Voice Impersonation Detection Using LSTM based RNN and Explainable AI

Abdur Rahim
Department of Computer Science and Engineering
Brac University
Dhaka-1212, Bangladesh
abdur.rahim@g.bracu.ac.bd

Kawshik Barua
Department of Computer Science and Engineering
Brac University
Dhaka-1212, Bangladesh
kawshik.barua@g.bracu.ac.bd

Prantozit Saha Parizat
Department of Computer Science and Engineering
Brac University
Dhaka-1212, Bangladesh
prantozit.saha.parizat@g.bracu.ac.bd

Md.Asad Uzzaman Noor
Department of Computer Science and Engineering
Brac University
Dhaka-1212, Bangladesh
md.asad.uzzaman.noor@g.bracu.ac.bd

Miftahul Jannah
Department of Computer Science and Engineering
Brac University
Dhaka-1212, Bangladesh
miftah.bracu@gmail.com

Md.Golam Rabiul Alam,PhD
Department of Computer Science and Engineering
Brac University
Dhaka-1212, Bangladesh
rabiul.alam@bracu.ac.bd

***Abstract*—The advancing field of artificial synthetic media introduced deepfakes which made it easier to synthesize a person's voice, identical to their original voice mechanically to use it for negative means. People's voices are exposed to public as it is a proficient and more convenient media of exchanging information over various mediums, entertainment, speech delivering, news reading and so on, making it easier to collect voice samples for creating fake yet almost identical voice samples to trick people. So it has become vital to prevent this crime which led us to work on this research paper with intention to help the victims of voice impersonation attacks. Here we used LSTM based RNN model in order to distinguished between real and synthesize voice.Furthermore, to compare the results we got from the mentioned process, we build a SVM classifier and finally we've explained the predicted outputs(fake or real) of both LSTM and SVM model by using an Explainable AI method named LIME. Our research resulted in 98.33% accuracy rate through our proposed LSTM model and very low percentage of error in detecting fake/synthesized voices.**

***Index Terms*—Deep-fake, Voice Impersonation Detection, LSTM based RNN, SVM, LIME, Explainable AI**

I. Introduction

Voice impersonation detection is a system where fake voice features are identified in order to prevent any unwanted circumstances. Earlier impersonating someone's voice was limited in the field of entertainment where an impressionist or a mimic artist used to impersonate celebrities or famous people to make a funny act in order to entertain audience. But after deepfake technology was developed, more apps softwares are being built to make fake voice presentations that promotes a type of entertainment, especially over social media. Apparently it may project the idea of entertainment but looking at it from another dimension, it can be noticed that, voice impersonation may also be used for harmful purposes such as fraudulent, security breach, defaming famous personalities and so on. With the thriving improvement of technology and availability of smart devices, voice based applications and software are becoming more common. The use of voice based biometric systems such as unlocking a device or a door, smart home assistance (Google Home, Amazon Alexa), access control of a device and security control are becoming more popular gradually. Additionally, in recent times, voice based texting system is introduced over many social media platforms such as Facebook messenger, Whatsapp, WeChat and many more. Undoubtedly it has made communication and exchange of information less time consuming and easier for both educated and uneducated people. With all these voice based improved technology; we cannot ignore the underlying fact that our voices are getting easily exposed to the public more easily than ever. Unlike other biometric systems (eg. fingerprint scanning, facial recognition, iris recognition etc). voice recognition systems are easier to breach since it is easier to collect a person's voice data. So identifying fake voices has become more vital, which led us think about working on this topic since there have been very few researches available about it. In this work, we have introduced a structured model which will be able to differentiate between synthesized voice and original voice by analyzing and comparing the given voice feature data. At first, we collected original voice clip and the fake voice clip in order to create our database. Secondly, we did feature extraction of the data to convert the audio files into feature data in order to make it trainable for machine learning deep learning models. Feature extracting included following features: MFCC, STFT, CQT, CENS, ZCR, RMSE, Spectral centroid, Spectral roll off. Then after doing data processing, the extracted data was used to train our LSTM base RNN model and SVM classifier for comparing the result which lead us to our final step where we used an Explainable AI method called LIME for explaining the predictions.

II. Related Work

This section will concentrate on some relevant efforts in the subject of synthetic voice detection.

DNN-based speech synthesis algorithms can synthesize fake voice with naturalness by using several models such as Boltzmann machines [1], deep belief networks [2], mixed density networks [3], and Bidirectional LSTM [4] etc. These models are proposed for synthesizing high-quality and natural speech. Using these models recently many advance voice synthesis technologies has been developed. DeepMind WaveNet [5] in 2016 and Google Tacotron [6] in 2017 are two landmarks in voice synthesis. These two models use considerably advance DNN based speech synthesis, allowing for large-scale commercial applications for developing TTS and VC systems. Because of the strong capabilities of WaveNet and Tacotron, commercial systems like as Baidu TTS , Amazon AWS Polly [7], and Google Cloud TTS [8] have been created.

In order to detect these advanced synthetic voices, A variety of researches regarding the identification of real and fake voices are accessible in the literature.A Deep Learning based solution on Synthetic Speech Detection was proposed in the research work [9]. The aim of the work was to build a model which could extract the dynamic features of voice and which will be able to differentiate between artificially generated voice and real human speech. To do so they have used a dataset from APTLY Lab named Fake-Or Real dataset of size 4GB which was freely available. However, the model was built on Convolution Neural Networks (CNN) and Recurrent Neural Network (RNN). CNN helps to learn positional relation between pixels and it enables neural networks to identify shapes and pattern.In addition to this, they have used three different approaches named Short-Term-Fourier Transformation (STFT), Mel-Frequency Cepstral Coefficients (MFCC) and Mel-Spectrogram on the audio file for extracting features. Finally, the loss function and accuracy were used to assess the proposed model's performance during and after training. The loss value indicates how poorly or well a model performs after each iterations of optimization and it's a measure of how accurate the model's prediction compared to the real data. The model's accuracy was 94 percent while the loss value was 0.691 percent.

Paul et al. suggested a collection of short-term spectral characteristics in mid-2017 that can significantly enhance the accuracy of synthetic speech identification [10]. The authors give a detailed examination of the differences between synthetic and natural speech and discover intriguing patterns, such as the fact that lower frequencies (1kHz) and higher frequencies (greater than 7khz) are the most useful frequencies for discrimination between synthetic and real speech. Yu et al. released a study on the usage of Deep Neural Networks (DNNs) for voice spoofing detection as the popularity of DNN solutions grew [11]. The fundamental concept is to utilize DNNs to extract dynamic acoustic characteristics and identify an utterance as genuine or faked. According to the findings,the suggested technique outperforms standard static feature analysis using GMM classifiers.

Mimicry voice can be detected using convolutional neural networks according to [11]. In the research work they focused on Automatic speaker verification system and shows how spoofing attacks are occurred on this ASV system. However, the main purpose of this research was to detect this kind of mimicry voice using CNN. The main problem of ASV biometric system is that it's powerless against mocking assaults. There is no impersonation (mimicry) sample available publically so for mimicry detection purpose they have created some mimicry samples generated by professional mimicry artist and some samples which are collected from the celebrity speech available on internet. Additionally, creating a high-quality mimic database in not an easy work and for this reason speech samples were collected at 8 kHz and for doing this they have selected two Telugu-speaking celebrities. The celebrity's speeches are recorded as short frame, each with a maximum duration of 3 sec. A total 68 samples were collected from the speeches of two celebrities and mimic artist. Furthermore, imitators who want to mimic the target voice adopt prosodic features which include intonation, stress and rhythm of the target speaker and for this reason spectral features such as MFCC were selected. Apart from this, for evaluating the result a classifier based on CNN model was established. This model is successfully adopted from the detection of speech presentation attacks and cost optimization analysis was performed. After doing all the above steps they got a testing accuracy of 0.6693 or 67% and the equal error rate was 0.3585.

Previous researches [12] [13] demonstrates that dynamic acoustic characteristics (such as dynamic filter banks, dynamic MFCCs, and dynamic linear prediction cepstral coefficients) are better candidates for spoofing detection than standard static features (such as magnitude-based features and cosine normalized phase features). Based on this, as well as the fact that DNNs are widely recognized for their ability to extract dynamic characteristics, the researchers opted to use a 5-layer deep neural network to conduct classification on the AVSpeech2015 dataset. The experiment findings demonstrate that DNNs with dynamic features outperform earlier approaches based on static features and GMM models. Zhang et al. released a paper in mid-2017 on their research into deep-learning frameworks for speaker verification anti-spoofing [14]. The authors propose using CNNs in combination with RNNs to recognize synthetic speech in their study. The suggested technique shows the state-of-the-art performance for an end-to-end single system using the ASVSpoof2015 dataset as a baseline. Xiaohai Tian and Xiong Xiao presented a study on faked speech identification using temporal convolutional networks [15]. Instead of utilizing handmade feature extractors with standard machine learning techniques, they propose using a single convolutional neural network to categorize an utterance.

As we can see, several deep learning-based works in the field of voice detection have been done earlier, but none of the works have taken the vanishing gradient problem of voice input into account. As audio input is obtained in time domain, part of the input information may be lost due to the vanishing gradient problem, which can have a significant impact on the output. The LSTM-based RNN is an excellent model for tackling this problem, and we utilized it in our study to obtain high accuracy in recognizing false and real voices. Furthermore, to the best of our knowledge, no similar work in the related field has employed Explainable AI technique to explain the predictions made by the models. However, in our study, we employed an Explainable AI called LIME to explain the predictions generated by our LSTM-based model, which will be able to assist us in understanding how LSTM works while making the predictions.

III. Methodology

A. Proposed Approach

We proposed a model based on LSTM based RNN which will be able to differentiate between real and fake voice.For this purpose first of all we have collected our dataset named FoR dataset which consists of total 13956 audio files.As all the files were audio file so we have extracted 8 important features like Mel-Frequency Cepstral Coefficients (MFCC), Chroma Short Time Fourier Transformation (STFT), Chroma Constant Q Transform (CQT),Chroma Energy Normalized Statistics(CENS),Spectral Centroid, Spectral Roll Off, Zero Crossing Rate(ZCR), Root Mean Square Energy (RMSE).Then for data preprocessing purpose we have used Standard Scaler for scaling the data in range and Recursive Feature Elimination (RFE) to keep the most important or relevant features in the dataset and then train the dataset into our model to get the desired result. A model based on SVM was also build in order to compare it's result with our proposed model.In addition, we explained the predicted output we obtained from our LSTM model and from the SVM model using an Explainable AI method called LIME. The following diagram is a top level layout for our working strategy:

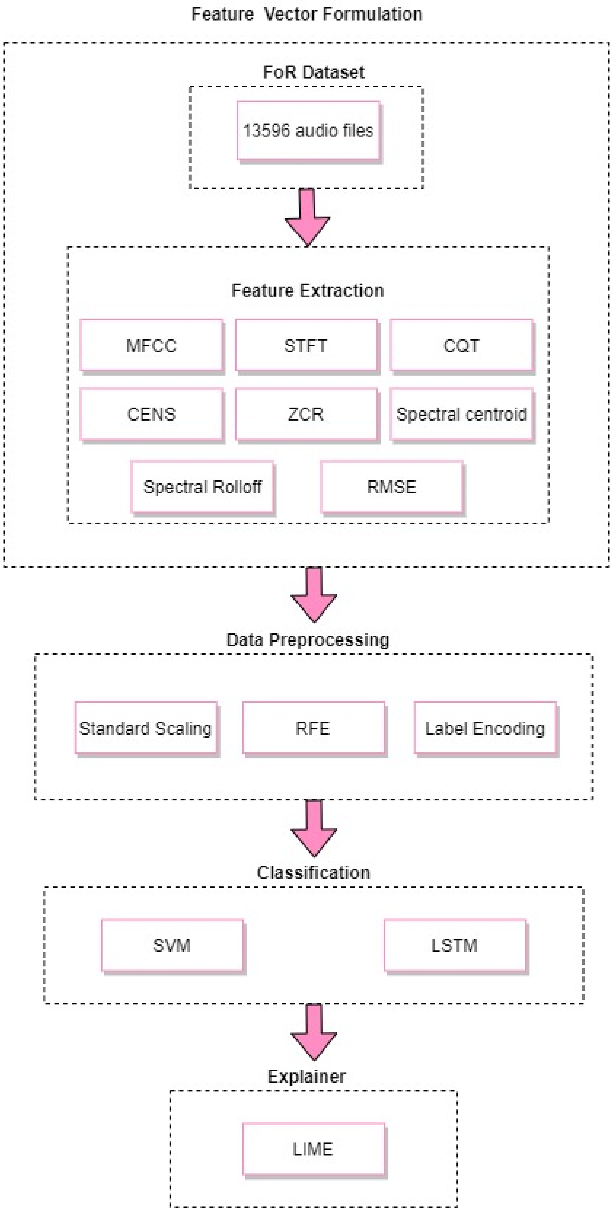


Fig. 1. Top level layout of the voice impersonation detection model

B. Dataset

For training our proposed models to classify between fake and real voice, we needed a dataset of raw audio files containing both real human voice samples from various sources along with AI synthesized fake voice samples using latest techniques. Very few public datasets such as The “AsvSpoof” dataset [16], the “Anti-Spoofing for Text-Independent Speaker Verification: AnInitial Database” [17] the “FoR (Fake or Real)” [18] dataset are available in this regard. Among these we chose the FoR dataset created by Aptly lab as it contains more number of utterances and also because the fake utterances are synthesized with the latest advanced technology and voice synthesis products such as Amazon AWS Polly, Google Cloud TTS and Microsoft Azure TTS etc, which results in naturalness similar to a real person’s voice. Real voices in FoR dataset are gathered from publically available speech related datasets and free videos/audios on internet mediums like Facebook,Youtube,Twitter,Ted Talks which cover a great variation of speaker age,sex,accent,tone etc. The FoR dataset is available in different versions publically . In our case we used the “for-2sec” version which has files truncated at 2 seconds.However,This 2-sec version of FoR dataset consists of a total 13,956 raw audio file and some of them are real voice and some of them are fake.

Since LSTM works for sequential data, so window sliding the two seconds audio file and merging all the necessary extracted features for each slide can make the data sequential. Hence this would also make the dataset suitable for the LSTM model. However, we had 265 columns initially but if the audio file undergoes window sliding for 20 milliseconds we would have 6 slides for each audio files. Hence, each row in our dataset will have datas of all the slides in a single row.

C. Feature Extraction

As discussed in the previous section our dataset contains audio files with extension like .mp3, .wav etc. But these data’s are provided as a form of audio files which cannot be understood by the deep learning models directly. So in order to establish our model and train data we need to process audio files and convert them into data features so that deep learning machine learning models could use them .Feature extraction is one such process to convert audio files into an understandable format.In audio analysis process, an initial crucial step is to extract features from the audio files. The goal is to pull out a collection of features from the initial audio files containing dataset. These features extracted should be informative in respect to the qualities which defines an audio signal.The process of feature extraction may also be interpreted as a conversion technique because while doing feature extraction while we are getting tabular data from an audio signal which is in time frequency domain.

In this work, we have extracted eight features of audio files that will help our proposed model to differentiate between real and fake synthetic voice. We used python based Librosa library for analyzing and extracting features.These eight features and their extraction process using Librosa is discussed below:

- **Mel-Frequency Cepstral Coefficients (MFCC):** A signal’s Mel frequency cepstral coefficients (MFCCs) are generally a collection of characteristics (typically 10-20) that succinctly characterizes the all-inclusive form or shape of a spectrum. MFCCs are usually calculated by taking a signal’s Fourier transform and then mapping the spectrum’s powers onto the scale named mel scale .The mel scale is a perceptual scale of sounds that listeners consider to be equally spaced apart. A well-known formula for converting f hertz to m mels is:

$$\mathrm{Mel}(h) = 2595 \log\left(1 + \frac{f}{700}\right) \tag{1}$$

then the logs of the powers at each of the mel frequencies are taken and converted by discrete cosine transform. The amplitudes of the resulting spectrum is called MFCC.

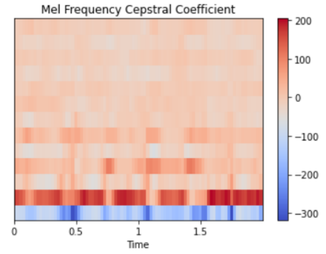
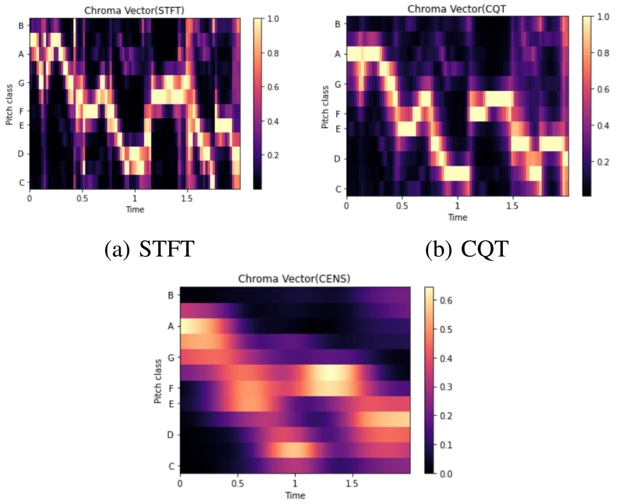


Fig. 2. MFCC waveplot for single audio file

- **Chroma Vector:** Chroma-based characteristics, which are also familiar as ”pitch class profiles,” are a strong method for evaluating music with usefully classified pitches into twelve groups. The chroma vector is basically a twelve(12) element spectral energy representation in which the bins reflect the 12 equally toned down pitch classes of western-style music. .Chroma_stft (Short Time Fourier Transform) is used for extracting chroma features with time intervals between the frequencies, While Chroma_cqt (Constant Q Transform) extracts features with geometrically placed frequency axis. Chroma energy normalized statistics (CENS) is statistics of energy distribution between the energy bands.



(a) STFT

(b) CQT

(c) CENS

Fig. 3. Chroma vector wave plots for single audio sample

- **Zero-Crossing Rate (ZCR):**An audio frame's Zero-Crossing Rate (ZCR) is basically the value of rate at which the signal's sign changes during the frame. To put it another way, it's the sum of number at which the signal's value shifts from positive side to negative side of an axis and vice versa. The ZCR is calculated using the following equation (2). Here s represents signal of length L. if $s(t) = 1$ if the signal has a positive amplitude at time t or 0 in the other case.

$$ZCR = \frac{1}{T}\sum_{t=1}^{T} |s(t) - s(t-1)| \qquad (2)$$

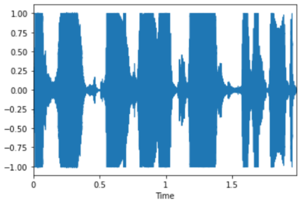


Fig. 4. ZCR wave plots for single audio sample

- **Spectral Centroid:** Spectral Centroid is determined as the mean weight of the frequencies contained in the sound and shows where the sound's "center of mass" is located. If the frequencies in music remain consistent throughout, the spectral centroid will be towards the middle, and if the sound ends with high frequencies, the centroid will be near the end.

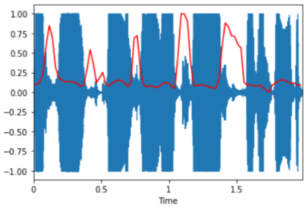


Fig. 5. Spectral Centroid wave plots for single audio sample

- **Spectral Roll off:**Spectral roll off is the frequency below which a specified percentage of the total spectral energy, e.g. 85%, lies. The spectral roll off point is the fraction of bins in the power spectrum at which 85% of the power is at lower frequencies. That is, the roll-off is the frequency below which 85% of accumulated spectral magnitude is concentrated.

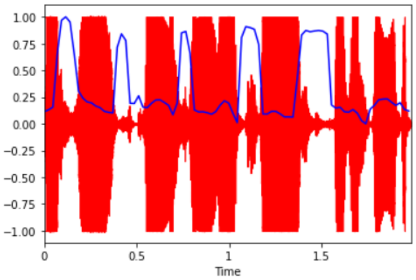


Fig. 6. Spectral Roll off wave plots for single audio sample

- **The Root-Mean-Square Energy (RMSE):**Root Mean Square (RMS) value is the most important parameter that signifies the size of a signal.RMS energy is computed for each frame of the audio samples.The energy of a signal corresponds to the total magnitude of the signal. For audio signals, that roughly corresponds to how loud the signal is.

We've extracted all these features along with their Mean,Median, Maximum, Minimum and Standard Deviation value. We also counted all the coefficients of MFCC and Chroma based features .Thus our csv data set had 266 columns or features including the label column.

D. *Classifications*

LSTM: LSTM stands for long short-term memory networks, used in the field of Deep Learning. It is a variety of recurrent neural networks (RNNs) that are capable of learning long-term dependencies, especially in sequence prediction problems. RNNs using LSTM units is explicitly designed to avoid the vanishing gradient problem which occurs in normal RNN . Remembering information for long periods of time is a default behavior in LSTM. A common LSTM unit is composed of a cell, an input gate, an output gate and a forget gate. Cell states have the ability to remove or add information to the cell which is carefully regulated by structures called gates. Gates are a way to optionally let information through so that LSTM can overcome the vanishing gradient problem. We will use this LSTM based RNN Model mainly to perform voice impersonation detection task as voice input is a time series data and by using LSTM we want to ensure that no input sequence is lost so that our classification method becomes accurate as possible.

SVM: The support vector machine or SVM is a popular supervised learning technique which is used to solve both regression and classification problems. SVM generates a hyperplane in an iterative manner in order to divide the data-point. The main objective of SVM is to separate the given dataset in the best possible ways and select a hyperplane with the maximum possible margin between support vectors in the given dataset.However,to handle nonlinear input spaces it use kernel tricks.An input data space is transformed into the appropriate form using a kernel. We will use SVM in order to compare it's accuracy against our LSTM based model.

LIME: Local Interpretable Model-Agnostic Explanations or LIME can explain the predictions of any classifier model, by approximating it locally with an interpretable model which eventually tells us which feature or input is most useful while predicting the output.However,it's model-agnostic, meaning that it can be applied to any machine learning model. The technique attempts to understand the model by perturbing the input of data samples and understanding how the predictions change. Lime will be used in our work to explain how our models are making the predictions to show which inputs are more important while predicting.

IV. Result Analysis

A. Performance Metrics

After preprocessing the collected data and train it to our proposed model, now we need to evaluate the performance of our proposed model. In this purpose we have calculate the accuracy score,F-1 score, sensitivity score, specificity score and also plotted the confusion matrix for each of the classification model.

A confusion matrix is a simple and straightforward approach to illustrate a classifier's prediction result. The matrix compares the actual target values to the machine learning model's prediction which in result gives us a proper idea about how the model is working. However, the four basic things of a confusion matrix are True Positive, True Negative, False Positive and False Negative values.

True Positive: This is the case when the predicted value matches with the actual value and both of the values are positive. In our case it means that our proposed model predicts the sample voice as a real voice and the sample voice is actually real.

True Negative: In this case the predicted value also matches with the actual value and both the values are negative. This means that our model is predicted the voice sample as a fake voice and the sample voice is actually fake.

False Positive: Actual value doesn't matches with the predicted values. This is the case when the model predicted the voice sample as a real voice but the voice is actually a fake voice.

False Negative: This is the case when the model predicted the sample voice as a fake voice but the sample voice is actually real.

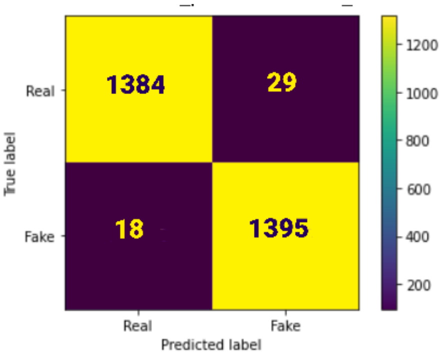


Fig. 7. Confusion matrix of our proposed LSTM model

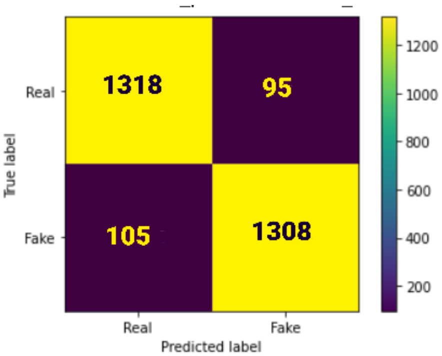


Fig. 8. Confusion matrix of SVM model

The F-score, also known as the F-1 score is a measure for how accurate a model is on a given dataset. However, it's a way of combining the precision and recall of the model and it's defined as the harmonic mean of the model's precision and recall. Here Precision is the number of True Positive divided by the number of True Positive and Number of False Positive and Recall is the number of True Positives divided by the number of True Positives and the number of False Negatives. On the other hand, Sensitivity refers to the measure of the proportion of actual positive cases that got predicted as positive (True Positive) and Specificity refers to the proportion of actual negative which got predicted as Negative (True Negative). Table 1 shows these scores in details: From the table it's clear that our proposed LSTM

Scores	**Classifiers**	
	SVM	**LSTM**
Accuracy	92.92%	98.33%
F-1	92%	98%
Sensitivity	92%	98%
Specificity	93%	99%

TABLE I
Calculated Scores for LSTM and SVM

based model have better score in every sectors than SVM classifier.

A ROC curve (receiver operating characteristic curve) is a graph that shows the classification model‘s performance over all categorization threshold.To compute the points in an ROC curve, an efficient, sorting-based algorithm named AUC (Area under the ROC curve) is used. It measures the entire two-dimensional area underneath the entire ROC curve. AUC has a value ranging from 0 to 1. The AUC model of a model whose predictions are 100% incorrect is 0, whereas the AUC of a model whose predictions are 100 percent accurate is 1.0.

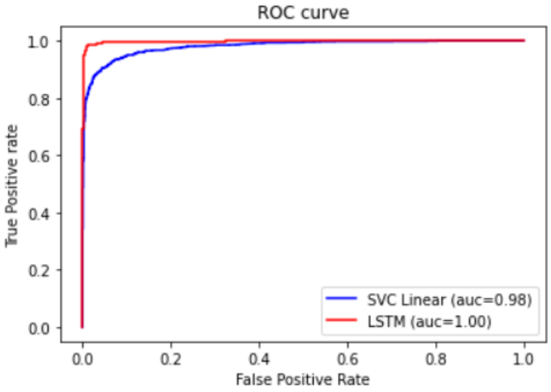
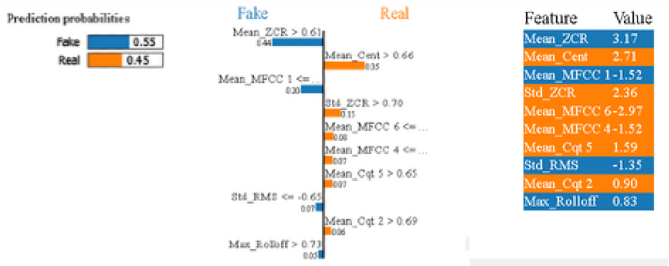


Fig. 9. ROC curve for LSTM and SVM

We have also explained the predicted output of the classification models using a technique of Explainable AI called Lime. Local model interpretability is provided by Lime. It modifies a single data sample by adjusting feature values and observes the resulting impact on the output.The output of LIME is a list of explanations, reflecting the contribution of each feature to the prediction of a data sample.

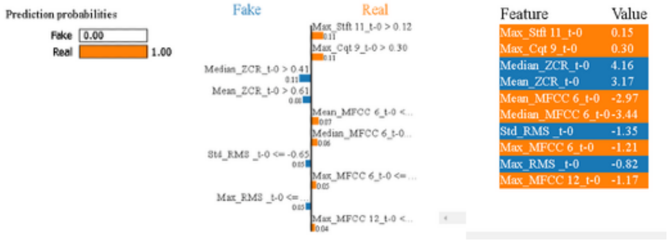


Feature	Value
Mean_ZCR	3.17
Mean_Cent	2.71
Mean_MFCC 1	-1.52
Std_ZCR	2.36
Mean_MFCC 6	-2.97
Mean_MFCC 4	-1.52
Mean_Cqt 5	1.59
Std_RMS	-1.35
Mean_Cqt 2	0.90
Max_Rolloff	0.83

Fig. 10. Explaining LSTM model’s prediction result using LIME

Figure 10 shows lime explanation for LSTM based RNN model. Note that the sample voice was a real voice and lime is showing based on which features and how much contribution each features have behind predicting the result.

Figure 11 shows the explanation behind predicting the output for SVM model. The contribution of each features behind predicting the output of SVM is shown in the figure.



Feature	Value
Max_Stft 11_t-0	0.15
Max_Cqt 9_t-0	0.30
Median_ZCR_t-0	4.16
Mean_ZCR_t-0	3.17
Mean_MFCC 6_t-0	-2.97
Median_MFCC 6_t-0	-3.44
Std_RMS _t-0	-1.35
Max_MFCC 6_t-0	-1.21
Max_RMS _t-0	-0.82
Max_MFCC 12_t-0	-1.17

Fig. 11. Explaining SVM model’s prediction result using LIME

B. *Comparative Analysis*

We try to compare our study to other similar studies in this area. The accuracy scores were used to perform the comparison investigation(TABLE II).We used the publically available FoR dataset as the foundation for our learning model. We have found some model trained based on this dataset [18] and also found some reasearch related to voice detection based on other datasets. [11] [9]Several of these studies used various preprocessing and extraction methods. We try to enhance all extraction and pre-processing methods in order to achieve the maximum level of accuracy in our analysis.The table demonstrates that the techniques in our suggested model approach are more efficient and produce better outcomes than any previous study.

TABLE II
ACCURACY COMPARISON BETWEEN DIFFERENT CLASSIFIERS

Existing Work	Classifiers	Dataset	Accuracy Score	Explainability
Recardo et al [18]	Naive Bayes	FoR	67.27%	No
Recardo et al [18]	SVM	FoR	73.46%	No
Recardo et al [18]	Decision Tree(J48)	FoR	70.26%	No
Recardo et al [18]	Random Forest	FoR	71.47%	No
Neelima et al [11]	CNN Based	Own generated	67%	No
Wijethunga et al [9]	CNN and RNN Based	Fake Or Real	94%	No
Proposed LSTM based model	LSTM Based	FoR	98.33%	Yes

V. Conclusion

Voice detection technology is capable of detecting between fake or real voices that are made from advanced voice synthesis technologies. As advanced AI-synthesized techniques are capable of producing extremely realistically sounding voices,, it also raises security and privacy concerns to everyone. So, we proposed a model which can detect real and fake voices by using LSTM based RNN model and Explainable AI method. In this paper, we have listed and examined various characteristics for impersonated voice identification. We have provided an overview of known techniques with the goal of categorizing features by speech properties that are used. For our research we used FoR dataset in which we got raw audio files containing both real human voice samples from various sources and also AI synthesized fake voices using latest techniques. In order to establish our model and train data, we need to extract features from the audio files amp; convert them, so that our proposed model could use them for training and testing purpose. After extracting features from voice samples,we combined features from various speech properties to get more accuracy to detect fake or real voices. Our analyses showed that earlier researchers did not take the vanishing gradient problem of voice input into account.The LSTM can erase, write and read data from the cell and also the LSTM architecture makes it easier for the RNN to preserve data across multiple timesteps. As the input audio files are obtained in time domain, so our proposed model based on LSTM based RNN model can easily tracking every bit of information from voice inputs to the cell. Thus, we are able to solve the vanishing gradient problem. We will also build a SVM classifier to compare the result we got from our proposed model. The SVM classifier showed less accuracy than our proposed LSTM based model. We calculated the F-1 score of both models which can measure the accuracy of a model on a dataset. The F-1 score showed promising percentage of our proposed model. From which we can understand that our proposed model can successfully differentiate between fake or real voices. We also calculated the accuracy,sensitivity and specificity score where in every score the percentage is high in LSTM than the SVM classifier. Moreover, it would seem logical that knowing the predicted output of specific predictions would help us decide whether to trust or distrust the prediction, or the classifier as a whole. So, we explained our predicted output using Explainable AI method called Lime which can adjust important feature values from every individual data and explains the contributions of each features. The FoR dataset have used in various voice synthesis model [TABLE], but the accuracy we got from our proposed model is 98.33% which is way higher than the previous models. We basically concentrated on identifying vocal imitation throughout the study.The results of our research suggest that our analysis can be used in a larger scale to prevent the crimes by misusing impersonated voice.

References

[1] Z.-H. Ling, L. Deng, and D. Yu, "Modeling spectral envelopes using restricted boltzmann machines and deep belief networks for statistical parametric speech synthesis," *IEEE transactions on audio, speech, and language processing*, vol. 21, no. 10, pp. 2129–2139, 2013.

[2] S. Kang, X. Qian, and H. Meng, "Multi-distribution deep belief network for speech synthesis," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 8012–8016.

[3] C. M. Bishop, "Mixture density networks," 1994.

[4] R. Li, Z. Wu, X. Liu, H. Meng, and L. Cai, "Multi-task learning of structured output layer bidirectional lstms for speech synthesis," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 5510–5514.

[5] 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.

[6] 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: Towards end-to-end speech synthesis," *arXiv preprint arXiv:1703.10135*, 2017.

[7] "Amazon corporation. 2020. amazon aws polly," 2020. [Online]. Available: https://aws.amazon.com/polly/

[8] "Text-to-speech: Lifelike speech synthesis — google cloud," 2020. [Online]. Available: https://cloud.google.com/text-to-speech

[9] R. Wijethunga, D. Matheesha, A. Al Noman, K. De Silva, M. Tissera, and L. Rupasinghe, "Deepfake audio detection: A deep learning based solution for group conversations," in *2020 2nd International Conference on Advancements in Computing (ICAC)*, vol. 1. IEEE, 2020, pp. 192–197.

[10] D. Paul, M. Pal, and G. Saha, "Spectral features for synthetic speech detection," *IEEE journal of selected topics in signal processing*, vol. 11, no. 4, pp. 605–617, 2017.

[11] M. Neelima and I. Santiprabha, "Mimicry voice detection using convolutional neural networks," in *2020 International Conference on Smart Electronics and Communication (ICOSEC)*. IEEE, 2020, pp. 314–318.

[12] M. Sahidullah, T. Kinnunen, and C. Hanilçi, "A comparison of features for synthetic speech detection," 2015.

[13] H. Yu, A. Sarkar, D. A. L. Thomsen, Z.-H. Tan, Z. Ma, and J. Guo, "Effect of multi-condition training and speech enhancement methods on spoofing detection," in *2016 First International Workshop on Sensing, Processing and Learning for Intelligent Machines (SPLINE)*. IEEE, 2016, pp. 1–5.

[14] C. Zhang, C. Yu, and J. H. Hansen, "An investigation of deep-learning frameworks for speaker verification antispoofing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 4, pp. 684–694, 2017.

[15] X. Tian, X. Xiao, E. S. Chng, and H. Li, "Spoofing speech detection using temporal convolutional neural network," in *2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*. IEEE, 2016, pp. 1–6.

[16] Z. Wu, T. Kinnunen, N. Evans, J. Yamagishi, C. Hanilçi, M. Sahidullah, and A. Sizov, "Asvspoof 2015: the first automatic speaker verification spoofing and countermeasures challenge," in *Sixteenth annual conference of the international speech communication association*, 2015.

[17] Z. Wu, P. L. De Leon, C. Demiroglu, A. Khodabakhsh, S. King, Z.-H. Ling, D. Saito, B. Stewart, T. Toda, M. Wester *et al.*, "Anti-spoofing for text-independent speaker verification: An initial database, comparison of countermeasures, and human performance," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 4, pp. 768–783, 2016.

[18] Ricardo and Vassilios, "For: A dataset for synthetic speech detection," 2020. [Online]. Available: http://bil.eecs.yorku.ca/publications/