Metadata of the chapter that will be visualized in SpringerLink

Book Title	Data Science	
Series Title		
Chapter Title	Real-Time AI Voice Clone Detection: A Deep Learning Approach to Safeguard Authenticity	
Copyright Year	2025	
Copyright HolderName	The Author(s), under exclusive license to Springer Nature Switzerland AG	
Author	Family Name	**Chen**
	Particle	
	Given Name	**Cody**
	Prefix	
	Suffix	
	Role	
	Division	Department of Computer and Information Science
	Organization	Fordham University
	Address	New York, NY, 10023, USA
	Email	cchen187@fordham.edu
	ORCID	http://orcid.org/0009-0007-1391-218X
Corresponding Author	Family Name	**Hayajneh**
	Particle	
	Given Name	**Thaier**
	Prefix	
	Suffix	
	Role	
	Division	Department of Computer and Information Science
	Organization	Fordham University
	Address	New York, NY, 10023, USA
	Email	thayajneh@fordham.edu
	ORCID	http://orcid.org/0000-0002-8952-1499

Abstract	The proliferation of voice-activated technologies and the increasing sophistication of AI-generated voice clones pose significant security challenges. Speaker identification, despite advancements in automatic speech recognition (ASR) and natural language processing (NLP), requires more robust authentication mechanisms. This paper explores the potential of deep learning models to distinguish between authentic human speech and AI-generated voice clones. Due to the limited availability of AI-generated voice datasets, we created a custom dataset using both commercial and open-source voice cloning tools. We employed a Convolutional Neural Network (CNN) combined with a Gated Recurrent Unit (GRU) to classify voice samples as authentic or AI-generated. Our results demonstrate the potential of deep learning in detecting AI voice clones, providing a foundation for future research into more comprehensive and secure speaker authentication methods.

Keywords (separated by '-')	AI voice cloning - Deep learning - Speaker recognition - Convolutional Neural Network (CNN) - Gated Recurrent Unit (GRU)

Real-Time AI Voice Clone Detection: A Deep Learning Approach to Safeguard Authenticity

Cody Chen and Thaier Hayajneh(✉)

Department of Computer and Information Science, Fordham University, New York, NY 10023, USA
{cchen187,thayajneh}@fordham.edu

Abstract. The proliferation of voice-activated technologies and the increasing sophistication of AI-generated voice clones pose significant security challenges. Speaker identification, despite advancements in automatic speech recognition (ASR) and natural language processing (NLP), requires more robust authentication mechanisms. This paper explores the potential of deep learning models to distinguish between authentic human speech and AI-generated voice clones. Due to the limited availability of AI-generated voice datasets, we created a custom dataset using both commercial and open-source voice cloning tools. We employed a Convolutional Neural Network (CNN) combined with a Gated Recurrent Unit (GRU) to classify voice samples as authentic or AI-generated. Our results demonstrate the potential of deep learning in detecting AI voice clones, providing a foundation for future research into more comprehensive and secure speaker authentication methods.

AQ1

AQ2

Keywords: AI voice cloning · Deep learning · Speaker recognition · Convolutional Neural Network (CNN) · Gated Recurrent Unit (GRU)

1 Introduction

The expansion and integration of Artificial Intelligence (AI) across various sectors have been increasingly evident in the past year. Large Language Model (LLM) platforms like OpenAI's ChatGPT and Microsoft's Copilot attempt to be an everyday companion making AI more accessible and familiar to a broader audience. Although the spotlight on AI might seem recent, this technology has been in development for over six decades. Prominent examples of AI applications today are virtual assistants such as Siri, Alexa, and Cortana. These advanced systems leverage several AI/ML technologies, including Natural Language Processing (NLP), Deep Neural Networks (DNN), and speech recognition. This allows virtual assistants to comprehend and respond to user commands, interpret human speech, execute tasks, and answer queries. The physical implementation of AI technologies has been seen in robotics through companies like Boston

R. Stahlbock and H. R. Arabnia (Eds.): CSCE 2024, CCIS 2253, pp. 1–15, 2025.
https://doi.org/10.1007/978-3-031-85856-7_14

Dynamics and their development of the "Atlas" robot with intelligent decision-making. These systems can learn from interactions, continuously refining their responses over time through reinforcement learning.

Despite the advancements in technology offering immense benefits, they also open new vulnerabilities for threat actors to exploit them for malicious intents. The sphere of cybersecurity is continually challenged by growing threats like ransomware, malware, and an assortment of social engineering techniques. Among these, phishing attacks, especially the emerging trend of 'vishing' or voice phishing, pose significant concerns. Vishing, a technique wherein fraudsters manipulate victims over the phone to extract sensitive information, is significantly increasing. These scams cleverly capitalize on the trust a voice call seemingly provides, luring susceptible individuals into divulging confidential details. Phone scams are experiencing an alarming increase in frequency. According to estimates from Truecaller, a leading spam call-blocking app, approximately 70 million Americans fell prey to phone scams in 2022 alone, leading to a staggering loss of nearly 40 billion dollars [1].

Since many vishing attacks are executed through automated recordings, these threats effortlessly target individuals. Attackers skillfully employ NLP, DNNs, and other machine-learning capabilities to mimic human speech convincingly, thereby deceiving victims into trusting the call's authenticity. The Internal Revenue Service (IRS) warned taxpayers about a variant of this attack in recent scam calls. In this scheme, a robot-generated call falsely informs individuals that they owed money to the IRS, pressing them to make immediate contact to resolve the fictitious issue. This case underscores the sophisticated use of technology by threat actors and the pressing need for equally advanced countermeasures [2].

Robot-generated scam calls are not new and are relatively easy to spot due to how robotic and monotone the voices are. However, with the increasing improvements in Generative Artificial Intelligence (Gen AI), voice clones can now be synthesized to sound extremely human-like and identical to the person that was sampled. Cybercriminals can and have abused this technology for campaigns such as more sophisticated vishing attacks or voice biometric spoofing. This paper aims to further this line of inquiry by constructing and training various deep learning models intended to test their ability to discern if the speaker is an AI voice clone and deepen our understanding of AI's impact on cybersecurity.

2 Related Works

The history of speech synthesis can be traced back to December 20, 1845, when "The Wonderful Talking Machine" was displayed and the inventor, Joseph Faber, claimed it could talk. This invention used a small chamber organ, strings, and levers with a wooden face to regulate the airflow leaving the chamber and simulate how a human would speak [3]. Many more mechanical and electromechanical adaptations of the machine would be built but it wouldn't be until 1936 that Homer Dudley introduced a true electrical speech synthesizer from Bell Labs called the "VODER". Following the success of this machine came the birth of research into creating machines that can simulate the human voice [4].

From that idea, we now have text-to-speech applications that can produce spoken computer-generated sentences from a given text prompt. Depending on the level of sophistication, these applications can even mimic human speech. With the commercialization of Generative Artificial Intelligence, voice cloning has been made easier through speech-to-speech technology. In [5], a survey research was done on the state of Generative AI and outlined the latest platforms/companies that have made advancements with text-given audio generation. Listed in the findings were Coqui, Descript Overdub, ElevenLabs, Lovo AI, Resemeble AI, Replica Studios, Voicemod, Wellsaid, and AudioLM. Speech-to-speech platforms were included in their survey such as ACE-VE and VALL-E.

Through these platforms, AI-synthesized audio deepfakes are possible allowing adversaries to clone a victim's voice and use it for malicious reasons. The work in [6] shows that recent advancements in this area have resulted in algorithms that can produce realistic cloned voices indistinguishable from the real voice. This breakthrough was important but lacked quality and naturalness which recently came with improvements in deep learning techniques. Models like WaveNet, Tacotron and DeepVoice3 are examples of generating synthetic speech from text inputs while maintaining realism utilizing different configurations of DNN, encoder and decoder, and recurrent neural networks. To detect audio manipulation, multiple approaches have already been attempted with models built on large margin cosine lost function, DenseNet-BidrectionalLSTM, light Convolutional Neural Network, and ResNet.

Building off the limitations mentioned in [4,6] looked into developing a voice cloning system that will maintain the naturalness of synthetically generated voices for longer text inputs. This was achieved by improving how letters were pronounced by adding a "text determination module" to the synthesizer module of a voice cloning system that splits the letters in a word. In implementation, a noise reduction algorithm combined with the SV2TTS framework was used to replace the pre-net module of the synthesizer module. The proposed module was tested by generating male and female voices, long and short sentences, extra lexical words, and different voice tones. These voice samples were compared to those generated by existing speech synthesizing models and found that their proposed model offers superior fluency, naturalness, and clarity showing how fast voice cloning systems are advancing.

Transitioning to the threat landscape of this technology, [6] briefly discusses the malicious activities threat actors can perform while utilizing audio deepfakes. To investigate this, [7] tested the security of voice assistants when confronted with a clone of another person's voice. In their work, they developed a pipeline using Coqui YourTTS and a Telegram bot to generate voice clones in attempts to spoof an Amazon Alexa. Through Coqui, voices were cloned and audio samples were generated from a given text input which was delivered through a Telegram bot as audio messages. Four voice profiles were saved in the Alexa, 2 male and 2 female voices, and all were cloned. Additional testing was done on male-to-male cloning, female-to-female cloning, and cross-gender cloning. Results showed

male-to-male cloning successfully spoofed Alexa 100% of the time, female-to-female was successful 80% of the time, and cross-gender was unsuccessful.

Another approach, [8], proposed a model to fight AI with AI to detect fake speech with deep learning. In this work, the attack model of fake speech was defined as either an impersonation attack or an injection attack. However, they claim that these attacks can be detected since generative models often leave artifacts in the cloned audio samples that can be distinguished at the spectrogram level. To verify this, a Convolutional Neural Network was built to identify those artifacts from the spectrograms and classify them as either real or cloned. The proposed CNN consisted of four convolutional layers with a max pooling layer in between followed by a flattening layer into a fully connected network for classification. When reviewing the results, it showed that the CNN model was able to achieve 100% accuracy when detecting real and cloned samples on the test dataset.

Although not designed directly for detecting voice clones, the hybrid CNN-GRU model proposed in [9] showed promising results in speaker identification. Similar to [8], this process used convolutional neural networks to isolate characteristics from a spectrogram but with the additional power of a gated-recurrent unit. Since audio samples also exist on a time-series domain, the GRU will help with feature learning for the time-series information of the spectrogram. The model consists of two convolution and pooling layers that are flattened and fed into three GRU cells which are outputted to a fully connected layer for classification. Their results showed the model having a 98.96% overall accuracy on the testing data and 91.56% accuracy when Gaussian noise was added.

3 AI Voice Cloning Overview

As mentioned in the earlier section, the concept of computer-synthesized voices has been a decades-old research concept to generate natural-sounding voices from a given text input. A well-known implementation of early AI voice usage is the TTS synthesizer built into the wheelchair of the late Stephen Hawking who communicated through the system during his battle with Amyotrophic Lateral Sclerosis (ALS).

3.1 How Does It Work?

A high-level overview of the framework behind a typical AI voice synthesizer is comprised of gathering and processing a given text input through text analysis and producing a waveform that closely resembles how the text input would sound when spoken. With the introduction of Gen AI with TTS, this action is done with the integration of "Audio Diffusion Models" [10]. This type of system consists of two main modules called the "Synthesis module" and the "Vocoder". In the synthesis module, text analysis is performed by correlating the given text to acoustic features from a stored sample database to generate mel-spectrogram

images. This is passed to the Vocoder to generate audio frequencies which essentially converts the mel-spectrogram images to audio waveforms that can be used for audio outputs. In the Vocoder, audio diffusion models generate waveforms based on acoustic features through methods such as autoregression [10]. A popular system that integrates audio diffusion into TTS is "Tortoise TTS" which applies autoregression with denoising diffusion probabilistic models (DDPMs) to improve the generation of long continuous text inputs [11].

Voice conversion is another type of voice synthesizing system where one person's voice recording is used to sound like another person's. Typical voice conversion models require a large dataset of speech recordings of the targeted speaker. Retrieval-based voice conversion (RVC) is an open-source tool that performs lightweight and fast short speech conversions that can be used in real-time voice-to-voice transformation. In [12], RVC was used in generating models that can convert a given sample recording to sound like well-known public figures.

3.2 Threat Landscape

Through the integration of AI voice clones, many systems can achieve more user-friendly interfaces. Aside from text-to-speech applications, this technology has been seen in various other categories such as language translation. By modifying the output type, RVC-enabled systems can perform real-time language translation during voice conversion. This allows individuals who speak two different languages to have full conversations with minimal difficulty. Media content creation is another field where this technology has impacted as the concern of voice rights becomes increasingly discussed.

While the benefits of AI voice cloning are promising, AI also negatively changes the cyber-threat landscape by providing threat actors with a new set of tools they can use to execute cybercrime. Through AI voice cloning, cybercriminals can perform impersonations, spoof voice biometric-enabled systems, and fraudulent scans/vishings. The work in [12] shows how powerful RVC can be by generating 8 cloned models of public figures. From a threat actor or nation-state's perspective, they can impersonate a public figure and can "leak" information to the public and spread it amongst news networks. The range of impact can quickly escalate from harmless impersonations to matters of national security. In [7], virtual assistants that operate through voice commands can be spoofed by AI voice clones giving unauthorized individuals the ability to make queries. This allows threat actors to force Internet of Things (IoT) devices into performing actions which they were not authorized such as unlocking doors with voice-activated locks or making purchases on an Amazon Alexa. Depending on the voice cloning platform, the minimum requirement of a reference sample is 5 s which can be collected with a brief phone call making this exploit extremely feasible for a threat actor to execute.

4 Understanding Machine Learning Models

Machine learning is a sub-field of Artificial Intelligence that involves developing algorithms and models to learn from data and improve their performance over time. Machine learning aims to enable computers to learn and make predictions or decisions, like in Fig. 1, without being explicitly programmed.

4.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) share similarities with traditional Artificial Neural Networks (ANNs) in their use of self-optimizing neurons that process inputs through operations such as scalar products and non-linear functions [13]. Despite these similarities, CNNs have been adapted specifically for pattern recognition within images, making them more suited for image-focused tasks and reducing model setup parameters by adding convolutional operations for feature extraction before the fully connected layers seen in ANNs. Even though CNNs are used primarily for image-focused tasks, they can be applied to audio data when it has been transformed into a 2-D matrix in the form of spectrograms. A typical CNN is comprised of convolutional and pooling layer(s), followed by a flattening layer and a fully connected layer(s).

4.2 Gated Recurrent Units

Gated Recurrent Units (GRUs) are a variant of Recurrent Neural Networks (RNNs) tailored for handling sequential data [14]. It is a simplified version of a Long Short-Term Memory unit, another variant of RNNs, where fewer gates are used which makes them faster. GRUs, with their inherent design, can effectively maintain the memory of previous inputs in a sequence, making them apt for processing time-series data or any data with temporal dependencies. The distinctive aspect of GRUs lies in their gating mechanisms – specifically, the reset and update gates. These gates, through intricate interactions, allow the GRU to mitigate the vanishing gradient problem, a limitation observed in traditional RNNs, and to capture long-term dependencies in sequential data.

4.3 The CNN-GRU Model

The integration of CNNs and GRUs into a unified model harnesses the strengths of both architectures. In the hybrid CNN-GRU model, the input data first undergo processing by the convolutional layers from a CNN, which extract crucial spatial features. These extracted features, now transformed into a sequence of feature maps, are subsequently fed into the GRU layers. This sequential operation ensures that the model efficiently captures the spatial patterns by the CNNs and the temporal dependencies via the GRUs.

Such a combined approach is particularly advantageous for tasks requiring simultaneous extraction of spatial and temporal information. For instance, in

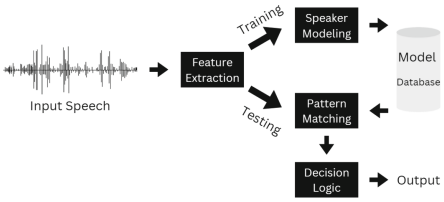

Fig. 1. Speaker recognition architecture

audio-based tasks like speaker recognition, while the CNNs adeptly extract features from spectrograms (representing different sound frequencies over time), the GRUs subsequently analyze these features in their temporal sequence. This collaboration aids in identifying and characterizing unique speaker attributes based on the temporal evolution of audio features. The CNN-GRU hybrid model stands for the versatility achieved by matching spatial feature extraction with temporal sequence analysis, offering robust capabilities for many applications. AQ3

CNN-GRU Framework. The CNN-GRU neural network is built following the architecture proposed by [9] with slight modifications as shown in Fig. 2. The framework consists of three components, the convolutional block, GRU block, and fully connected block. In the convolution block, two layers of convolutions are used with max pooling following it with ReLU as the activation function. Batch normalization is also utilized in this block to help stabilize optimization and dropout is placed before flattening to prevent overfitting. The features extracted by the convolution block are passed to the GRU block which is made up of three GRU units linked together. The output from the GRU block is passed to three fully connected layers with ReLU in between and Sigmoid for final classification. The only hyper-parameters for the neural network are the stride lengths, number of kernels, kernel size for the convolutional and pooling layer, and number of hidden states in the GRU.

5 Methodology

The paper aims to discover if the implementation of deep learning models can distinguish between authentic human speakers and AI-generated voice clones. Locating a dataset of actual human voices was straightforward, given the rich availability of such features for analysis. However, a challenge arose in sourcing a dataset of AI-generated voices due to the limited information available in this domain. In response to this, a proactive approach was taken by generating a custom dataset meeting the study's specific requirements, thereby overcoming this limitation.

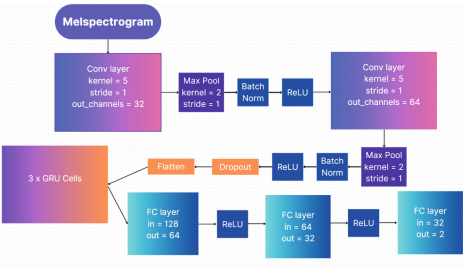

Fig. 2. CNN-GRU Network Framework

5.1 Datasets

The Sampled Speaker Dataset. The acquisition of a sampled speaker dataset was simple. It was obtained from the LibriVox project, which is an organization that offers free audiobooks generated from public-domain texts. LibriVox's diverse array of audio content makes it a popular resource for NLP studies. From their library of audiobooks, various speaker-specific datasets were curated such as the LJ dataset. The LJ dataset is a collection of 13,100 short audio clips by a single reader from the LibriVox library. Seven non-fiction books make up the 13,100 audio samples varying from 1 s to 10 s in length with the average length being 6.57 s. In total, approximately 24 h of audio was collected. Additionally, each audio clip was sampled at 22050 Hz and outputs as a single-channel 16-bit PCM WAV file [15]. However, the original LibriVox recordings were 128 kbps MP3 files and the LJ audio clips may contain artifacts from the MP3 encoding and conversion to WAV.

The AI Speaker Dataset. Securing a comprehensive dataset that aligns with the objectives of this paper presented challenges due to the lack of available open-source resources. To address this, the initiative was taken to create a custom dataset tailored to the project's requirements as well as experiment with how easily AI voice clones can be generated from a threat actor's point of view. Two types of AI voice cloning tools were employed for this paper; a commercial-grade product and an open-source implementation were chosen to test the difference between these two approaches.

Speechify is a TTS application known for having various speakers and audiobook features. However, Speechify has become an industry leader in AI voice cloning by offering services to users to generate their voice clones for their TTS needs. To create an AI-generated voice sample, a 5-second clip was required as a sample followed by a text prompt for the output. The generation time of a clip varied on the length of the prompt but was fairly quick with most clips generated in less than 10 s. A total of 2000 AI voice clone samples were generated with half of the dataset being direct clones of the LJ dataset by using the audio transcripts as text prompts. The other half consists of random text prompts

that were generated with the assistance of OpenAI's ChatGPT chatbot built on the GPT-3.5 LLM architecture. Throughout the dataset generation, different LJ clips were sampled to get a variety of LJ's intonation and emotions. Only 2000 samples were generated at the time due to a change in Speechify's subscription model for the AI voice cloning tool resulting in limited features with the tool.

A Tortoise TTS-based platform with RVC output conversion and a web-UI RVC tool was used as the open-source alternative as it is publicly available on GitHub. These tools allow more than one recording sample to be used during inference resulting in a higher-quality model. A tortoise autoregressive model was trained using the Tortoise based platform while an RVC model was trained using the web-UI RVC software which was combined with the Tortoise model. The same process from the Speechify dataset was used in the dataset curation on the open-source platforms. More samples could be generated through this method but were limited to the same size as the Speechify dataset for continuity reasons.

5.2 Data Pre-processing and Transformation

Data Pre-processing. For our data to be accepted as input vectors by the CNN-GRU, data pre-processing and transformation needed to be done. Due to how the LibriVox dataset was initially presented, the data needed to be reorganized. A new metadata file was created to facilitate data flow between the network and the dataset repository. The samples in the repository were separated into training, validation, and test sets. Due to a large dataset imbalance due to the limited AI voice cloning, only 2000 LJ samples were used for training, validating, and testing. This resulted in the usable dataset consisting of 4000 samples of which 3,200 was the training set, 400 was the validation set and another 400 were held out for testing.

Data Transformation. Transformation is an important step in the methodology as audio samples are analog by nature and need to be transformed into digital signals that can be processed by a computer. Features are extracted from the audio files during this process which will become the input vectors to the neural networks. The initial transformation of audio samples consists of isolating the sample's waveform. Waveforms exist in a time domain that shows how the amplitude of the audio sample changes over time which is equivalent to the loudness. Additional information can be extracted from the waveform such as the hertz range which is a crucial part of speaker recognition as everyone typically speaks at their unique frequency ranges. This information can be extracted using a Fourier Transformation called the "Short Time Fourier Transformation" represented with the following equation.

$$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \xi} dx \tag{1}$$

The information extracted after the Fourier Transformation is the frequency range of the audio sample and the amplitude of each frequency band within

that range over time. As a result, the data is transformed from a time domain to a time-frequency domain called Spectrograms. The spectrograms are then converted to Mel Spectrograms by applying the Mel Scale to ideally represent the perpetual change of the amplitude and frequency of a sample.

Mel Spectrograms are generated for every sample and are stored in a class object to be batched for training. For standardization purposes, when performing the transformations, all samples were limited to a 7-second duration as the average length of the LJ clips was 6.57 s. Samples that were longer were cut down and shorter samples were zero-padded to resemble no sound, which should have negligible effects on classification.

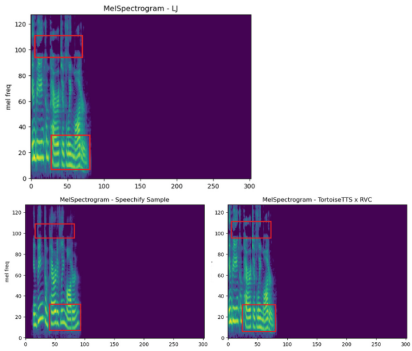

Fig. 3. Mel-spectrogram comparison of all 3 datasets with the same sentence. The boxes highlight areas of similarities.

5.3 Training

As mentioned earlier, the training sets consisted of 3,200 samples and had two classes, LJ and AI. Post-data transformation, the dataset was batched into sizes of 128 samples per batch to optimize training time and memory usage. The dataset was also split into a train and validation for cross-validation, resulting in a train and validation dataloader with 25 and 4 batches, respectively. Two variations of the dataset were assembled, one for LJ to Speechify and one for LJ to open-source comparisons.

Different from the training procedure in [9], this implementation of the proposed CNN-GRU network was implemented using PyTorch as opposed to TensorFlow. Additionally, the hardware utilized for training was a dedicated Nvidia RTX 4070 Ti Super GPU with 8448 CUDA cores, a 2.61 GHz boost clock, and 16GB GDDR6X memory.

Hyper-parameter selection during training was done using a brute force approach resulting in multiple models trained but a set of hyper-parameters were found to produce promising results. These parameters were used for the data

transformation and determine how granular the amount of features will be extracted. Variables that were constant throughout the training were the optimizer, the learning rate, and the loss function. Adam optimizer was used with a learning rate of 0.0001 followed by cross-entropy loss function as the criterion. The following is a summary of the transformation parameters and epochs set for each model:

1. "LJ-Speechify 1": sample_rate = 16000, n_mels = 64, num_samps = 22050, [1,64,302], epochs = 100
2. "LJ-Speechify 2": sample_rate = 16000, n_mels = 64, num_samps = 22050, [1,64,302], epochs = 200
3. "LJ-RVC 1": sample_rate = 16000, n_mels = 64, num_samps = 22050, [1,64,302], epochs = 100
4. "LJ-RVC 2": sample_rate = 16000, n_mels = 64, num_samps = 22050, [1,64,302], epochs = 200
5. "LJ-RVC 3": sample_rate = 16000, n_mels = 64, num_samps = 22050, [1,64,302], epochs = 100, 4 GRU units

6 Results

In this section, we will discuss the various models that were trained, the effectiveness of the training, evaluate the models on testing data, and discuss the difference between the commercial and open-source approaches.

6.1 Model Results

As a baseline metric, an accuracy check was conducted for each model on the validation and test set before training to get an understanding of how well the framework performs without training. This will not only offer another way to evaluate training performance but also provide quick insights into how the input shape affects performance. For the majority of the models, the baseline accuracy was consistently 50% with a few instances of 40%–50% accuracy. Post-training evaluation metrics are as follows (Fig. 4): AQ4

Model	Training Time	Accuracy	Precision (AI)	Recall (AI)	F1-Score (AI)	ROC AUC Score	Confusion Matrix	Vector Shape
LJ-Speechify 1	5m 32.5s	0.89	0.919	0.855	0.886	0.89	171 15 29 185	[1,64,302]
LJ-Speechify 2	11m 1.3s	0.94	0.944	0.935	0.939	0.94	187 11 13 189	[1,64,302]
LJ-RVC 1	9m 13.8s	0.797	0.832	0.745	0.786	0.797	149 30 51 170	[1,64,302]
LJ-RVC 2	18m 19.4s	0.532	0.517	0.985	0.678	0.532	197 184 3 16	[1,64,302]
LJ-RVC 3	9m 54.9s	0.81	0.829	0.78	0.809	0.815	156 32 44 168	[1,64,302]

Fig. 4. Evaluation Metrics

The chart depicts the post-training performance of the 5 models trained and the scores collected on the test set. The metrics used for evaluation consist of accuracy, precision, recall, F1-score, ROC AUC score, and confusion matrix. Since this is a classification task, successfully predicting the AI speaker is considered the True Positive class and successfully predicting LJ is the True Negative class. As a result, precision, recall, and F1-score were calculated based on detecting the AI speaker.

6.2 Model Evaluation

Five models were trained on this framework and their performance improved drastically after training. Since these models are trained for a classification task and the training set is manually balanced, F1-score and ROC AUC score will be the metrics used to evaluate model performance. As shown in the results table, two series of models were trained based on the voice cloning software used to compare the platforms' effectiveness and the model framework's robustness. Starting with the LJ-Speechify models, the first round of training already showed promising results with an F1-score of 0.886 and an ROC AUC score of 0.89. This performance rapidly increased in the second round of training when 200 epochs were used instead of 100 resulting in an F1-score of 0.939 and an ROC AUC score of 0.94. When comparing the confusion matrices of the two models, we see a decline in the number of false negatives meaning fewer cloned samples were considered legitimate samples.

Switching over to the models trained on the Tortoise TTS + RVC samples, we see them perform slightly worse compared to Speechify samples. Results from the first round of training showed decent performance with an F1-score of 0.786 and ROC AUC Score of 0.797 but a significant drop compared to LJ-Speechify 1. Following the same process of increasing epochs to 200, we see an even more significant drop in results with an F1-score of 0.678 and ROC AUC Score of 0.532. Based on the ROC AUC score, LJ-RVC 2 is having difficulties differentiating the two samples which is evident in the confusion matrix with a 92% false positive rate, where the opposite occurred with LJ-Speechify 2. In the third round of training, an extra GRU cell was added to test for differences and performance was slightly improved. The F1-score increased to 0.809 and ROC AUC score increased to 0.815.

Overall, the models showed promising results, with the exception of the second model trained on the Tortoise TTS + RVC outputs. Observing training times, we see the models that experienced 100 epochs completed under 10 min and the models that experienced 200 epochs finished under 20 min. The speed of the training times is probably due to limiting the clips to 7 s during the mel-spectrogram transformation. Figure 5 shows the training and validation losses of the models; although the models produce decent results, they are underfitting.

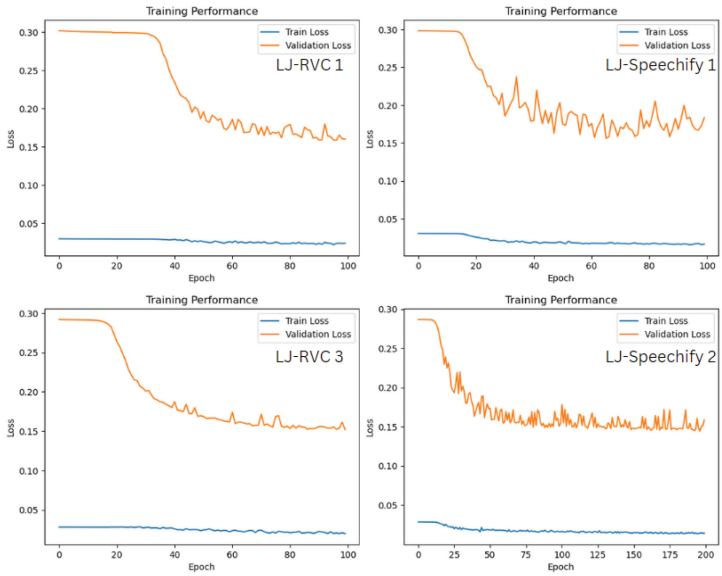

Fig. 5. Training Performance

6.3 Voice Cloning Evaluation

When comparing the process of using a commercially produced product to an open-source tool, significant differences were observed. Speechify offers a web application platform for its voice cloning tool and has an easily navigable user interface. The tool handles the technical adjustments and only requires the user to provide a recording sample and text prompt. As a web application, the tool is run server-side which takes the processing load away from the user. Transitioning to open-source tools, the operation becomes more complex and technical as the services are executed locally. Depending on the tool, there are excellent AI voice cloning tools actively maintained on GitHub and Hugging Face. When using an open-source tool, manual configuration is required to get the tool running but it gives the user more range of customization if they understand the settings. Additionally, computing power is a factor as processes are done locally requiring a powerful GPU and memory overhead to train powerful models.

As seen from the results and the mel-spectrogram comparisons in Fig. 3, the open-source approach outperforms the commercial approach. The open-source tools generate samples that are more identical than the commercial tool. However, even if the quality is better, there are drawbacks to consider if a threat actor were to use this approach. The open-source tools require a significant amount of sample material to train a high-quality model and perform inference. The Tortoise TTS + RVC model for this paper had approximately 24 h of sample data to train on, which resulted in nearly identical clones. In a real-world scenario, a threat actor might not have that many audio recordings of an individual,

unless they're a public figure or a live conservation was recorded without consent. Even then, many variables need to be accounted for such as background noise, quality of the microphone, recording method, muffled sound, etc. This would prompt the use of an alternative like a commercial tool, as cloning is possible with shorter/fewer samples but at the cost of quality.

7 Conclusion and Future Work

This paper highlights several key points that are crucial for those working in voice authentication to understand. There is potential through a deep learning approach to safeguard the authenticity of human voices from a security perspective in an age where they can be cloned with artificial intelligence. Using the tools shown in this paper, threat actors can easily generate cloned voices with the click of a few buttons which reinforces the notion that an efficient detection approach is necessary. Currently, the only defense deterring a threat actor is the slightly labor-intensive nature of manually curating cloned voice recordings and gathering the voice samples required. For more experienced cybercriminals, this problem could be solved with intelligently designed scripts and web bots. With the current lack of protection against AI voice clones, having a relatively simple model like the CNN-GRU model (estimated size is 1.66 Mb) can make a lot of difference as shown by its effectiveness at detecting the AI voice used in this paper.

In terms of future research, there are still many more hyper-parameters and architectural aspects that hasn't been experimented on such as different loss functions, optimizers, learning rates, and regularization techniques. Hyper-parameters were chosen based on standard practices for deep learning projects, but grid search could be implemented to learn optimal hyper-parameters. Data limitations are burdens for any machine learning project; in this case, an insufficient amount of data can be causing the underfitting issues. Additionally, long audio samples were cut down during data transformation and shorter samples were padded with "unnecessary" data. Robustness also becomes a concern as there are many different AI voice cloning options and multiple models will be required if more than one individual needs authentication.

Another aspect of this research can focus on addressing a broader range of artificial voices, as its popularity increases and access to it becomes more viable. Experiments can include other voice cloning tools since this project only utilized two programs. There are commercial tools that are arguably better than Speechify such as ElevenLabs, Voice.AI, and Murf.AI which can also be utilized. Internet of Things devices can also be introduced into this project as most home IoT devices are voice-activated, which can easily be tricked with AI voice clones. A training and detection pipeline can be developed using cloud computing on edge devices where they communicate with the cloud for processing needs. Many concerns still need to be addressed to secure voice authenticity as AI voice cloning technology continues to be refined.

References

1. "Truecaller insights 2022 U.S. Spam & Scam Report," Truecaller Blog. https://www.truecaller.com/blog/insights/truecaller-insights-2022-us-spam-scam-report. Accessed 11 Dec 2023
2. "Tax scams/consumer alerts," Internal Revenue Service. https://www.irs.gov/newsroom/tax-scams-consumer-alerts. Accessed 11 Dec 2023
3. Story, B.H.: History of speech synthesis, The Routledge Handbook of Phonetics, pp. 9–33, March 2019. https://doi.org/10.4324/9780429056253-2
4. Hu, W., Zhu, X.: A real-time voice cloning system with multiple algorithms for speech quality improvement. PLOS ONE **18**(4), e0283440–e0283440 (2023). https://doi.org/10.1371/journal.pone.0283440
5. Gozalo-Brizuela, R., Garrido-Merchán, E.C.: A survey of Generative AI Applications, arXiv.org, 14 June 2023. https://arxiv.org/abs/2306.02781
6. Masood, M., et al.: Deepfakes generation and detection: state-of-the-art, open challenges, countermeasures, and way forward. Appl. Intell. **53**(4) (2022). https://doi.org/10.1007/s10489-022-03766-z
7. Nacimiento-García, E., Nacimiento-García, A., Caballero-Gil, C., González González, C., Gutiérrez Vela, F.L.: Cybersecurity in Voice Virtual Assistants, presented at the The 21st International Conference on Security & Management, August 2022
8. Malik, H., Changalvala, R.: Fighting AI with AI: fake speech detection using deep learning, In: 2019 AES International Conference on Audio Forensics, June 2019
9. Ye, F., Yang, F.J.: A deep neural network model for speaker identification. Appl. Sci. **11**(8), 3603 (2021). https://doi.org/10.3390/app11083603
10. Zhang, C., et al.: A survey on audio diffusion models: Text to speech synthesis and enhancement in Generative AI, 2 April 2023. arXiv.org. https://arxiv.org/abs/2303.13336
11. Betker, J.: Better speech synthesis through scaling, 23 May 2023b. arXiv.org. https://arxiv.org/abs/2305.07243
12. Bird, J.J., Lotfi, A.: Real-time detection of AI-generated speech for Deepfake Voice conversion, 24 August 2023. arXiv.org. https://arxiv.org/abs/2308.12734
13. O'Shea, K., Nash, R.: An Introduction to Convolutional Neural Networks, November 2015. https://arxiv.org/abs/1511.08458
14. Dey, R., Salem, F.M.: Gate-variants of gated recurrent unit (GRU) neural networks. In: 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS), October 2017. https://doi.org/10.1109/mwscas.2017.8053243
15. Ito, K., Johnson, L.: The LJ speech dataset, Keith Ito. https://keithito.com/LJ-Speech-Dataset/. Accessed 11 Dec 2023
16. Mica, J.: AI Voice Cloning (Version 2.0) [Source Code] (2024). https://github.com/JarodMica/ai-voice-cloning
17. liujing04, Yuanwenyu, Ftps (2023) Retrieval-based-Voice-Coversion-WebUI (Version 2.0) [Source Code]. https://github.com/RVC-Project/Retrieval-based-Voice-Conversion-WebUI
18. Chen, C.: Real Time AI Voice Clone Detection (Version 2.0) [Source Code] (2024). https://github.com/Cody-not-Kody/Real-Time-AI-Voice-Clone-Detection

Author Queries

Chapter 14

Query Refs.	Details Required	Author's response
AQ1	Please check and confirm if the authors Given and Family names have been correctly identified.	
AQ2	This is to inform you that corresponding author has been identified as per the information available in the Copyright form.	
AQ3	References [16–18] are given in the list but not cited in the text. Please cite in text or delete from the list.	
AQ4	Please check and confirm if the inserted citation of Fig. 4 is correct. If not, please suggest an alternate citation.	