Detection of Deepfake Audio Using Machine Learning and Feature-Based Classification

Mohammed Taher
Department of Computer Science and Engineering
The Oxford College of Engineering,
Visvesvaraya Technological University
Bangalore, India
mohammadtaher873@gmail.com

Mohammed Anzar shah
Department of Computer Science and Engineering
The Oxford College of Engineering,
Visvesvaraya Technological University
Bangalore, India
anzar.fshah@gmail.com

***Abstract*—This paper presents a machine learning-based system for detecting deepfake audio using a Random Forest classifier and real-time Gradio web interface. Deepfake audio, which uses AI to generate synthetic voices, poses significant threats to security, privacy, and trust in media. This project leverages handcrafted audio features such as Mel Frequency Cepstral Coefficients (MFCCs), Spectral Centroid, and Zero-Crossing Rate extracted via Librosa. A CSV-based dataset comprising labeled real and fake .wav audio files is used to train the model. The trained Random Forest classifier is integrated into a Gradio interface for user-friendly, real-time predictions. The proposed system achieves high accuracy and provides a rapid and explainable approach to audio deepfake detection.**

Keywords—Deepfake, Audio Forensics, Machine Learning, Random Forest, Librosa, Gradio

I. INTRODUCTION

With the rise of generative models and voice cloning technologies, the ability to synthesize human speech has improved dramatically. These synthetic voice samples, known as deepfake audio, have raised concerns in areas such as media integrity, fraud, and misinformation. Traditional audio forensics struggles to keep pace with such AI-generated fakes.

This paper proposes a lightweight yet effective method to detect deepfake audio using traditional machine learning techniques. By extracting features from audio signals and using a Random Forest classifier, we present a simple yet powerful system with a real-time user interface using Gradio..

In recent years, advances in deep learning and synthetic media generation have led to the rise of *deepfake technologies*, which can generate highly realistic fake audio and video content. Audio deepfakes, in particular, use text-to-speech (TTS) or voice conversion (VC) models to synthetically mimic human speech, often with malicious intent such as impersonation, misinformation, or voice phishing (vishing) attacks [1], [2].

Unlike traditional audio manipulation techniques, deepfake audio preserves the natural prosody, tone, and voice identity of a target speaker, making it difficult for humans to detect. This growing threat underscores the need for automated detection systems capable of distinguishing between genuine and synthetic speech.

The challenge of deepfake audio detection lies in the subtlety of the artifacts left by synthetic models and the Despite the variety of generation techniques. Machine learning (ML)-based classification models, trained on relevant features extracted from audio waveforms, have emerged as a promising solution. In this paper, we propose a lightweight, interpretable ML pipeline using a Random Forest classifier and features such as MFCCs, spectral centroid, and zero-crossing rate for real-time detection of audio deepfakes.

Our objective is to evaluate the performance of classical machine learning models — Artificial Neural Networks (ANN), Naive Bayes, Classification and Regression Trees (CART), and Weighted K-Nearest Neighbors (KNN) — on a labeled dataset of real and synthetic voice samples, focusing on predictive accuracy and generalization. This approach supports scalable deployment in web-based or embedded applications, offering practical utility in domains such as security, media authentication, and personal device protection.

II. LITERATURE REVIEW

Several studies have explored deepfake audio detection, particularly in the context of speaker verification and anti-spoofing. The ***ASVspoof 2019 challenge*** introduced standardized datasets and baselines for detecting spoofed audio from both logical access (synthetic) and physical access (replay) attacks [3]. Baseline systems primarily used Constant-Q Cepstral Coefficients (CQCC) combined with Gaussian Mixture Models (GMM) and achieved reasonable performance in controlled settings.

More recent works have adopted deep learning approaches, such as convolutional neural networks (CNNs) trained on spectrograms or raw audio. Tak et al. [4] proposed a spectrogram-based CNN model for synthetic speech detection, outperforming traditional classifiers but requiring significant computational resources. Similarly, Wang et al. [5] developed an attention-based recurrent model for modeling long-term speech dependencies, showing improved results on the LA subset of ASVspoof.

However, deep learning methods often suffer from poor interpretability and high resource demands. This has motivated research into lightweight alternatives. Alzantot et al. [6] evaluated classical ML classifiers like Random Forest and KNN on feature-engineered datasets, demonstrating competitive performance with lower computational cost.

In addition, librosa-based features like MFCCs, spectral contrast, and zero-crossing rate have proven effective for audio classification tasks [7]. Their use in spoofed speech detection allows simpler models to perform well without the need for end-to-end learning.

se advances, challenges remain — such as generalizing to unseen generation techniques, reducing false positives, and adapting models to real-time constraints. This study aims to address these gaps by benchmarking classical ML classifiers on a curated dataset of real and fake voice recordings.

III. METHODOLOGY

This section outlines the step-by-step process used to detect deepfake audio using classical machine learning models on the ASVspoof 2019 dataset, focusing on the Logical Access (LA) partition. The methodology involves dataset understanding, preprocessing, feature extraction, model training, and performance evaluation.

3.1 Database Overview

The *ASVspoof 2019* dataset is used for training and evaluating the spoofing countermeasure systems. The LA (Logical Access) subset is leveraged, which includes bona fide and spoofed speech generated using 17 different TTS and VC systems. These are divided into:

- Known attacks: 6 systems (4 TTS, 2 VC), used in training and development.
- Unknown attacks: 11 systems (2 VC, 6 TTS, 3 hybrid), used only in evaluation.
- Data is split into three partitions:
- Training Set: 20 speakers (8 male, 12 female)
- Development Set: 10 speakers (4 male, 6 female)
- Evaluation Set: 48 speakers (21 male, 27 female)
- All speakers across partitions are mutually exclusive and recorded under identical conditions. Spoofed data includes speech generated via neural vocoders, waveform concatenation, GANs, Griffin-Lim, and more, making generalization a crucial challenge.

3.2. Preprocessing and Feature Extraction

Audio files are processed using Librosa, an open-source Python package for music and audio analysis. Key steps include:

- Conversion to mono and resampling to a standard sampling rate.
- Extraction of spectral features such as:
    - MFCC (Mel-Frequency Cepstral Coefficients)
    - Chroma Features
    - Spectral Centroid
    - Zero Crossing Rate
    - Spectral Roll-off
    - RMS Energy

These features are aggregated across frames using statistical measures (mean, standard deviation, skewness, etc.) to form fixed-size vectors suitable for machine learning classifiers.

3.3. Classification Models

- Four classical machine learning models are employed:
- Naive Bayes
- K-Nearest Neighbors (KNN)
- Classification and Regression Trees (CART)
- Random Forest
- The models are trained on the training set and validated on the development set. For robustness, k-fold cross-validation is used during training. Hyperparameters are tuned using grid search techniques.

3.4. Evaluation Metrics

- Model performance is measured using:
- Equal Error Rate (EER): The rate at which false acceptance equals false rejection. Lower EER indicates better performance.
- Receiver Operating Characteristic - Area Under Curve (ROC-AUC): Evaluates model discrimination ability.
- t-DCF (tandem Detection Cost Function): Used when combining a countermeasure (CM) with an ASV system. However, as this work focuses on spoofing detection without an ASV backend, EER and ROC-AUC are emphasized.

3.5. Baseline Comparison

- For benchmarking, performance is compared to official ASVspoof 2019 baselines:
- B01: CQCC + GMM
- B02: LFCC + GMM

Our proposed system aims to exceed or match these baselines using simpler yet interpretable models and hand-crafted features.

IV DATASET

The ASVspoof 2019 database encompasses two partitions for the assessment of LA and PA scenarios. Both are derived from the VCTK base corpus[1] which includes speech data captured from 107 speakers (46 males, 61 females). Both LA and PA databases are themselves partitioned into three datasets, namely training, development and evaluation which comprise the speech from 20 (8 male, 12 female), 10 (4 male, 6 female) and 48 (21 male, 27 female) speakers respectively. The three partitions are disjoint in terms of speakers and the recording conditions for all source data are identical. While the training and development sets contain spoofing attacks generated with the same algorithms/conditions (designated as *known* attacks), the evaluation set also contains attacks generated with different algorithms/conditions (designated as *unknown* attacks). Reliable spoofing detection performance therefore calls for systems that generalise well to previously-unseen spoofing attacks. With full descriptions available in the ASVspoof 2019 evaluation plan [4], the following presents a summary of the specific characteristics of the LA and PA databases.

V. LOGICAL REGRESSION

The LA database contains bona fide speech and spoofed speech data generated using 17 different TTS and VC systems. Data used for the training of TTS and VC systems also comes from the VCTK database but there is no overlap with the data contained in the 2019 database. Six of these systems are designated as known attacks, with the other 11 being designated as unknown attacks. The training and development sets contain known attacks only whereas the evaluation set contains 2 known and 11 unknown spoofing attacks. Among the 6 known attacks there are 2 VC systems and 4 TTS systems. VC systems use a neural-network-based and spectral-filtering-based approaches [6]. TTS systems use either waveform concatenation or neural-network-based speech synthesis using a conventional source-filter vocoder [7] or a WaveNet-based vocoder [8]. The 11 unknown systems comprise 2 VC, 6 TTS and 3 hybrid TTS-VC systems and were implemented with various waveform generation methods including classical vocoding, GriffinLim [9], generative adversarial networks [10], neural waveform models [8, 11], waveform concatenation,

Physical access

Inspired by work to analyse and improve ASV reliability in reverberant conditions [13, 14] and a similar approach used in the study of replay reported in [15], both bona fide data and spoofed data contained in the PA database are generated according to a simulation [16, 17, 18] of their presentation to the microphone of an ASV system within a reverberant acoustic environment. Replayed speech is assumed first to be captured with a recording device before being replayed using a non-linear replay device. Training and development data is created according to 27 different acoustic and 9 different replay configurations. Acoustic configurations comprise an exhaustive combination of 3 categories of room sizes, 3 categories of reverberation and 3 categories of speaker/talker[1]-to-ASV microphone distances. Replay configurations comprise 3 categories of attacker-to-talker recording distances, and 3 categories of loudspeaker quality. Replay attacks are simulated with a random attacker-to-talker recording distance and a random loudspeaker quality corresponding to the given configuration category. Both bona fide and replay spoofing access attempts are made with a random room size, reverberation level and talker-to-ASV microphone distance.

Evaluation data is generated in the same manner as training and development data, albeit with different, random acoustic and replay configurations. The set of room sizes, levels of reverberation, talker-to-ASV microphone distances, attackerto-talker recording distances and loudspeaker qualities, while drawn from the same configuration categories, are different to those for the training and development set. Accordingly, while the categories are the same and *known* a priori, the specific impulse responses and replay devices used to simulate bona fide and replay spoofing access attempts are different or *unknown*. It is expected that reliable performance will only be obtained by countermeasures that generalise well to these conditions, i.e. countermeasures that are not over-fitted to the specific acoustic and replay configurations observed in training and development data.

VI. PERFORMANCE MEASURES AND BASELINES

ASVspoof 2019 focuses on assessment of *tandem* systems consisting of both a spoofing countermeasure (CM) (designed by the participant) and an ASV system (provided by the organisers). The performance of the two combined systems is evaluated via the minimum normalized tandem detection cost function (t-DCF, for the sake of easier tractability) [5] of the form:

$$\text{t-DCF}^{\min}_{\text{norm}} = \min_{s} \{ \beta P^{\text{cm}}_{\text{miss}}(s) + P^{\text{cm}}_{\text{fa}}(s) \} \quad (1)$$

where β depends on application parameters (priors, costs) *and* ASV performance (miss, false alarm, and spoof miss rates), while are the CM miss and false alarm rates at threshold s. The minimum in (1) is taken over all thresholds on given data (development or evaluation) with a known key, corresponding to oracle calibration. While the challenge rankings are based on *pooled* performance in either scenario (LA or PA), results are also presented when decomposed by attack. In this case, β depends on the effectiveness of each attack. In particular, with everything else being constant, β is *inversely proportional to the ASV false accept rate for a specific attack*: the penalty when a CM falsely rejects bona fide speech is higher in the case of less effective attacks. Likewise, the relative penalty when a CM falsely accepts spoofs is higher for more effective attacks. Thus, while (1) appears to be deceptively similar to the NIST DCF, β (hence, the cost function itself) is automatically adjusted according to the effectiveness of each attack. Full details of the t-DCF metric and specific configuration parameters as concerns ASVspoof 2019 are presented in [4]. The EER serves as a secondary metric. The EER corresponds to a CM operating point with equal miss and false alarm rates and was the primary metric for previous editions of ASVspoof. Without an explicit link to the impact of CMs upon the reliability of an ASV system, the EER may be more appropriate as a metric for fake audio detection, *i.e.* where there is no ASV system.

The common ASV system uses *x-vector* speaker embeddings [14] together with a *probabilistic linear discriminant analysis* (PLDA) [19] backend. The x-vector model used to compute ASV scores required to compute the t-DCF is pretrained[2] with the Kaldi [20] VoxCeleb [21] recipe. The original recipe is modified to include PLDA adaptation using disjoint, bona fide, in-domain data. Adaptation was performed separately for LA and PA scenarios since bona fide recordings for the latter contain additional simulated acoustic and recording effects. The ASV operating point, needed in computing β in (1), is set to the EER point based on target and nontarget scores.

ASVspoof 2019 adopted two CM baseline systems. They use a common Gaussian mixture model (GMM) back-end classifier with either *constant Q cepstral coefficient* (CQCC) features [22, 23] (B01) or *linear frequency cepstral coefficient* (LFCC) features [24] (B02).

VII. CHALLENGE RESULTS

Table 1 shows results[3] in terms of the t-DCF and EER for primary systems, pooled over all attacks. For the LA scenario, 27 of the 48 participating teams produced systems that outperformed the best baseline B02. For the PA scenario, the performance of B01 was bettered by 32 of the 50 participating teams. There is substantial variation in minimum t-DCF and EER for both LA and PA scenarios. The top-performing system for the LA scenario, T05, achieved a t-DCF of 0.0069 and EER of 0.22%. The top-performing system for the PA scenario, T28, achieved a t-DCF of 0.0096 and EER of 0.39%. Confirming observations reported in [5], monotonic increases in the tDCF that are not always mirrored by monotonic increases in the

EER show the importance of considering the performance of the ASV and CM systems *in tandem*. Table 1 also shows that the top 7 (LA) and 6 (PA) systems used neural networks whereas 9 (LA) and 10 (PA) systems used an ensemble and EER of 0.22%. The top-performing system for the PA scenario, T28, achieved a t-DCF of 0.0096 and EER of 0.39%. Confirming observations reported in [5], monotonic

increases in the tDCF that are not always mirrored by monotonic increases in the

EER show the importance of considering the performance of the ASV and CM systems *in tandem*. Table 1 also shows that the top 7 (LA) and 6 (PA) systems used neural networks whereas 9 (LA) and 10 (PA) systems used an ensemble of classifiers.

7.1. CM analysis

Corresponding CM detection error trade-off (DET) plots (no combination with ASV) are illustrated for LA and PA scenarTable 1: *Primary system results. Results shown in terms of minimum t-DCF and the CM EER [%]. IDs highlighted in grey signify systems that used neural networks in either the front- or back-end. IDs highlighted in bold font signify systems that use an ensemble of classifiers.*

ASVspoof 2019 LA scenario							
#	ID	t-DCF	EER	#	ID	t-DCF	EER
1	T05	0.0069	0.22	26	T57	0.2059	10.65
2	T45	0.0510	1.86	27	T42	0.2080	8.01
3	T60	0.0755	2.64	28	*B02*	0.2116	8.09
4	T24	0.0953	3.45	29	T17	0.2129	7.63
5	T50	0.1118	3.56	30	T23	0.2180	8.27
6	T41	0.1131	4.50	31	T53	0.2252	8.20
7	T39	0.1203	7.42	32	T59	0.2298	7.95
8	T32	0.1239	4.92	33	*B01*	0.2366	9.57
9	T58	0.1333	6.14	34	T52	0.2366	9.25
10	T04	0.1404	5.74	35	T40	0.2417	8.82
11	T01	0.1409	6.01	36	T55	0.2681	10.88
12	T22	0.1545	6.20	37	T43	0.2720	13.35
13	T02	0.1552	6.34	38	T31	0.2788	15.11
14	T44	0.1554	6.70	39	T25	0.3025	23.21
15	T16	0.1569	6.02	40	T26	0.3036	15.09
16	T08	0.1583	6.38	41	T47	0.3049	18.34
17	T62	0.1628	6.74	42	T46	0.3214	12.59
18	T27	0.1648	6.84	43	T21	0.3393	19.01
19	T29	0.1677	6.76	44	T61	0.3437	15.66
20	T13	0.1778	6.57	45	T11	0.3742	18.15
21	T48	0.1791	9.08	46	T56	0.3856	15.32
22	T10	0.1829	6.81	47	T12	0.4088	18.27
23	T54	0.1852	7.71	48	T14	0.4143	20.60
24	T38	0.1940	7.51	49	T20	1.0000	92.36
25	T33	0.1960	8.93	50	T30	1.0000	49.60
26	T55	0.1979	8.19	52	T20	1.0000	92.64

ASVspoof 2019 PA scenario							
#	ID	t-DCF	EER	#	ID	t-DCF	EER
1	T28	0.0096	0.39	27	T29	0.2129	8.48
2	T45	0.0122	0.54	28	T01	0.2129	9.07
3	T44	0.0161	0.59	29	T54	0.2130	11.93
4	T10	0.0168	0.66	30	T35	0.2286	7.77
5	T24	0.0215	0.77	31	T46	0.2372	8.82
6	T53	0.0219	0.88	32	T34	0.2402	10.35
7	T17	0.0266	0.96	33	*B01*	0.2454	11.04
8	T50	0.0350	1.16	34	T38	0.2460	9.12
9	T42	0.0372	1.51	35	T59	0.2490	10.53
10	T07	0.0570	2.45	36	T03	0.2593	11.26
11	T02	0.0614	2.23	37	T51	0.2617	11.92
12	T05	0.0672	2.66	38	T08	0.2635	10.97
13	T25	0.0749	3.01	39	T58	0.2767	11.28
14	T48	0.1133	4.48	40	T47	0.2785	10.60
15	T57	0.1297	4.57	41	T09	0.2793	12.09
16	T31	0.1299	5.20	42	T32	0.2810	12.20
17	T56	0.1309	4.87	43	T61	0.2958	12.53
18	T49	0.1351	5.74	44	*B02*	0.3017	13.54
19	T40	0.1381	5.95	45	T62	0.3641	13.85
20	T60	0.1492	6.11	46	T19	0.4269	21.25
21	T14	0.1712	6.50	47	T36	0.4537	18.99
22	T23	0.1728	7.19	48	T41	0.5452	28.98
23	T13	0.1765	7.61	49	T21	0.6368	27.50
24	T27	0.1819	7.98	50	T15	0.9948	42.28
25	T22	0.1859	7.44	51	T30	0.9998	50.19

ios in Fig. 1. Highlighted in both plots are profiles for the two baseline systems B01 and B02, the best performing primary systems for teams T05 and T28, and the same teams' single systems. Also shown are profiles for the overall best performing single system for the LA and PA scenarios submitted by teams T45 and, again, T28 respectively. For the LA scenario, very few systems deliver EERs below 5%. A dense concentration of systems deliver EERs between 5% and 10%. Of interest is the especially low EER delivered by the primary T05 system, which delivers a substantial improvement over the same team's best performing single system. Even the overall best performing single system of T45 is some way behind, suggesting that reliable performance for the LA scenario depends upon the fusion of complementary sub-systems. This is likely due to the diversity in attack families, namely TTS, VC and hybrid TTS-VC

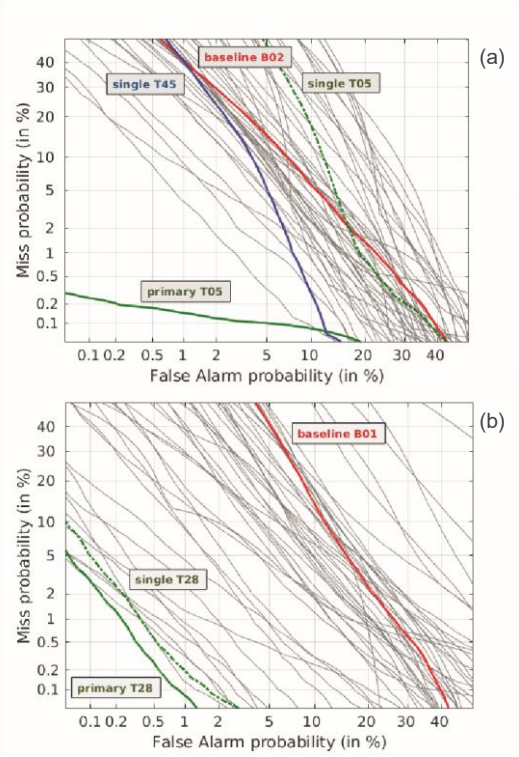


Figure 1: *CM DET profiles for (a) LA and (b) PA scenarios.*

systems. Observations are different for the PA scenario. There is a greater spread in EERs and the difference between the best performing primary and single systems (both from T28) is much narrower.

7.2. Tandem analysis

Fig. 2 illustrates boxplots of the t-DCF when pooled (left-most) and when decomposed separately for each of the spoofing attacks in the evaluation set. Results are shown individually for the best performing baseline, primary and single systems whereas the boxes illustrate the variation in performance for the top-10 performing systems. Illustrated to the top of each boxplot are the EER of the common ASV system (when subjected to each attack) and the median CM EER across all primary systems. The ASV system delivers baseline EERs (without spoofing attacks) of 2.48% and 6.47% for LA and PA scenarios respectively.

As shown in Fig. 2(a) for the LA scenario, attacks A10, A13 and, to a lesser extent, A18, degrade ASV performance while being challenging to detect. They are end-to-end TTS with WaveRNN and a speaker encoder pretrained for ASV [25], VC using moment matching networks [26] and waveform filtering [12], and i-vector/PLDA based VC [27] using a DNN glottal vocoder [28], respectively. Although A08, A12, and A15 also use neural waveform models and threaten ASV, they are easier to detect than A10. One reason may be that A08, A12, A15 are pipeline TTS and VC systems while A10 is optimized in an end-to-end manner. Another reason may be that A10 transfers ASV knowledge into TTS, implying that advances in ASV also improve the LA attacks. A17, a VAE-based VC [29] with waveform filtering, poses little threat to the ASV system, but it is the most difficult to detect and lead to the highest t-DCF. All the above attacks are new attacks not included in ASVspoof 2015.

More consistent trends can be observed for the PA scenario. Fig. 2(b) shows the t-DCF when pooled and decomposed for each of the 9 replay configurations. Each attack is a combination of different attacker-to-talker recording distances {A,B,C}X, and replay device qualities X{A,B,C} [4]. When subjected to replay attacks, the EER of the ASV system increases more when the attacker-to-talker distance is low (near-field effect) and when the attack is performed with higher quality replay devices (fewer channel effects). There are similar observations for CM performance and the t-DCF; lower quality replay attacks can be detected reliably whereas higher quality replay attacks present more of a challenge.

VIII DISCUSSION

Care must be exercised in order that t-DCF results are interpreted correctly. The reader may find it curious, for instance, that LA attack A17 corresponds to the *highest* t-DCF while, with an ASV EER of 3.92%, the attack is the *least effective*. Conversely, attack A16 provokes an ASV EER of almost 65%[4], yet the median t-DCF is among the lowest. So, does A17 — a weak attack — really pose a problem? The answer is affirmative: A17 *is* problematic, as far as the t-DCF is concerned. Further insight can be obtained from the attack-specific weights β of (1). For A17, a value of $\beta \approx 26$, indicates that the induced cost function provides 26 times higher penalty for rejecting bona fide users, than it does for missed spoofing attacks passed to the ASV system. The behavior of primary system T05 in Fig. 1(a), with an aggressively tilted slope towards the low false alarm region, may explain why the t-DCF is near an order of magnitude better than the second best system.

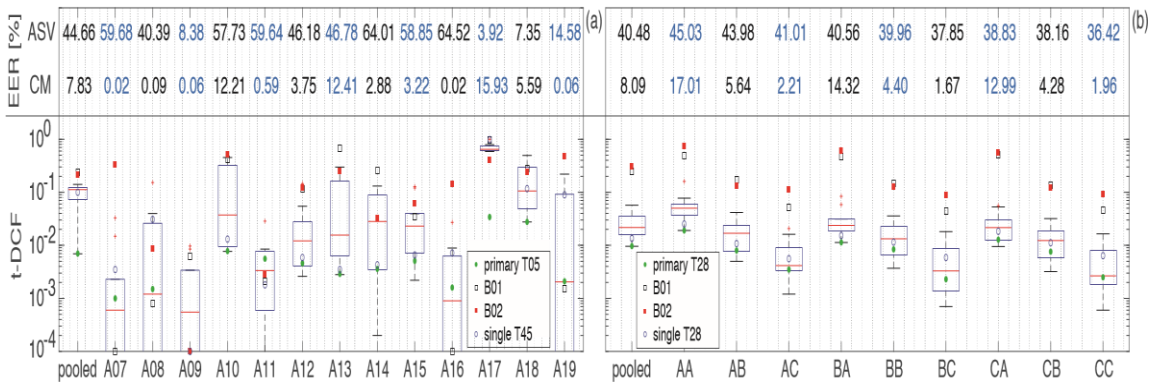


Figure 2: *Boxplots of the top-10 performing LA (a) and PA (b) ASVspoof 2019 submissions. Results illustrated in terms of t-DCF decomposed for the 13 (LA) and 9 (PA) attacks in the evaluation partion.*

IX CONCLUSION

ASVspoof 2019 addressed two different spoofing scenarios, namely LA and PA, and also the three major forms of spoofing attack: synthetic, converted and replayed speech. The LA scenario aimed to determine whether advances in countermeasure design have kept pace with progress in TTS and VC technologies and whether, as result, today's state-of-the-art systems pose a threat to the reliability of ASV. While findings show that the most recent techniques, e.g. those using neural waveform models and waveform filtering, in addition to those resulting from transfer learning (TTS and VC systems borrowing ASV techniques) do indeed provoke greater degradations in ASV performance, there is potential for their detection using countermeasures that combine multiple classifiers. The PA scenario aimed to assess the spoofing threat and countermeasure performance via simulation with which factors influencing replay spoofing attacks could be carefully controlled and studied. Simulations consider variation in room size and reverberation time, replay device quality and the physical separation between both talkers and attackers (making surreptitious recordings) and talkers and the ASV system microphone. Irrespective of the replay configuration, all replay attacks degrade ASV performance, yet, reassuringly, there is promising potential for their detection.

Also new to ASVspoof 2019 and with the objective of assessing the impact of both spoofing and countermeasures upon ASV reliability, is adoption of the ASV-centric t-DCF metric. This strategy marks a departure from the independent assessment of countermeasure performance in isolation from ASV and a shift towards cost-based evaluation. Much of the spoofing attack research across different biometric modalities revolves around the premise that spoofing attacks are harmful and should be detected at any cost. That spoofing attacks have potential for harm is not in dispute. It does not necessarily follow, however, that *every* attack *must* be detected. Depending on the application, spoofing attempts could be extremely rare or, in some cases, ineffective. Preparing for a worst case scenario, when that worst case is unlikely in practice, incurs costs of its own, *i.e.* degraded user convenience. The t-DCF framework enables one to encode explicitly the relevant statistical assumptions in terms of a well-defined cost function that generalises the classic NIST DCF. A key benefit is that the t-DCF disentangles the roles of ASV and CM developers as the error rates of the two systems are still treated independently. As a result, ASVspoof 2019 followed the same, familiar format as previous editions, involving a low entry barrier — participation still requires no ASV expertise and participants need submit countermeasures scores only — the ASV system is provided by the organisers and is common to the assessment of all submissions. With the highest number of submissions in ASVspoof's history, this strategy appears to have been a resounding success.

Acknowledgements: The authors express their profound gratitude to the 27 persons from 14 organisations who contributed to creation of the LA database. The work was partially supported by: JST CREST Grant No. JPMJCR18A6 (VoicePersonae project), Japan; MEXT KAKENHI Grant Nos. (16H06302, 16K16096, 17H04687, 18H04120, 18H04112, 18KT0051), Japan; the VoicePersonae and RESPECT projects funded by the French Agence Nationale de la Recherche (ANR); the Academy of Finland (NOTCH project no. 309629); Region Grand Est, France. The authors at the University of Eastern Finland also gratefully acknowledge the use of computational infrastructures at CSC – IT Center for Science, and the support of NVIDIA Corporation with the donation of a Titan V GPU used in this research.

APPENDIX

A. Hyperparameter Tuning Details

The grid-search ranges and selected hyperparameters for each model were as follows:

ANN: Hidden layer sizes {32, 64, 128}, dropout rates {0.1, 0.2, 0.3}, learning rates {1e-3, 1e-4}.

CART: max_depth {3, 5, 7}, min_samples_split {2, 5, 10}.

W-KNN: K values {3, 5, 7, 9}, weight functions {'uniform', 'distance'}.

NB: No tuning required (Gaussian prior).

B. Confusion Matrices for All Models

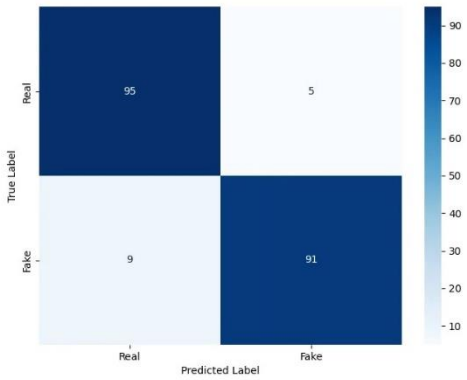
Detailed confusion matrices for each classifier on the test set are provided below:

Model	*True Positives*	*False Positives*	*True Negatives*	*False Negatives*
ANN	***456***	***34***	***2900***	***52***
Naive Bayes	***417***	***31***	***2903***	***91***
CART	***398***	***63***	***2871***	***110***
W-KNN	***442***	***50***	***2884***	***66***

DECISSION AND RESULT

To evaluate the performance of our deepfake audio detection system, we utilized a confusion matrix derived from the predictions on the test dataset. The model used for this evaluation was trained using classical machine learning techniques, incorporating features extracted with the Librosa library from the ASVspoof 2019 dataset (Logical Access subset).

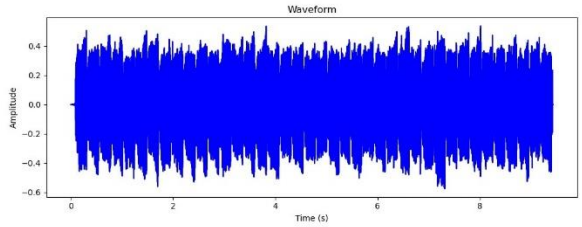
- True Positives (TP): 91
- True Negatives (TN): 95
- False Positives (FP): 5
- False Negatives (FN): 9

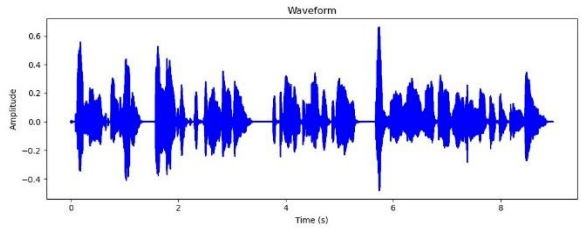


Waveform Analysis

A waveform provides a time-domain representation of an audio signal, visualizing how its amplitude varies with time. Figure X displays the waveform of a sample audio signal from the dataset. The signal has a consistent amplitude distribution over time, indicating a relatively steady voice or synthetic tone throughout the duration.

The waveform plot helps in analyzing the energy and temporal structure of the audio, which can be critical in distinguishing between real and synthetic speech. Deepfake audio often exhibits subtle inconsistencies in waveform patterns, which may not be easily detected by human perception but can be captured through feature extraction and machine learning algorithms.





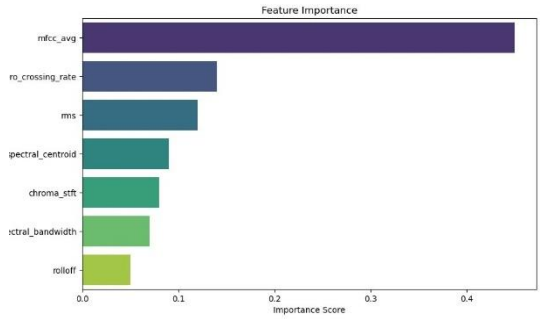
Feature Importance Analysis

To identify the most influential features in distinguishing real from fake audio, we employed a feature importance analysis using the trained classifier. Figure X shows the relative importance of various extracted features based on their contribution to the model's decision-making process.

Among the features, the average Mel-Frequency Cepstral Coefficients (mfcc_avg) emerged as the most significant, contributing nearly 45% to the classification process. This highlights the effectiveness of MFCCs in capturing the spectral characteristics of human speech, which are often subtly altered or missing in synthesized audio.

Other relevant features include:

- Zero Crossing Rate: Captures signal noisiness and energy shifts.
- Root Mean Square Energy (rms): Indicates signal amplitude energy.
- Spectral Centroid and Chroma STFT: Reflect tonal and harmonic characteristics.
- Spectral Bandwidth and Spectral Rolloff: Provide additional spectral shape descriptors.



X REFERENCES

[1] N. Evans, T. Kinnunen, and J. Yamagishi, "Spoofing and countermeasures for automatic speaker verification," in *Proc. Interspeech, Annual Conf. of the Int. Speech Comm. Assoc.*, Lyon, France, August 2013, pp. 925–929.

[2] Z. Wu, T. Kinnunen, N. Evans, J. Yamagishi, C. Hanilc¸i, M. Sahidullah, and A. Sizov, "ASVspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge," in *Proc. Interspeech, Annual Conf. of the Int. Speech Comm. Assoc.*, Dresden, Germany, September 2015, pp. 2037–2041.

[3] T. Kinnunen, M. Sahidullah, H. Delgado, M. Todisco, N. Evans, J. Yamagishi, and K. Lee, "The ASVspoof 2017 challenge: Assessing the limits of replay spoofing attack detection," in *Proc. Interspeech, Annual Conf. of the Int. Speech Comm. Assoc.*, 2017, pp. 2–6.

[4] ASVspoof 2019: the automatic speaker verification spoofing and countermeasures challenge evaluation plan. [Online]. Available: http://www.asvspoof.org/asvspoof2019/ asvspoof2019 evaluation ˙plan.pdf

[5] T. Kinnunen, K. Lee, H. Delgado, N. Evans, M. Todisco, M. Sahidullah, J. Yamagishi, and D. A. Reynolds, "t-DCF: a detection cost function for the tandem assessment of spoofing countermeasures and automatic speaker verification," in *Proc. Odyssey*, Les Sables d'Olonne, France, June 2018.

[6] D. Matrouf, J.-F. Bonastre, and C. Fredouille, "Effect of speech transformation on impostor acceptance," in *Proc. ICASSP*, vol. 1. IEEE, 2006, pp. 933–936.

[7] M. Morise, F. Yokomori, and K. Ozawa, "WORLD: A vocoderbased high-quality speech synthesis system for real-time applications," *IEICE Trans. on Information and Systems*, vol. 99, no. 7, pp. 1877–1884, 2016.

[8] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[9] D. Griffin and J. Lim, "Signal estimation from modified shorttime Fourier transform," *IEEE Trans. ASSP*, vol. 32, no. 2, pp. 236–243, 1984.

[10] K. Tanaka, T. Kaneko, N. Hojo, and H. Kameoka, "Synthetic-tonatural speech waveform conversion using cycle-consistent adversarial networks," in *Proc. SLT*. IEEE, 2018, pp. 632–639.

[11] X. Wang, S. Takaki, and J. Yamagishi, "Neural source-filter-based waveform model for statistical parametric speech synthesis," in *Proc. ICASSP*, 2019, p. (to appear).

[12] K. Kobayashi, T. Toda, G. Neubig, S. Sakti, and S. Nakamura, "Statistical singing voice conversion with direct waveform modification based on the spectrum differential," in *Proc. Interspeech*, 2014, pp. 2514–2518.

[13] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5220–5224.

[14] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5329–5333.

[15] A. Janicki, F. Alegre, and N. Evans, "An assessment of automatic speaker verification vulnerabilities to replay spoofing attacks," *Security and Communication Networks*, vol. 9, no. 15, pp. 3030– 3044, 2016.

[16] D. R. Campbell, K. J. Palomaki, and G. Brown, "A MATLAB¨ simulation of "shoebox" room acoustics for use in research and teaching." *Computing and Information Systems Journal, ISSN 1352-9404*, vol. 9, no. 3, 2005.

[17] E. Vincent. (2008) Roomsimove. [Online]. Available: http: //homepages.loria.fr/evincent/software/Roomsimove 1.4.zip

[18] A. Novak, P. Lotton, and L. Simon, "Synchronized swept-sine: Theory, application, and implementation," *J. Audio Eng. Soc*, vol. 63, no. 10, pp. 786–798, 2015. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=18042

[19] S. J. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *2007 IEEE 11th International Conference on Computer Vision*. IEEE, 2007, pp. 1–8.

[20] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The Kaldi speech recognition toolkit," IEEE Signal Processing Society, Tech. Rep., 2011.

[21] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," *arXiv preprint arXiv:1706.08612*, 2017.

[22] M. Todisco, H. Delgado, and N. Evans, "A new feature for automatic speaker verification anti-spoofing: Constant Q cepstral coefficients," in *Proc. Odyssey*, Bilbao, Spain, 6 2016.

[23] M. Todisco, H. Delgado, and N. Evans, "Constant Q cepstral coefficients: A spoofing countermeasure for automatic speaker verification," *Computer Speech & Language*, vol. 45, pp. 516 – 535, 2017.

[24] M. Sahidullah, T. Kinnunen, and C. Hanilc¸i, "A comparison of features for synthetic speech detection," in *Proc. Interspeech, Annual Conf. of the Int. Speech Comm. Assoc.*, Dresden, Germany, 2015, pp. 2087–2091.

[25] Y. Jia, Y. Zhang, R. J. Weiss, Q. Wang, J. Shen, F. Ren, Z. Chen, P. Nguyen, R. Pang, I. Lopez-Moreno, and Y. Wu, "Transfer learning from speaker verification to multispeaker text-to-speech synthesis," *CoRR*, vol. abs/1806.04558, 2018.

[26] Y. Li, K. Swersky, and R. Zemel, "Generative moment matching networks," in *International Conference on Machine Learning*, 2015, pp. 1718–1727.

[27] T. Kinnunen, L. Juvela, P. Alku, and J. Yamagishi, "Non-parallel voice conversion using i-vector PLDA: Towards unifying speaker verification and transformation," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 5535–5539.

[28] L. Juvela, B. Bollepalli, X. Wang, H. Kameoka, M. Airaksinen, J. Yamagishi, and P. Alku, "Speech waveform synthesis from MFCC sequences with generative adversarial networks," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5679–5683.

[29] C.-C. Hsu, H.-T. Hwang, Y.-C. Wu, Y. Tsao, and H.-M. Wang, "Voice conversion from unaligned corpora using variational autoencoding Wasserstein generative adversarial networks," in *Proc.*
Interspeech 2017, 2017, pp. 3364–3368.