underfitting the training data The results of this study are summarized in table 4.

Table 4 Results with underfitting

Metric	Cross Validation	Testing
Accuracy %	0.7538	0.5933
Precision	0.7391	0.5883
Recall	0.7845	0.6213
F1-Score	0.7611	0.6044
MCC	0.5086	0.1869

The results of the second study show a significant drop in performance on the training folds with accuracy 75% and MCC 0.92, confirming that the model was underfitting the training data. On the other hand, the model's performance on the test set remained nearly identical to that of the original overfit-prone model and the version trained on augmented data.

DISCUSSION

These findings collectively disprove the overfitting hypothesis as the primary cause of poor generalization of the proposed stacking ensemble model. Subsequently, we strongly suggests that the poor performance of the stacking model due to the inability of classical models to accurately capture complex and domain-robust patterns. This highlights the need for future work to focus on domain generalization and representation learning from large-scale pretrained models like wav2vec2.

Nevertheless, the good performance of the on the in-domain task provides a vital insight. Although this approach may be unsuitable as a general-purpose deepfake detector, it demonstrates substantial effectiveness for specialised and constrained applications. For example, regarding its simplicity and lightness, it could be effectively implemented on edge devices within a specific system where the types of potential fakes are precisely known and clearly defined. In such a controlled environment, the model's superior accuracy and computational efficiency would constitute considerable advantages.

CONCLUSION

We experimented with a novel lightweight stacking ensemble architecture, as presented in this paper. Using XGBoost and Random Forest as the base learners and Multilayer Perceptron (MLP) as the meta-learner. The model obtained a very good performance when it was tested on a 10-fold cross-validation using the Fake or Real (FoR) dataset with accuracy and MCC values greater than 94% and 0.88, respectively. However, when evaluated on an unseen test set, the accuracy is expected to degrade and be closer to 60%. This experiment can be used as a reference to show the generalization ability of the model on data it has never seen before and how it might perform in a real-world application.

To provide an explanation for the generalization, we can follow two possible paths (1) apply heavy data augmentation to avoid overfitting, and (2) attempt to underfit the model by reducing the capacity of the model. However, if we follow these two paths, the results were not as expected, and it shows no significant improvement on the unseen data. The experiments in this paper provide a strong empirical proof that general overfitting is not the reason for generalization failure. It can be argued that there is a distribution mismatch between the seen and unseen data. It could also be due to the classic ML algorithms' generalization gap to an unseen dataset, especially when there are highly complex tasks, such as audio deepfake detection.

On the other hand, it is important to note that classical ML, when combined as an ensemble and fine-tuned, can still deliver high accuracy, and fast inference time, and can be competitive with deep learning approaches in a low resource or real-time deployment setting, with a significantly reduced training and computation cost.

REFRENCES

[1] Abdul, Zrar Kh., and Abdulbasit K. Al-Talabani. 2022. Mel Frequency Cepstral Coefficient and Its Applications: A Review. IEEE Access 10: 122136–122158.
[2] Agarwal, Harsh, Ankur Singh, and Rajeswari D. 2021. Deepfake Detection Using SVM. *In* 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC) Pp. 1245–1249. https://ieeexplore.ieee.org/abstract/document/9532627, accessed July 6, 2025.
[3] Breiman, Leo. 2001 Random Forests. Machine Learning 45(1): 5–32.
[4] Chen, Tianqi, and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. *In* Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining Pp. 785–794.
[5] Das, Arnab, Suhita Ghosh, Tim Polzehl, and Sebastian Stober. 2023. StarGAN-VC++: Towards Emotion Preserving Voice Conversion Using Deep Embeddings. arXiv.
[6] Hamza, Ameer, Abdul Rehman Rehman Javed, Farkhund Iqbal, et al. 2022. Deepfake Audio Detection via MFCC Features Using Machine Learning. IEEE Access 10: 134018–134028.
[7] Jung, Jee-weon, Hee-Soo Heo, Hemlata Tak, et al. 2021. AASIST: Audio Anti-Spoofing Using Integrated Spectro-Temporal Graph Attention Networks. arXiv.Org.
[8] Kameoka, Hirokazu, Takuhiro Kaneko, Kou Tanaka, and Nobukatsu Hojo. 2018. StarGAN-VC: Non-Parallel Many-to-Many Voice Conversion with Star Generative Adversarial Networks.
[9] Kaneko, Takuhiro, Hirokazu Kameoka, Kou Tanaka, and Nobukatsu Hojo. 2020. CycleGAN-VC3: Examining and Improving CycleGAN-VCs for Mel-Spectrogram Conversion. arXiv.
[10] Kim, Jaehyeon, Jungil Kong, and Juhee Son. 2021. Conditional Variational Autoencoder with Adversarial Learning for End-to-End Text-to-Speech. *In* Proceedings of the 38th International Conference on Machine Learning Pp. 5530–5540. PMLR.
[11] Long, Ziang, Yunling Zheng, Meng Yu, and Jack Xin. 2022. Enhancing Zero-Shot Many to Many Voice Conversion via Self-Attention VAE with Structurally Regularized Layers. *In* 2022 5th International Conference on Artificial Intelligence for Industries (AI4I) Pp. 59–63.
[12] Modak, Sharmistha, Arnab Kumar Das, and Ruchira Naskar. 2025. SpecViT: A Custom Vision-Transformer Based Approach for Audio Deepfake Detection. *In* ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) Pp. 1–5.
[13] Qian, Kaizhi, Yang Zhang, Shiyu Chang, Xuesong Yang, and Mark Hasegawa-Johnson. 2019. AutoVC: Zero-Shot Voice Style Transfer with Only Autoencoder Loss. *In* Proceedings of the 36th International Conference on Machine Learning Pp. 5210–5219. PMLR.
[14] Rana, Preeti, and Sandhya Bansal. 2024. Exploring Deepfake Detection: Techniques, Datasets and Challenges. International Journal of Computing and Digital Systems 15(1): 769–781.
[15] Reimao, Ricardo, and Vassilios Tzerpos. 2019. FoR: A Dataset for Synthetic Speech Detection. *In* 2019 International Conference on Speech Technology and Human-Computer Dialogue (SpeD) Pp. 1–10. Timisoara, Romania: IEEE.
[16] Ren, Yi, Yangjun Ruan, Xu Tan, et al. 2019. FastSpeech: Fast, Robust and Controllable Text to Speech. *In* Proceedings of the 33rd International Conference on Neural Information Processing Systems Pp. 3171–3180. , 285. Red Hook, NY, USA: Curran Associates Inc.
[17] Shen, Jonathan, Ruoming Pang, Ron J. Weiss, et al. 2018. Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions. *In* 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) Pp. 4779–4783. IEEE.
[18] Tak, Hemlata, Jose Patino, Massimiliano Todisco, et al. 2021. End-to-End Anti-Spoofing with RawNet2. *In* ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) Pp. 6369–6373.
[19] Toda, Tomoki, Alan W. Black, and Keiichi Tokuda. 2007. Voice Conversion Based on Maximum-Likelihood Estimation of Spectral Parameter Trajectory. IEEE Transactions on Audio, Speech, and Language Processing 15(8): 2222–2235.
[20] Wang, Xin, Héctor Delgado, Hemlata Tak, et al. 2025. ASVspoof 5: Design, Collection and Validation of Resources for Spoofing, Deepfake, and Adversarial Attack Detection Using Crowdsourced Speech. arXiv.
[21] Xiao, Xiong, Xiaohai Tian, Steven Du, et al. 2015. Spoofing Speech Detection Using High Dimensional Magnitude and Phase Features: The NTU System for ASVspoof 2015 Challenge.
[22] Yi, Jiangyan, Chenglong Wang, Jianhua Tao, et al. 2023. Audio Deepfake Detection: A Survey. arXiv. http://arxiv.org/abs/2308.14970.

Journal of Information Systems Engineering and Management
2025, 10(57s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

[23] Yue, Feng, Jiale Chen, Zhaopin Su, Niansong Wang, and Guofu Zhang. 2022. Audio Spoofing Detection Using Constant-Q Spectral Sketches and Parallel-Attention SE-ResNet. *In* European Symposium on Research in Computer Security. 756–762. Cham: Springer Nature Switzerland.