# Deepfake Audio Detection via Feature Engineering and Machine Learning⋆

Farkhund Iqbal[1], Ahmed Abbasi[2], Abdul Rehman Javed[2,∗], Zunera Jalil[2] and Jamal Al-Karaki[1]

[1]*College of Technological Innovation, Zayed University, UAE*

[2]*Department of Cyber Security, Air University, Islamabad, Pakistan*

## Abstract

With the advancement of technologies in synthetic speech generation, audio deepfake is becoming the most common source of deception. As a result, distinguishing between fake and real audio is becoming increasingly difficult. Several studies were conducted based on machine learning approaches using ASVSpoof or AVSpoof to deal with these challenges. This study experiments on the latest fake or real (FoR) dataset. The audio samples of this dataset are generated using the best text-to-speech (TTS) models. The proposed approach is based on optimal feature engineering and selecting the best machine learning models to detect fake or real audio. Feature engineering employs various methods for extracting features from audio. In contrast, the feature selection method employs the best-performing minimum features, which are then fed to machine learning classifiers. The experiments use six machine learning (ML) classifiers and three subsets of the FoR dataset. The experimental results show that the proposed approach can accurately detect real or fake audio. The proposed method outperforms the baseline method by an accuracy gain of 26%.

## Keywords

Deepfake, Audio classification, Machine learning, Feature extraction, Feature selection

## 1. Introduction

Deep fake is fake data generated using deep learning; though they are entertaining, we have seen many examples where they could be misused and used to spread disinformation. In the past couple of years, deepfakes have been intentionally used to spread fake news. With recent breakthroughs in voice conversion algorithms and text-to-speech [1, 2], synthesizing human speech will become much easier, opening the way for a future in which audio will play an equivalent role in deepfake detection as video [3, 4]. This research investigates the interaction between these two modalities, which might be essential in identifying audio-visual deepfakes. Recent work has concentrated on recognizing visual artifacts and 'fingerprints' from various generating frameworks and detecting local texture inconsistencies produced by face-swapping. Another area of research employs biometric signals, such as recognizing distinct facial motion patterns inherent in certain persons; nevertheless, such ID-specific approaches are restricted in

*Woodstock'22: Symposium on the irreproducible science, June 07–11, 2022, Woodstock, NY*

⋆You can use this document as the template for preparing your publication. We recommend using the latest version of the ceurart style.

∗Corresponding author.

their capacity to generalize to new identities[5, 6].

In the fields of voice forensics and artificial intelligence, various automatic detection algorithms for deepfakes have been created[7]. However, it has been observed that there is a shortage of experiments on humans compared to machines' comprehension of modified audio, mainly controlled speech. Pictures and video data remain the center of human deepfake detection and are the most studied. Many researchers have tried to compare the detecting abilities of humans and machines [1, 8]. Their study was based on their unique deepfake video and picture database, classified into three quality levels: original, high-quality deepfake, and low-quality deepfake. They discovered that the AI detection method beats the human participants, particularly when it comes to low-quality picture deepfakes. This means that humans have trouble differentiating between actual and fraudulent pictures at lesser resolutions[9].

In [10], the most comprehensive video deepfake dataset to date was created and examined. It consisted of three distinct online research that drew in 15,016 people. Participants were either oblivious to AI detection predictions or were made aware of them. They discovered that although naive human participants and AI detection systems had equal detection accuracy, they were deceived by distinct characteristics. Humans made aware of the AI algorithm's prediction judgments performed better on the challenge. However, human accuracy would also suffer if the AI model forecasts were incorrect [11, 12]. The capacity to recognize synthetically created sounds is known as audio deepfake detection. The authors of [13] utilized the ASVspoof dataset to determine if human-based subjective judgments of spoofed can be predicted automatically. They conducted an inter-language investigation in which 68 native English and 206 native Japanese listeners classified samples as faked or benign. They observed that the general trends in both language groups were similar, with slight deviations. The study's authors made no distinction between people with and without IT technical experience. There is also a substantial body of similar work [14, 15] that uses machine learning to detect abnormalities in audio waveforms that may be diagnostic or distinctive of deepfakes. Artifacts such as noisy malfunctioning, unable to match a phase, reverberation, or unintelligibility are examples of such artifacts [16]. There are additional abnormalities that humans do not generally sense, such as various sorts of prolonged quiet, even though such artifacts are critical for AI identification of deepfakes[17]. In this research, we are the first to investigate whether ML can recognize audio deepfakes (i.e., fake audio data generated using deep learning). We investigated how successfully machine learning or deep learning could differentiate audio deepfakes from actual material and whether specific characteristics aided detection performance.

## 1.1. Motivation

The main motivations for this research are presented in this section. Audio deepfake has become the most prevalent source of deceit as synthetic voice synthesis technology advances. Hence, distinguishing between fake and real audio is becoming more challenging. This is why a system that can detect authentic or fraudulent audio in a short amount of time is so important. Previous research has been done on it. However, the present methodologies are computationally intensive. Consequently, the suggested method comprises the best feature extraction methods, resulting in higher classification results in less time.

### 1.2. Contribution

This study proposes a method for detecting fake audio and distinguishing deepfake audio from non-synthetic or actual audio. The following are the specific contributions:

- Provides a unique method for distinguishing between real and counterfeit audios in FoR dataset subsets using feature engineering and machine learning classifiers.
- The supreme feature engineering strategy is employed, which involves the best feature extraction and feature selection approaches to extract the most appropriate features from an audio source.
- Experimental results showed that the proposed approach accurately detects real or fake audio. The proposed method outperforms the baseline method by an accuracy gain of 26%.

### 1.3. Organization

A detailed description of previous studies on deepfake audio detection is presented in section 2. The explanation of the dataset used to detect fake or real audio is presented in section 3. The proposed approach for detecting and classifying real and fake audio is presented in section 4. The experimental results detail is provided in section 5. Finally, this work's future and conclusion are presented in section 6.

## 2. Literature Review

Audio deepfakes are ML/DL-generated audio that appears to be real. To identify audio deepfakes, one must first understand the processes of creation. Detecting audio generated through deepfakes is critical since audio deepfakes have been used in several illegal actions in recent years. A replay attack is one of the three categories used before generative networks; speech synthesis and voice conversion were the three subcategories of audio deepfake approaches. The reader is given this part's most recent and relevant research and tools for each category.

### 2.1. Replay Attack

Replay attacks can work by simply replaying a voice of a target speaker. The main concern is detection, and there are two types of detection. Far-field and cut and paste detection assaults are the two categories. The test segment in far-field detection replay attacks is a far-field microphone recording of the target that has been repeated on a phone handset with a loudspeaker[18]. If a recording is created by cutting and pasting small recordings to simulate the sentence required by a text-based system, it is referred to as a cut-and-paste detection system[19]. Text-dependent speaker verification can be used to guard against replay assaults. Deep convolutional networks are a recent technology for detecting end-to-end replay threats [20]. Some replay attack detection techniques have been proposed by focusing on the network properties[21]. Replaying a voice recording of a target speaker. Far-field detection and cut-and-paste detection assaults are the two categories. The test segment in far-field detection replay attacks is a far-field microphone recording of the target that has been repeated on a phone handset with a loudspeaker[22]. If a recording is created by cutting and pasting small recordings to simulate the sentence required by a text-based system, it is referred to as a cut-and-paste detection system[23]. Text-dependent speaker verification can guard against replay assaults[24]. Deep convolutional networks are a recent technology for detecting end-to-end replay threats. Some replay attack detection techniques have been proposed by focusing on the network properties.

## 2.2. Speech Synthesis

Speech Synthesis can be artificially generating human speech with software or hardware programs. Speech synthesis may be used for various reasons, including text reading and serving as a personal AI assistant. Text-To-Speech is included in SS, which analyses the text and generates speech following the text supplied using the norms of linguistic description of the text. Another advantage of speech synthesis is providing multiple accents and voices instead of pre-recorded human voices. Lyrebird, a significant voice synthesis company, employs deep learning models to synthesize 1,000 phrases in a second. Text-To-Speech primarily relies on the quality of the speech corpus to build the system; regrettably, creating speech corpora is expensive. Another drawback of SS systems is that they do not detect periods or special characters[25]. The most common are homographs, which occur when two words have distinct meanings yet are spelled in the same way[26]. Char2Wav is a framework for speech synthesis production from start to finish. PixleCNN is also the foundation of WaveNet[27], an SS framework. WaveGlow[28] focuses on the second phase of text-to-speech synthesis systems, which typically involve two phases (encoder and decoder). As a result, WaveGlow is concerned with converting time-aligned information, such as a mel-spectrogram acquired from an encoder, into audio samples. Tacotron was first proposed in 2017[1]. Tacotron features CBHG, which comprises a 1-D convolution bank, a highway network, and a bidirectional GRU[29]. Tacotron 2[30] is made up of two parts. The first component is an attention-based recurrent sequence-to-sequence feature prediction network. This component's output is an anticipated series of mel spectrogram frames. Deep Voice3, a TTS framework, [31] includes three parts, encoder, decoder, and converter, a CNN-based Network. MelNet, a TTS framework [32], predicts a distribution element-by-element over the time and frequency dimensions of a spectrogram in an autoregressive manner. The network comprises many computational stacks that extract characteristics from various input bits. Then, these elements will be summed collectively to provide the overall context. The frequency-delayed stack's previous-layer outputs are the time-delayed and centralized stacks' current-layer outputs. The last layer of the frequency-delayed stack outputs is utilized to compute the audio-generating parameters. Using neural network TTS synthesis, it is possible to create speech sounds in the voices of numerous speakers, including those who have not been trained. This took only five seconds[33]. Char2Wav, end-to-end speech synthesis with a reader and a neural vocoder, was the first model to synthesize audio straight from text[34]. Deep Voice 1 was the first to use deep neural networks for text-to-speech in real-time, laying the groundwork for end-to-end neural speech synthesis[35], while Deep Voice 2[36] was able to replicate many voices using the same technology. Furthermore, most neural network-based speech synthesis models are auto-regressive, which means they condition audio samples on prior samples for long-term modeling and are straightforward to train and deploy[28]. Because the SS detection systems are also utilized for voice conversion detection, the detection methods for these two categories were examined in the VC summary section.

### 2.3. Voice conversion (VC) and Impersonation

The last type of audio deepfake is voice conversion, which takes the speech signal from the first speaker, the source, and alters it to seem like the second speaker uttered the target speaker. Impersonation is a type of VC in which one pretends to be someone else. Using improved technology, it is now faster to imitate, and one business called Overdub[1] can produce an imitation of any voice with one minute of sample audio. GANs may also be utilized for voice mimicry[37]. The joint density Gaussian mixture model with maximum likelihood parameter trajectory generation considering global variance is one of the essential foundations for VC[38]. This approach also serves as the foundation for the open-source Festvox system, which served as the primary VC toolkit in "The Voice Conversion Challenge 2016." Other voice conversion methods, such as neural networks and speaker interpolation, can also be used. However, given their versatility and high-quality outputs, GANs have recently been increasingly popular for VC. The Griffin Lim approach was utilized to rebuild time-domain data using a neural network architecture to imitate the voices of different genders. As a result, the model produced persuasive examples of impersonated speech. There are also other systems for detecting audio faking. ResNet, which was initially employed for image recognition, serves as the foundation of the audio spoofing detection system.

## 3. Dataset selection

We use the Fake or Real (FoR) Dataset in this research investigation [39]. It comprises over 195,000 audio samples. These audio samples were created by a computer utilizing the most up-to-date speech synthesis technologies. The dataset was built by integrating many datasets from various studies. The primary purpose of developing this dataset is to train algorithms for identifying phony speech [40, 41, 42]. The FoR dataset is divided into four subsets: (i) for-orig, (ii) for-norm, (iii) for-2sec, and (iv) for-rerec. The experiments in this work are confined to three datasets:: for-norm, for-2sec, and for-rerec.

### 3.1. For-Norm

The norm dataset is composed of 69400 audio, including duplicate audio signals. We removed the duplicate values from the original dataset and obtained 53868 unique audio samples. The dataset contains audio of different gender and target class (fake/real). The dataset can be preprocessed to remove duplicate values, set the sampling rate, and set the volume and number of channels.

### 3.2. For-2sec

The dataset contains a training set of 17720 audios and a testing set with 3731 audio samples. The dataset contains an audio length of 2 seconds. The dataset is completely balanced in terms of target class and gender. The sampling rate of audio samples of this dataset is 41000.

[1] https://www.descript.com/overdub

### 3.3. For-rerec

This dataset also contains an audio length of 2 seconds. The dataset contains 13268 audio samples with various genders and real and fake classes. The dataset has been trimmed to focus more on the signal and extract more insights from it. The sampling rate of audio samples is similar to the For-2sec audio dataset, which is 441000.

## 4. Proposed Approach

The proposed approach of this study is presented in Figure 1. We use Fake or Real (FoR) to perform experiments in this study. The dataset is divided into three subsets. In the first step, we analyze the dataset, perform exploratory analysis, and extract useful information for deepfake audio detection. Next, we convert the audio signal from the time domain to the frequency domain to analyze the audio graphically. Furthermore, the most important feature extraction techniques are utilized to extract the important features from an audio signal. The audio signal is normalized and reduces the dimension of the features so that only valuable features may be selected. We split the dataset into two sets; training and testing. 80% of the complete dataset is used to train the models, and the remaining 20% data is used to measure the performance of the models. We selected multiple machine learning (ML) models to detect fake or real audio from the datasets.

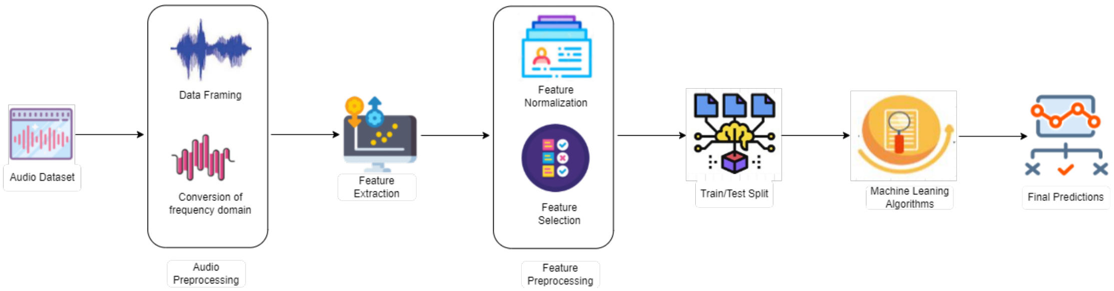

**Figure 1:** Proposed methodology to detect fake audio

### 4.1. Data Pre-Processing

Preprocessing is a necessary step in machine learning to obtain better results. The primary goal of audio data preprocessing is to transform the audio signal into a format that the machine learning model can easily interpret; that is why it is important to perform standard preprocessing steps before the model implementation. Data framing is a preprocessing of audio singles that involves converting audio signals into a format that a machine can understand. We remove the duplicate audio from the dataset and normalize the audio features in this step. We extract the value after every second from an audio signal called audio sampling and set the sampling rate of an audio signal is 44100. The sampling rate shows the frame rate value of an audio signal, and it is defined by Equation 1.

$$Frames = sampling\_rate \times time \tag{1}$$

In this scenario, we set the sampling rate of each signal in the dataset using data framing. The frequency-domain visualization of an audio signal is shown in the next step. As shown in Figure 2, we plot a Mel power spectrogram, which indicates the audio strength at certain time intervals and shows distinct frequency levels. It displays a Mel power spectrogram of a 2-second audio stream. The frequency is presented on the vertical axis of the spectrogram and has a range of 0-8KHz, while the time of the signal is indicated on the horizontal axis. The color combination in the spectrogram represents the signal's amplitude, while the signal's volume is indicated by decibels ($dB$). The standard scaler is used to normalize the characteristics. We utilized it to scale the characteristics into a particular range. The goal of Standard Scaler is to normalize characteristics such that they are fairly evenly distributed. As demonstrated in the equation 2, it reduces the variance to one and eliminates the mean of the data. Sc is our standardized version of "i".

$$Sc = \frac{(i - \mu)}{\sigma} \tag{2}$$

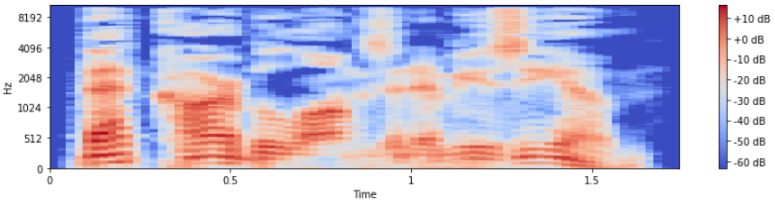

**Figure 2:** Mel Power Spectrogram Audio signal

## 4.2. Feature Extraction

Each audio in the FoR dataset contains useful characteristics we need to extract and feed to the ML model for better results. In this study, we applied various feature extraction approaches to obtain the features from the audio signal. In the first step, we converted the audio signal into machine-readable form, and then we applied MFCC, spectral_rolloff, spectral_centroid, spectral_contrast, spectral_bandwidth and zero_crossing_rate features extraction approaches to extract features from the signal. This results in an array, then the mean, median, and standard deviation is computed, yielding the final value of the feature for that specific audio signal.

We discovered that all 270 characteristics taken from a single audio file are unimportant and increase the classification time of ML models. Thus we used the feature reduction strategy. The 270 features from each audio are extracted using the sliding window technique. We employed Principal Component Analysis (PCA) to identify the most significant characteristics. We experimented with various numbers of n components, but in the end, we adjusted the PCA n component to 65 and got superior results. To evaluate the usefulness of selected features, we computed the explained_variance_ratio, which is 97%. It shows that this approach selected very suitable features. .Finally, 65 unique characteristics out of 270 are passed to ML classifiers to identify actual and fake audios.

### 4.3. Classification Models

This study performed experiments by extracting features from the audio signal Moreover, applying different ML classifiers to these audio features to detect real or fake audio. After pre-processing the audio sample, 270 features are retrieved using various feature extraction approaches. However, with just 2 seconds of audio and 270 features, the classification procedure takes a long time. We reduced the feature vector length, and 65 feasible features were fed to the classification models. For the classification, six ML classifiers, SVM [43], MLP [44], DT [45], LR [46], NB [47] and XGB [48] are implemented, and their performance was compared. We used the default parameters of these classifiers. These models' output determines whether a particular audio signal is authentic or fake.

## 5. Experimental Results and Comparative Analysis

This section briefly describes results obtained from various classifiers (SVM, MLP, DT, LR, XGB, and NB). We performed experiments using the Fake or Real (FoR) dataset and obtained promising results. Except for the baseline technique, this dataset was not used in any other research [49]. Existing research conducted experiments on the original dataset, but there is presently no research available that works on the multiple variants of this dataset, so we conducted experiments on the three subsets of the FoR dataset. The experiments involve six classifiers and three datasets (For-Norm, For-2sec, For-rerec). The results of this study using various datasets are presented in Table 1. We used accuracy evaluation metrics to evaluate the classification performance of six classifiers on datasets (For-Norm, For-2sec, For-rerec). The dataset was divided into two parts: training and testing. The first 80% of the whole dataset is utilized for training the models, while the remaining 20% is used to measure model performance. It is observed that all the ML models performed outstandingly on the given datasets. The SVM model outperformed other ML models with 97.57% and 98.83% accuracy using the For-2sec and For-rerec datasets. It can be observed that the XGB model obtained the highest accuracy score of 92.60% using the For-Norm dataset. It was discovered that the XGB model performed very well on average across all datasets, with an average accuracy score of 93.50 %. We also conducted a larger comparison with state-of-the-art research.

The baseline approach performed its experiments on the original dataset, while this study worked on various subsets of the FoR dataset. To our best knowledge, no prior work has been done on the subsets of the FoR dataset. The goal of generating these subsets from the original dataset is to improve classification while reducing computational costs. The proposed and baseline approaches' results are presented in table 2. The original dataset was divided into three subsets(For-Norm, For-2sec, For-rerec) for better classification. We highlighted the best average result of the proposed approach from three datasets generated using the XGB model. The accuracy score of the XGB model shows that the proposed approach outperforms the baseline approach with a 26 % accuracy gain. The highest accuracy of the baseline approach is obtained from SVM, with 67 % model accuracy. It is concluded that the suggested strategy outperforms the baseline approach in detecting real or fake audio from the FoR dataset. The best features aid the classification process in accurately identifying real or fake audio in a short period.

**Table 1**
Machine learning models Accuracy(%) from each dataset

| Models | for-2sec | for-norm | for-rerec |
|---|---|---|---|
| SVM | 97.57 | 71.54 | 98.83 |
| MLP | 94.69 | 86.82 | 98.79 |
| DT | 87.13 | 62.16 | 88.28 |
| LR | 89.92 | 82.80 | 88.31 |
| XGB | 94.52 | 92.60 | 93.40 |
| NB | 88.20 | 81.80 | 81.91 |

**Table 2**
Comparative Analysis of proposed approach with baseline approach results

| Reference | Algorithm | Test accuracy(%) |
|---|---|---|
| Algorithms implemented in [49] | SVM | 0.67 |
| **Proposed Approach Result** | **XGB** | **0.93** |

## 6. Conclusion and Future Work

Deepfakes have recently become a critical topic to be discussed. In manipulated audio-visual content, this type of data spreads hate and misinformation to the entire world. Therefore, it is very important to develop a system for the early detection of deepfake to prevent the spread of misinformation. This research focuses primarily on the task of detecting deepfake audio. An optimal feature engineering technique with machine learning approaches is used to classify real and fake audio. We employed Fake or Real (FoR) datasets and their various subsets in this study. The proposed approach is implemented on the subsets of FoR dataset. This study shows that the SVM model has the highest test accuracy using For-2sec and For-rerec datasets, while the XGM model exhibits the highest accuracy using the For-Norm dataset. The results show that the proposed approach obtained better results than the baseline approach. This work can be extended in the future by exploring the latest feature-extracting methodologies that will aid in obtaining better results with both machine learning and deep learning. Furthermore, the proposed system's architecture is simpler than the models used in paper [49] and achieves comparable accuracy. In addition, the deep learning model can exhibit better performance than machine learning because of the feedforward and feed-backward strategy. However, Deep learning models can also be used to implement amplitude-based classification. One limitation of this work is that deep learning with audio features has not been utilized. Furthermore, we can expand this work by using the original dataset and the subsets of the dataset and performing a deep learning approach with new feature extraction approaches.

## Acknowledgment

The research is supported by Zayed University grant number R21096.

## References

[1] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio, et al., Tacotron: A fully end-to-end text-to-speech synthesis model, arXiv preprint arXiv:1703.10135 164 (2017).

[2] S. Arik, J. Chen, K. Peng, W. Ping, Y. Zhou, Neural voice cloning with a few samples, Advances in Neural Information Processing Systems 31 (2018).

[3] Y. Zhou, S.-N. Lim, Joint audio-visual deepfake detection, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 14800–14809.

[4] A. Qais, A. Rastogi, A. Saxena, A. Rana, D. Sinha, Deepfake audio detection with neural networks using audio features, in: 2022 International Conference on Intelligent Controller and Computing for Smart Power (ICICCSP), IEEE, 2022, pp. 1–6.

[5] S. Agarwal, H. Farid, T. El-Gaaly, S.-N. Lim, Detecting deep-fake videos from appearance and behavior, in: 2020 IEEE International Workshop on Information Forensics and Security (WIFS), IEEE, 2020, pp. 1–6.

[6] G. Drakopoulos, I. Giannoukou, P. Mylonas, S. Sioutas, A graph neural network for assessing the affective coherence of twitter graphs, in: 2020 IEEE International Conference on Big Data (Big Data), IEEE, 2020, pp. 3618–3627.

[7] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, M. Nießner, Faceforensics++: Learning to detect manipulated facial images, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 1–11.

[8] N. M. Müller, K. Markert, K. Böttinger, Human perception of audio deepfakes, arXiv preprint arXiv:2107.09667 (2021).

[9] P. Korshunov, S. Marcel, Deepfake detection: humans vs. machines, arXiv preprint arXiv:2009.03155 (2020).

[10] M. Groh, Z. Epstein, C. Firestone, R. Picard, Deepfake detection by human crowds, machines, and machine-informed crowds, Proceedings of the National Academy of Sciences 119 (2022).

[11] A. R. Javed, F. Shahzad, S. ur Rehman, Y. B. Zikria, I. Razzak, Z. Jalil, G. Xu, Future smart cities requirements, emerging technologies, applications, challenges, and future aspects, Cities 129 (2022) 103794.

[12] A. Abbasi, A. R. Javed, F. Iqbal, Z. Jalil, T. R. Gadekallu, N. Kryvinska, Authorship identification using ensemble learning, Scientific Reports 12 (2022) 1–16.

[13] R. K. Das, T. Kinnunen, W.-C. Huang, Z. Ling, J. Yamagishi, Y. Zhao, X. Tian, T. Toda, Predictions of subjective ratings and spoofing assessments of voice conversion challenge 2020 submissions, arXiv preprint arXiv:2009.03554 (2020).

[14] Z. Wang, S. Cui, X. Kang, W. Sun, Z. Li, Densely connected convolutional network for audio spoofing detection, in: 2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), IEEE, 2020, pp. 1352–1360.

[15] B. Chettri, D. Stoller, V. Morfi, M. A. M. Ramírez, E. Benetos, B. L. Sturm, Ensemble models for spoofing detection in automatic speaker verification, Proc. Interspeech 2019 (2019) 1018–1022.

[16] M. Sahidullah, T. Kinnunen, C. Hanilçi, A comparison of features for synthetic speech detection (2015).

[17] N. M. Müller, F. Dieckmann, P. Czempin, R. Canals, K. Böttinger, J. Williams, Speech is silver, silence is golden: What do asvspoof-trained models really learn?, arXiv preprint arXiv:2106.12914 (2021).

[18] S. Pradhan, W. Sun, G. Baig, L. Qiu, Combating replay attacks against voice assistants, Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 3 (2019) 1–26.

[19] J. Villalba, E. Lleida, Preventing replay attacks on speaker verification systems, in: 2011 Carnahan Conference on Security Technology, IEEE, 2011, pp. 1–8.

[20] F. Tom, M. Jain, P. Dey, End-to-end audio replay attack detection using deep convolutional networks with attention., in: Interspeech, 2018, pp. 681–685.

[21] M. Witkowski, S. Kacprzak, P. Zelasko, K. Kowalczyk, J. Galka, Audio replay attack detection using high-frequency features., in: Interspeech, 2017, pp. 27–31.

[22] J. Gonzalez-Rodriguez, A. Escudero, D. de Benito-Gorrón, B. Labrador, J. Franco-Pedroso, An audio fingerprinting approach to replay attack detection on asvspoof 2017 challenge data., in: Odyssey, 2018, pp. 304–311.

[23] L. Huang, C.-M. Pun, Audio replay spoof attack detection by joint segment-based linear filter bank feature extraction and attention-enhanced densenet-bilstm network, IEEE/ACM Transactions on Audio, Speech, and Language Processing 28 (2020) 1813–1825.

[24] L. Huang, C.-M. Pun, Audio replay spoof attack detection using segment-based hybrid feature and densenet-lstm network, in: ICASSP 2019-2019 IEEE international conference on acoustics, speech and signal processing (ICASSP), IEEE, 2019, pp. 2567–2571.

[25] K. Kuligowska, P. Kisielewicz, A. Włodarz, Speech synthesis systems: disadvantages and limitations, Int J Res Eng Technol (UAE) 7 (2018) 234–239.

[26] G. Watson, Z. Khanjani, V. Janeja, Audio deepfake perceptions in college going populations, UMBC Faculty Collection (2021).

[27] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, K. Kavukcuoglu, Wavenet: Agenerative model for raw audio, arXiv preprint arXiv:1609.03499 (2016).

[28] R. Prenger, R. Valle, B. Catanzaro, Waveglow: A flow-based generative network for speech synthesis, in: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2019, pp. 3617–3621.

[29] J. Lee, K. Cho, T. Hofmann, Fully character-level neural machine translation without explicit segmentation, Transactions of the Association for Computational Linguistics 5 (2017) 365–378.

[30] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan, et al., Natural tts synthesis by conditioning wavenet on mel spectrogram predictions, in: 2018 IEEE international conference on acoustics, speech and signal processing (ICASSP), IEEE, 2018, pp. 4779–4783.

[31] W. Ping, K. Peng, A. Gibiansky, S. Ö. Arik, A. Kannan, S. Narang, J. Raiman, J. Miller, Deep voice 3: Scaling text-to-speech with convolutional sequence learning., in: ICLR (Poster), 2018.

[32] S. Vasquez, M. Lewis, Melnet: A generative model for audio in the frequency domain (2019).

[33] Y. Jia, Y. Zhang, R. J. Weiss, Q. W. J. S. F. Ren, Z. C. P. N. R. Pang, I. L. M. Y. Wu, Transfer

learning from speaker verification to multispeaker text-to-speech synthesis (2018).

[34] J. Sotelo, S. Mehri, K. Kumar, J. F. Santos, K. Kastner, A. Courville, Y. Bengio, Char2wav: End-to-end speech synthesis (2017).

[35] S. Ö. Arık, M. Chrzanowski, A. Coates, G. Diamos, A. Gibiansky, Y. Kang, X. Li, J. Miller, A. Ng, J. Raiman, et al., Deep voice: Real-time neural text-to-speech, in: International Conference on Machine Learning, PMLR, 2017, pp. 195–204.

[36] A. Gibiansky, S. Arik, G. Diamos, J. Miller, K. Peng, W. Ping, J. Raiman, Y. Zhou, Deep voice 2: Multi-speaker neural text-to-speech, Advances in neural information processing systems 30 (2017).

[37] Y. Gao, R. Singh, B. Raj, Voice impersonation using generative adversarial networks, in: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2018, pp. 2506–2510.

[38] T. Toda, A. W. Black, K. Tokuda, Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory, IEEE Transactions on Audio, Speech, and Language Processing 15 (2007) 2222–2235.

[39] R. Reimao, V. Tzerpos, For: A dataset for synthetic speech detection, in: 2019 International Conference on Speech Technology and Human-Computer Dialogue (SpeD), IEEE, 2019, pp. 1–10.

[40] K. Ito, L. Johnson, The lj speech dataset, https://keithito.com/LJ-Speech-Dataset/, 2017.

[41] J. Kominek, A. W. Black, The cmu arctic speech databases, in: Fifth ISCA workshop on speech synthesis, 2004.

[42] K. MacLean, Voxforge, Ken MacLean.[Online]. Available: http://www.voxforge.org/home.[Acedido em 2012] (2018).

[43] T. Evgeniou, M. Pontil, Support vector machines: Theory and applications, in: Advanced Course on Artificial Intelligence, Springer, 1999, pp. 249–257.

[44] A. Abbasi, A. R. Javed, A. Yasin, Z. Jalil, N. Kryvinska, U. Tariq, A large-scale benchmark dataset for anomaly detection and rare event classification for audio forensics, IEEE Access (2022).

[45] A. J. Myles, R. N. Feudale, Y. Liu, N. A. Woody, S. D. Brown, An introduction to decision tree modeling, Journal of Chemometrics: A Journal of the Chemometrics Society 18 (2004) 275–285.

[46] A. Abbasi, A. R. Javed, C. Chakraborty, J. Nebhen, W. Zehra, Z. Jalil, Elstream: An ensemble learning approach for concept drift detection in dynamic social big data stream learning, IEEE Access 9 (2021) 66408–66419.

[47] I. Rish, et al., An empirical study of the naive bayes classifier, in: IJCAI 2001 workshop on empirical methods in artificial intelligence, volume 3, 2001, pp. 41–46.

[48] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining, 2016, pp. 785–794.

[49] J. Khochare, C. Joshi, B. Yenarkar, S. Suratkar, F. Kazi, A deep learning framework for audio deepfake detection, Arabian Journal for Science and Engineering (2021) 1–12.