HIGH SCHOOL EDITION
Journal of Student Research
Volume 14 Issue 1 (2025)

Voice Spoofing Detection Using Long Short-Term Memory Models with Mel-Spectrogram Analysis

Ian Baek[1] and Sojung Min[#]

[1]Korean Minjok Leadership Academy, Republic of Korea
[#]Advisor

ABSTRACT

Voice spoofing prevention has become a primary concern due to the increasing trend of frauds and scams using machine learning technology. To address this issue, I propose a novel voice spoofing detection system that classifies voices with accuracy, determining whether they are genuine or artificially generated. This system uses the fourier transform to convert the voice input into a mel-spectrogram, which is then processed using convolutional neural networks and long short term memory networks for classification. Comprehensive experiments demonstrate that the proposed method effectively classifies input signals with significant accuracy. In addition, I conducted frequency masking experiments to study how specific frequency bandwidths are correlated with enhancing real-fake classification. I expect that this method will inspire further innovations to prevent voice spoofing attacks.

Introduction

Voice fraud, commonly known as voice spoofing, involves the use of advanced technology to manipulate or fabricate voice recordings to deceive individuals or systems. Fraudsters can imitate someone's voice to authorize transactions, extract sensitive information, or conduct scams. The rise of AI (Artificial Intelligence) technology, particularly generative models, has significantly amplified the capabilities of these malicious actors. AI-generated voices can sound remarkably like the person being imitated which makes it challenging to distinguish between genuine and fake voices (Ergünay et al. 2015). These AI systems can create high-quality, realistic voice replicas that can easily bypass conventional detection mechanisms.

However, there is a noticeable discrepancy between real and fake voices in high-frequency ranges, which can be exploited for detection (Cohen et al. 2022). Real human voices typically exhibit natural variations and subtle nuances in high frequencies that are challenging for AI-generated voices to replicate accurately. By analyzing these high-frequency components, it is possible to distinguish between genuine and synthetic voices.

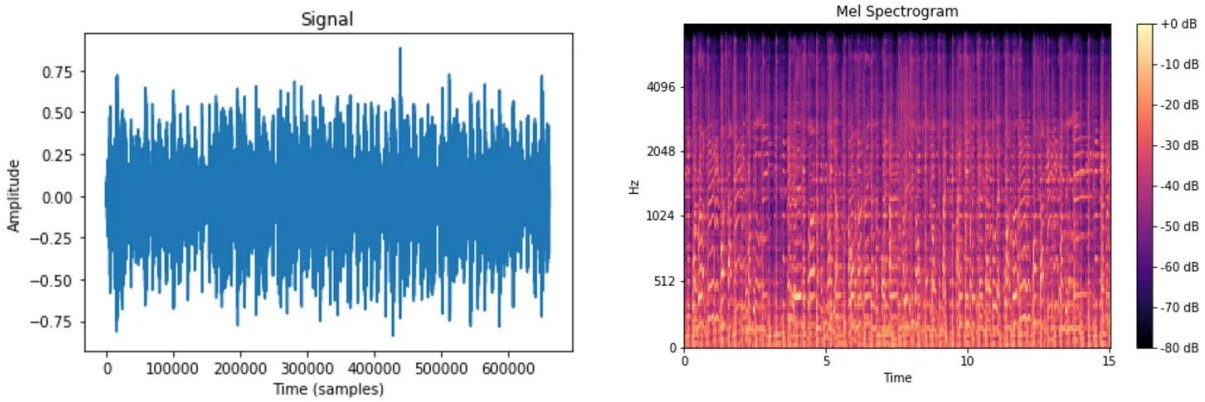
In this research, I propose a voice spoofing detection system utilizing Long Short-Term Memory networks. The proposed system takes mel-spectrogram images as input and produces a probability-based classification of voices as real or fake. A frequency masking experiment is also conducted to examine how correlated specific frequency bandwidths are when enhancing the accuracy of real-fake classification.

Background Knowledge

Mel-Spectrogram

A mel-spectrogram is a visual representation of sound that reflects how humans perceive audio. It is created by first applying a Short-Time Fourier Transform to convert an audio signal into the frequency domain, resulting in a spectrogram. The frequencies in this spectrogram are then mapped onto the mel scale, which spaces frequencies according

to human auditory perception, with lower frequencies spaced further apart than higher ones. The mel-spectrogram highlights perceptually relevant features of sound, and its amplitudes are often converted to a logarithmic decibel scale to better represent perceived loudness. This makes mel-spectrograms particularly useful for tasks such as speech recognition, music analysis, and voice spoofing detection (Ustubioglu et al. 2023).



(a) (b)

Figure 1. (a): input signal (time domain) and (b): mel spectrogram

In this study, the input time-domain audio signals are converted into mel-spectrogram images. The detailed information on this preprocessing step is further explained in Chapter 3.

Generative Adversarial Network

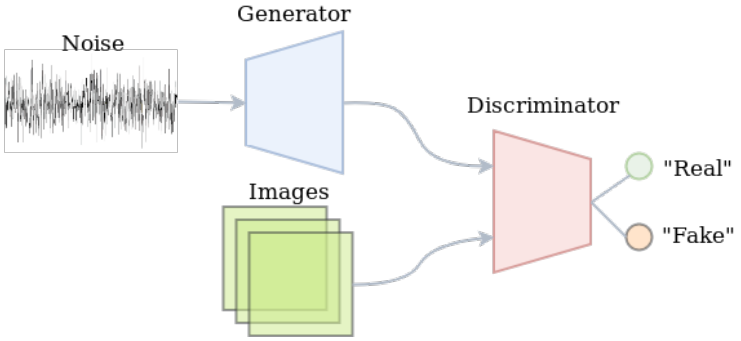


Figure 2. Architecture of generative adversarial network

A Generative Adversarial Network (GAN) is a type of machine learning model designed to generate new data that is similar to a given set of training data (Goodfellow et al. 2020). GANs are particularly popular for generating images, but they can also be used for other types of data like audio, text, and video. GANs are composed of two main components: the generator and the discriminator. The generator is a neural network that creates new data instances. Its goal is to generate data that is similar to the real data that the discriminator cannot distinguish between the real and generated data. The generator starts by taking in a random noise vector(usually a set of random numbers) and then

transforms this noise into a data instance, like an image. At first, the generated data will be far from realistic, but it improves over time as the GAN trains.

The discriminator is another neural network that evaluates the data it receives and tries to distinguish between real data (from the training set) and fake data (generated by the generator). The discriminator outputs a probability that the input data is real (i.e., comes from the training set) or fake (i.e., generated by the generator).

During training, the generator and discriminator play a game against each other. The generator tries to fool the discriminator by producing data that looks real, while the discriminator tries to get better at identifying real versus fake data. The training process could be explained in a sequence of four steps. The generator creates fake data from random noise. First, the discriminator evaluates a batch of real data and the fake data produced by the generator. Next, the discriminator updates its parameters to better distinguish between real and fake data. Third, the generator updates its parameters based on the feedback from the discriminator, learning how to create more realistic data.

Finally, The generator updates its parameters based on the feedback from the discriminator, learning how to create more realistic data. This process continues iteratively. Over time, the generator becomes better at creating realistic data, and the discriminator becomes better at identifying fakes. Ideally, they reach a point where the generator's outputs are so realistic that the discriminator can no longer reliably tell them apart from real data.

This GAN approach has been widely used to create fake voices due to its remarkable performance. However, it still struggles with generating finely detailed high-frequency components. In this research, I conducted frequency masking experiments to demonstrate this limitation and provide insights for improving the accuracy of deep voice detection systems.

Voice Spoofing Detection

Preprocessing

The input signal is transformed into a mel-spectrogram using the fourier transform. This process is capable of capturing information from both the time and frequency domain. If only the time-domain information is used, waves of different frequencies become mixed, making it difficult for the CNN to precisely analyze phonemes. Therefore, converting the signal into a mel-spectrogram, which contains both data of time and frequency, allows the CNN to analyze and classify phonemes more effectively, addressing the limitations of using the raw signal as input.

Voice Spoofing Detection Network

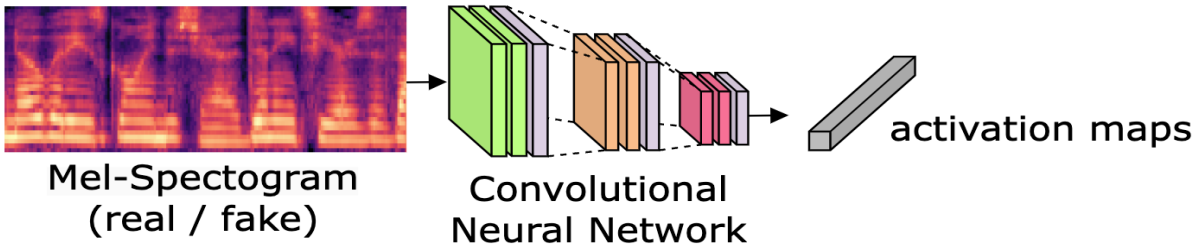


Figure 3. Using CNN to extract features from Mel-spectrograms

To invent a voice spoofing detection network, the usage of the convolutional neural network to extract essential features from the mel-spectrogram is necessary. The activation maps, containing key features from the mel-spectrogram, are the output of the convolutional neural network.

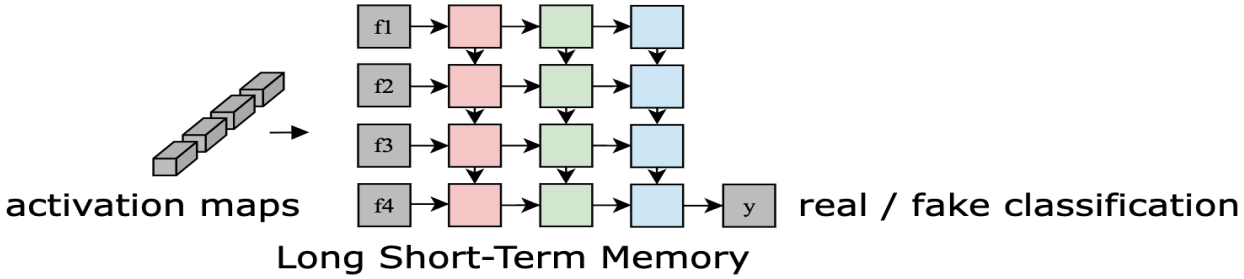


Figure 4. Architecture of the proposed LSTM-based voice spoofing network

These activation maps are then substituted into a deep-learning architecture, called the 'Long Short-Term Memory'. Before the activation maps are used, they are divided into four parts(for convenience, call these f_1, f_2, f_3, f_4). Then, change f_1, f_2, f_3 using the sigmoid function, and then change f_4 using the hyperbolic tangent function. These altered values would be called as, $f, i, g,$ and o, respectively. After that, we shall define the new cell state and the new hidden state as follows: $c_t = f \times c_{t-1} + i \times g$, $h_t = o \times tanh(c_t)$. As these become these new inputs of the Long Short Term Memory architecture, the training is able to be accomplished. Last, the probability, y, would be the output after a series of training inside the architecture, enabling the classification of fake and real signals. For example, if $y = 0.7$, then it would mean the detection network had predicted the input mel-spectrogram to be fake 70%, and real 30%.

Equation 1. Binary Cross Entropy Loss Function

$$BCE = -[\, y \times log_e(\hat{y}) + (1 - y) \times log_e(1 - \hat{y})\,]$$

Here, $\hat{y}$ and $\boldsymbol{y}$ denote the prediction of the proposed network and its corresponding ground truth, respectively. It is essential that the predicted value, $\hat{y}$, converges towards 0.0 or 1.0, as every signal has a binary ground truth(real or fake). To achieve this, by using the binary cross entropy loss function, it is possible to drive the $\hat{y}$ value closer to 1.0 or 0.0, depending on its corresponding ground truth.

Experimental Results

Artificial Voice Dataset

To train and test the proposed system, I used the Fake-or-Real (FoR) dataset (Kaggle 2024). The dataset is a comprehensive collection of over 195,000 utterances, including both real human speech and computer-generated speech. The dataset features a wide range of speech characteristics which provides a robust resource for training and testing machine learning models.

Evaluation Metrics and Experiment Design

I employed K-fold cross validation which is a widely adopted technique in machine learning for evaluating the performance and generalizability of predictive models. As illustrated in Figure 7, this cross validation method entails dividing a dataset into $\boldsymbol{K}$ equally sized folds or subsets. The model is then iteratively trained $\boldsymbol{K}$ times with each iteration utilizing different folds as the training set and the remaining data as the validation set.

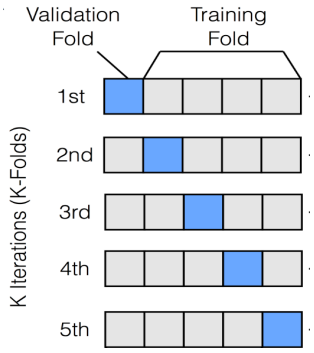


Figure 5. K-fold cross validation

I employed accuracy, and f1-score as the primary evaluation metrics. These are both the key metrics of evaluation object classification problems. Accuracy measures the overall correctness of the classification model and is calculated as the ratio of correctly predicted instances to the total instances. Recall quantifies the ability of a model to capture all relevant instances of a particular class. It is the ratio of correctly predicted positive instances to the total actual positive instances. Precision gauges the accuracy of positive predictions made by the model and is the ratio of correctly predicted positive instances to the total predicted positive instances. Finally, f1-score is the harmonic mean of precision and recall. It provides a balanced measure between precision and recall which is offering a single metric that considers both false positives and false negatives.

	equation	definition
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$	the proportion of correct predictions among the total number of cases examined.
Precision	$\frac{TP}{TP+FP}$	the proportion of true positive predictions in the total predicted positives.
Recall	$\frac{TP}{TP+FN}$	the proportion of actual positives that were identified correctly.
F1-score	$2\times\frac{P\times R}{P+R}$	the harmonic mean of precision and recall

Figure 6. Evaluation metrics used in this research

CNN Architecture Evaluation

The purpose of this process is to identify which CNN indicates the highest performance for the voice spoofing detection model. After conducting a series of experiments to state of the art CNNs, it was found out that the ResNet-101 model outperformed the other models, such as MobileNetV2, ConvNext and HRNet-W48, having 91.40% of accuracy and a 91.39% of f1-score.

Table 1. Two groups broken down with age ranges and the difference.

Method	Accuracy	F1-Score
MobileNetV2 (Sandler et al. 2018)	80.04	79.14
ConvNext (Liu et al. 2022)	84.09	83.84
HRNet-W48 (Wang et al. 2020)	87.14	87.66
ResNet-101 (He et al. 2016)	91.40	91.39

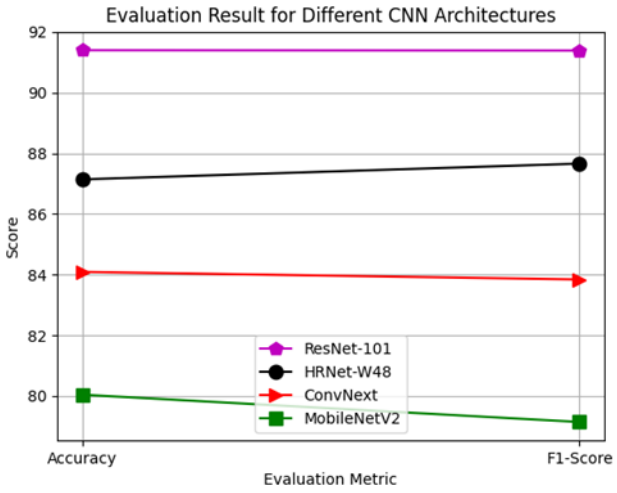


Figure 7. Evaluation result for different CNN architectures based on accuracy and f1-score

The below is the confusion matrix of the binary classification, using the model which had the best performance in the former experiment; ResNet-101

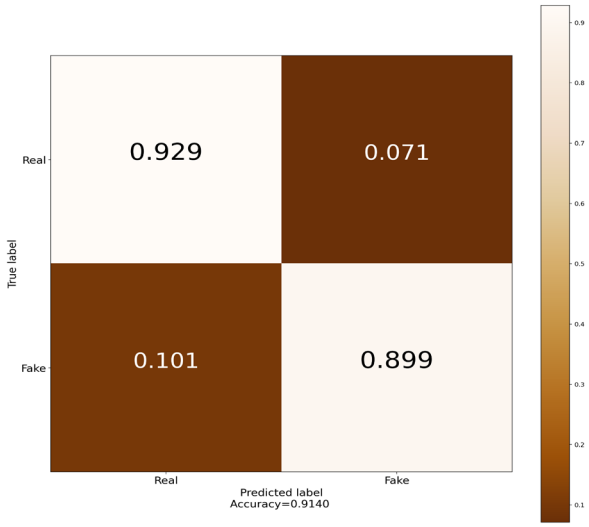


Figure 8. Evaluation result for different CNN architectures based on accuracy and f1-score

The accuracy of the model that predicted a real human voice signal as a human voice one is 92.9% whereas predicting it as a fake artificial voice is 7.1% based on figure 7. The accuracy of the model that predicted a fake artificial voice as a human voice signal is 10.1% whereas predicting it as a fake artificial voice is 89.9%. In sum, its total accuracy was 91.40%.

Table 2. The accuracy and f1 score of the models that conducted experiments based on the number of layers.

ResNet-101 based	**Accuracy**	**F1-Score**

Neural Network (layers: 3)	85.87	84.52
LSTM (layers: 4)	90.23	90.14
LSTM (layers: 5)	91.40	91.39
LSTM (layers: 6)	91.38	91.37

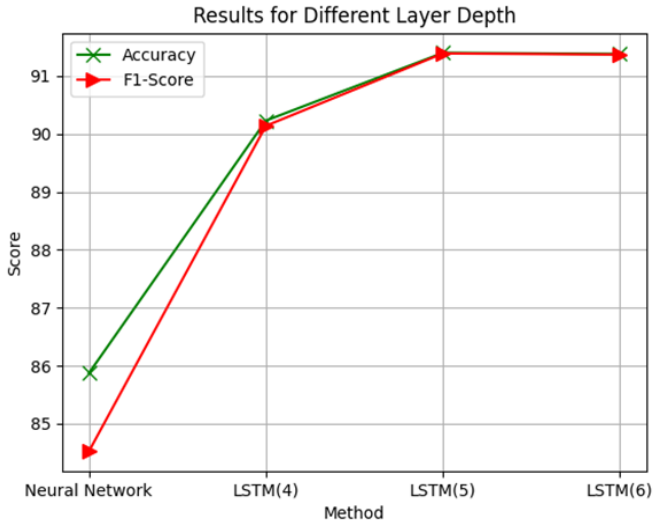


Figure 9. The accuracy and f1 score of the models that conducted experiments based on the number of layers.

As Table 2 and Figure 8 shows, another experiment had been conducted based on the different layer depth: neural networks had 3 layers whereas the LSTM models' layer depths were able to be altered from 4 to 6. Based on the experiment, the LSTM model that had 5 layers showed the highest accuracy and f1-score compared to other approaches with different layer depths.

Table 3. The accuracy and f1 score of the models that conducted frequency masking.

Frequency Masking	**Accuracy**	**F1-Score**
Band-D	81.05	81.12
Band-C	78.52	77.95
Band-B	89.76	88.42
Band-A	90.54	90.87
Baseline	91.40	91.39

Table 3 and Figure 9 shows the accuracy and f1-score of the given model when frequency masking, a method that checks the performance when each band of frequency is gone compared to the baseline experiment, is conducted. Here, I did not conduct the experiment of excluding the band with the lowest frequencies because they were crucial information based on the mel-spectogram. When conducting this experiment, significantly low performance was observed when masking bands with high frequencies. Since this experiment was conducted based on a sampling algorithm, this shows a weakness to the original system.

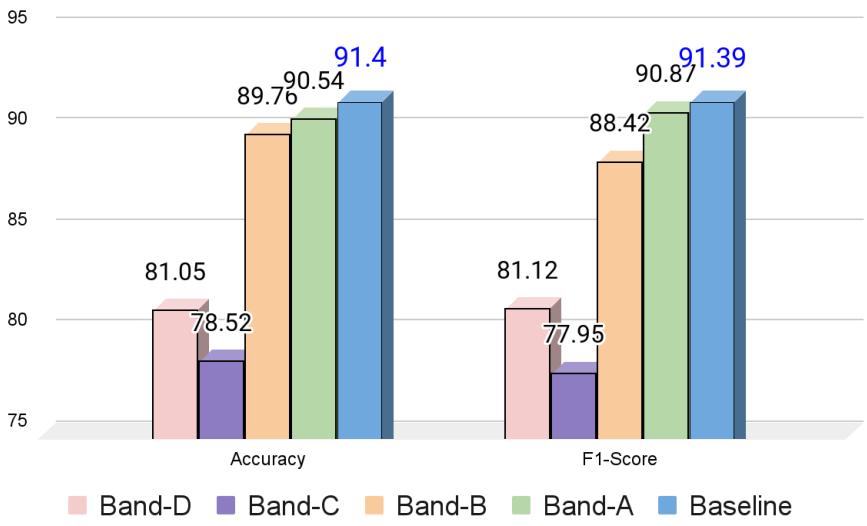


Figure 10. The accuracy and f1 score of the models that conducted frequency masking.

Conclusion

In this paper, I proposed a voice spoofing detection model that improves the binary classification (real or fake) of the input voice. The detection model, which is a combination of CNN and LSTM models, is constructed to have a voice as an input, changing that voice into a mel-spectrogram using the fourier transform, using the mel-spectrogram as the input for the CNN, and the activation maps, the output of the CNN, for the input of the LSTM. The experimental results indicated that the proposed network achieved an accuracy of 91.40%, and an F1-score of 91.39%, outperforming previous models using other CNN architectures. The training of the activation maps utilizing the LSTM model showed better results than using neural network models, and also showed that 5 layers is the optimal choice for high accuracy. The research also highlights the necessity to complement a deficiency: the algorithms intended to create artificial samples show low accuracy in high-frequency signals when utilizing frequency masking. In the future, I intend to further research for the feasibility of this method. The primary goal of the research would be the usage of this technique in the real world, helping people with the detection of voice spoofing.

Acknowledgments

I would like to thank my advisor for the valuable insight provided to me on this topic.

References

Cohen, A., Rimon, I., Aflalo, E., & Permuter, H. H. (2022). A study on data augmentation in voice anti-spoofing. Speech Communication, 141, 56-67.

Ergünay, S. K., Khoury, E., Lazaridis, A., & Marcel, S. (2015, September). On the vulnerability of speaker verification to realistic voice spoofing. In 2015 IEEE 7th international conference on biometrics theory, applications and systems (BTAS) (pp. 1-6). IEEE.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2020). Generative adversarial networks. Communications of the ACM, 63(11), 139-144.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778). https://doi.org/10.48550/arXiv.1512.03385

Kaggle. (2024, Sep 4). "*The Fake-or-Real (FoR) Dataset (deepfake audio)*": Kaggle. https://www.kaggle.com/datasets/mohammedabdeldayem/the-fake-or-real-dataset

Liu, Z., Mao, H., Wu, C. Y., Feichtenhofer, C., Darrell, T., & Xie, S. (2022). A convnet for the 2020s. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 11976-11986).

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4510-4520). https://doi.org/10.48550/arXiv.1801.04381

Ustubioglu, A., Ustubioglu, B., & Ulutas, G. (2023). Mel spectrogram-based audio forgery detection using CNN. Signal, Image and Video Processing, 17(5), 2211-2219.

Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., ... & Xiao, B. (2020). Deep high-resolution representation learning for visual recognition. IEEE transactions on pattern analysis and machine intelligence, 43(10), 3349-3364. https://doi.org/10.48550/arXiv.1908.07919