Hybrid Classical Quantum Learning Model Framework for Detection of Deepfake Audio

Atul Pandey[a] and Bhawana Rudra[b]
Department of Information Technology, National Institute of Technology, Srinivasnagar-575025, Mangalore, Dakshina Kannada, Karnataka, India
{*atulpandey.233it001, bhawanarudra*}*@nitk.edu.in*

Keywords: Quantum Machine Learning, Quantum Deep Learning, Quantum-Deepfake Audio, Hybrid Classical Quantum Model, Deepfake Audio.

Abstract: Artificial intelligence (AI) has simplified individual tasks compared to earlier times. However, it also enables the creation of fake images, audio, and videos that can be misused to tarnish the reputation of a person on social media. The rapid advancement of deepfake technology presents significant challenges in detecting such fabricated content. Therefore, in this paper, we particularly focus on the deepfake audio detection. Many Classical models exist to detect deepfake audio, but they often overlook critical audio features, and training these models can be computationally resource-intensive. To address this issue, we used a real-time AI-generated fake speech dataset, which includes all the necessary features required to train models and used Quantum Machine Learning (QML) techniques, which follow principles of quantum mechanics to process the data simultaneously. We propose a hybrid Classical-Quantum Learning Model that takes advantage of Classical and Quantum Machine Learning. The hybrid model is trained on a real-time AI-generated fake speech dataset, and we compare the performance with existing Classical and Quantum models in this area. Our results show that the hybrid Classical-Quantum model gives an accuracy of 98.81% than the Quantum Support vector Machine (QSVM) and Quantum Neural Network (QNN).

1 INTRODUCTION

Technological innovation continues to simplify human tasks, with one key advancement being Artificial Intelligence (AI), which enables people to work more efficiently and intelligently. AI can be used to generate images, videos, digital avatars, and even video dubbing (Nguyen et al., 2022). Unfortunately, this technology is sometimes exploited to tarnish the reputation of individuals by creating fake content, such as forged voices, images, and videos, using deepfake techniques—where deep learning models are employed to fabricate such content (Dagar and Vishwakarma, 2022). In a recent case, fraudsters utilized AI-driven software to imitate the voice of a company's CEO, successfully extorting USD 243,000 (The Wall Street Journal, 2019). As a result, there is growing interest in developing methods for detecting fraudulent voices (Khochare et al., 2021). People often rely on their knowledge and environmental awareness to identify fake audio. However, the rapid advancements in deepfake audio generation have underscored the importance of addressing these challenges. One specific type of deepfake audio is voice conversion, where the voice of a person is swapped with another (Yi et al., 2023).

Researchers commonly rely on Classical Deep-Learning models to identify deepfake content. However, training these models demands significant computational resources, even when using high-performance hardware like Graphics Processing Units (GPUs). While GPUs offer parallel processing capabilities, their performance is limited by the number of cores, leading to slower processing speeds than Quantum computers. Quantum computers leverage the principles of Quantum mechanics—such as superposition, entanglement, and interference, to process data much more efficiently and faster. These systems operate on Quantum bits (Qubits), each representing a superposition of Quantum states. Qubits provide exponential computational speed by simultaneously accessing multiple Quantum states. The processing power of the system is determined by the number of Qubits (denoted as N), with the ability to access 2^N states concurrently (Zaman et al., 2024).

[a] https://orcid.org/0009-0004-3106-4995
[b] https://orcid.org/0000-0001-7651-3820

Pandey, A. and Rudra, B.
Hybrid Classical Quantum Learning Model Framework for Detection of Deepfake Audio.
DOI: 10.5220/0013258700003899

In *Proceedings of the 11th International Conference on Information Systems Security and Privacy (ICISSP 2025) - Volume 2*, pages 231-239
ISBN: 978-989-758-735-1; ISSN: 2184-4356

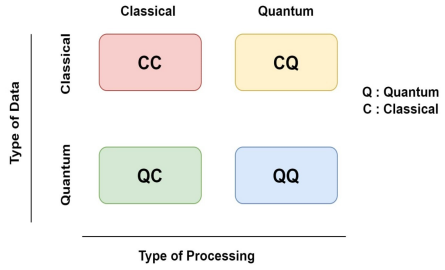

Figure 1: Four Sub-Categories of QML (Aïmeur et al., 2006).

Quantum Machine Learning (QML) offers a range of methods to address complex problems efficiently. Problems in QML can be approached using four distinct methods, as illustrated in figure 1. The four subcategories of the QML use the Quantum-inspired machine learning algorithm to process the Classical or Quantum data involved in the problem, and data processing is achieved on either a Classical or Quantum computer (Aïmeur et al., 2006). Each sub-categories are outlined below :

- **CC Approach**. Classical data is processed on a Classical computer using a Quantum-inspired machine learning algorithm.
- **CQ Approach**. Classical data is processed on a Quantum computer using a Quantum-inspired machine learning algorithm.
- **QC Approach**. Quantum data is processed on a Classical computer using a Quantum-inspired machine learning algorithm.
- **QQ Approach**. Quantum data is processed on a Quantum computer using a Quantum-inspired machine learning algorithm.

This paper is structured as follows: Section 2 discusses the existing Classical and Quantum models in the deepfake audio. This section also discusses the background of Quantum circuits. Section 3 provides a method to detect the deepfake audio and also discusses the proposed hybrid model. Section 4 provides the comparative analysis of both systems. Section 5 provides the insights of the implementation and the overview of the model performance. Section 6 provides a detailed analysis of the paper. Section 7 concludes the paper and discusses future work.

2 LITERATURE REVIEW

Several research papers have explored deepfake audio classification using machine learning and deep learning models. The author proposed a Convolutional Neural Network (CNN) based classifier, such as Light CNN, that filters noise in voice signals while preserving key information (Wu et al., 2018). Convolutional-Recurrent Neural Network (CRNN) based spoofing detection uses five 1D convolutional layers, a Long Short-Term Memory (LSTM) Layer, and two fully connected Layers to perform end-to-end detection of deepfake audio (Chintha et al., 2020). The authors (Khochare et al., 2021) proposed two approaches: one uses audio features for classification via machine learning models, while the other classifies using a temporal convolutional network and spatial transformer network based on images of the audio signals. While deep learning models achieve better results, they did not consider Short-Time Fourier Transform (STFT) and Mel-Frequency Cepstral Coefficients (MFCCs), which are two of the most effective features of the audio signal. The authors (Zhang et al., 2021) proposed a Squeeze and Excitation Network (SENet) that captures interdependencies between channels but requires more computational time for training. The authors (Hamza et al., 2022) presented a method for handling large datasets and classifying them using various machine learning algorithms. The Support Vector Machine (SVM) performed well on the For-rece and For-2-sec datasets, but for the For-norm dataset, gradient boosting generates better results. However, this work did not address fluctuations and distortions in the audio signals. The authors (Mcuba et al., 2023) extracted the various features from the fake audio file, such as MFCCs, Mel-Spectrum, Chromagram, and Spectrogram and converted them into images. The custom model and VGG model were trained on the audio features, and the results show that VGG performed well for the MFCCs feature and the custom model performed better for the rest of the features. The authors (Doan et al., 2023) proposed a breathing-talking-silence encoder to detect deepfake audio using ASVspoof 2019 and 2021 datasets. The results show that the performance of the classifier increased by 40%. The authors (Wu et al., 2024) proposed a deepfake detector based on contrastive Learning. This method minimized the variation in audio, which happened because of the manipulation of audio. This will increase the robustness of the model for the detection of deepfake audio. The author (Pham et al., 2024) used an ASVspoof 2019 benchmark dataset and extracted the Spectrogram from the audio. The CNN-based model, various pre-trained models and ensemble models were trained on the Spectrogram. The results show that the ensemble model performs better than the other models. The authors (Li et al., 2024) proposed a SafeEar framework to detect deepfake audio without relying on semantic content such that private content remains

secure in audio. They introduced the neural audio codec that separates the semantic and acoustic information, and they rely only on the acoustic information. The framework was tested on the 4 datasets and shows an error rate down to 2.02%, which made it suitable for anti-deepfake and anti-content recovery. However, the proposed method is limited to acoustic features, which makes it less effective against nuanced manipulations mimicking natural patterns. The authors (Saha et al., 2024) proposed a method to execute the machine learning and deep learning program for deepfake audio detection on the Central Processing Unit (CPU). This framework utilizes the self-supervised learning-based pre-trained model. The results show that the author achieved the 0.90% error rate with 1000 trainable parameters. Several papers focus on Classical models, and only a few have explored Quantum learning models for the deepfake image. Therefore, this paper focuses on the deepfake audio.

The authors (Mittal et al., 2020) proposed a method to detect fake images based on feature extraction using a Quantum-inspired evolutionary algorithm, though it lacks fine-tuning of parameters and noise filtering. The authors (Mishra and Samanta, 2022) introduced a Quantum-based transfer learning approach to detect deepfake images, where features are extracted from a pre-trained ResNet-18 model and classified using a Quantum Neural Network (QNN). The authors (Pandey and Rudra, 2024) proposed a method to detect deepfake audio speech using a Quantum Support Vector Machine (QSVM) and QNN. However, the performance of the model was not better compared to Classical.

The challenge to detect deepfake audio lies in recognizing features within the audio signal, whether they are genuine or fake. Literature shows that AI-based models are capable of effectively learning and predicting audio authenticity. These models are based on Classical Deep-Learning techniques. Many authors proposed the Quantum model to take Quantum advantage to overcome issues in a Classical computer. However, the Quantum model for detecting deepfake audio did not perform well compared to the Classical model. Therefore, to improve the model performance, we propose a hybrid Classical-Quantum learning model that takes advantage of Classical and Quantum Machine Learning.

2.1 Quantum Preliminaries

A Quantum circuit comprises two essential components: the feature map and the variational form. The feature map encodes Classical data into a Quantum state, while the variational form adjusts this Quantum state to the desired target state by iteratively tuning parameters.

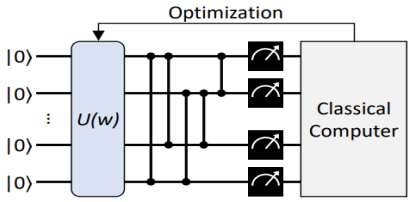

Figure 2: Parametrized Quantum Circuit (Benedetti et al., 2019).

- **Feature Maps.** There are several methods for embedding Classical data into Quantum states through feature mapping. Common techniques include Angle Embedding.
  - **Angle Embedding.** Angle Embedding is one of the simplest approaches for encoding floating-point data. It transforms a single floating-point value $x \in \mathbb{R}$ into a Quantum state using the following equation (1):

$$R_k(x)|0\rangle = e^{-ix\frac{\sigma_k}{2}}|0\rangle \tag{1}$$

    Here, $k \in \{x, y, z\}$ represents the rotation axis on the Bloch sphere, implemented through Pauli rotation gates. These rotations are applied to the data being encoded. In the case of Angle Embedding, the number of rotations corresponds to the number of features in the dataset (Schuld and Petruccione, 2018).
- **Parametrized Quantum Circuits (PQCs).** Variational Quantum Circuits (VQCs) or Parametrized Quantum Circuits (PQCs) are quantum algorithms by their reliance on free parameters. In QML, VQCs encode the Classical data into a Quantum state using the feature maps discussed in section 2.1 and then perform a variational form to create the QNN. The parameters used in the variational form are optimized through an iterative process. Measurement is performed on a Quantum circuit, which leads to stochastic output. We repeat the experiment multiple times to get the expectation value, and this will result in a probability distribution of the basis states. This probability distribution is given to the Classical algorithm to compute the loss function or cost function, which gives the difference between the predicted and true labels. These results are given to the Classical optimizer to update the parameters of the Quantum circuit to minimize the loss function. Figure 2 shows the working principle of the PQCs (Benedetti et al., 2019), which consists of feature mapping, variational forms and optimisation.

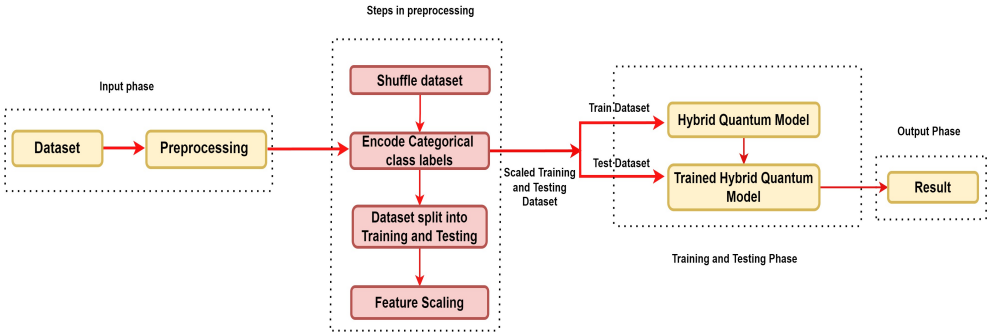

Figure 3: Proposed Methodology.

Algorithm 1: TwoLocal(n, k, θ).

Input: n, k, θ
for $r = 0$ *to* k **do**
  ▷ Add the r-th layer.;
  for $j = 1$ *to* n **do**
    Apply a $R_Y(\theta_{rj})$ gate on qubit j.;
  end
  ▷ Create entanglement between layers.;
  if $r < k$ **then**
    for $t = 1$ *to* $n-1$ **do**
      Apply a CNOT gate with control on qubit t and target on qubit $t+1$.;
    end
  end
end

– **Variational Form.** The variational form of a Quantum Neural Network (QNN) mimics the layered architecture of Classical Neural Networks. It relies on optimizable parameters $\vec{\theta}$ and introduces entanglement between Qubits through a parameter-independent circuit U_{ent}^{t}. Multiple Layers (or repetitions) can be stacked in the variational circuit. In our model, we employ a TwoLocal variational form which uses n Qubits and k repetitions, the total number of parameters needed for optimization is $n \times (k+1)$. These parameters, denoted as θ_{rj}, are indexed by r (from 0 to k) and j (from 1 to n). It will create k Layers not the $k+1$ Layers. Algorithm 1 defines the creation of the TwoLocal variational form (Elías Fernández, 2023).

- **Quantum Kernel.** Consider a Quantum model $f(x)$ defined as in equation (2):

$$f(x) = \langle \psi(x) | M | \psi(x) \rangle \tag{2}$$

In equation 2, $|\psi(x)\rangle$ is a Quantum state generated by an embedding circuit that encodes the input data x, and M is a chosen observable. $\langle \psi(x)|$ is the transpose of the $|\psi(x)\rangle$ i.e $\langle \psi(x)| = (|\psi(x)\rangle)^{\dagger}$.

This formulation encompasses variational QML models because the observable M can be realized through a simple measurement, which is preceded by a variational circuit. Instead of training the function f using variational methods, we can often achieve the same result by employing a Classical kernel method, where the kernel is computed on a Quantum device. The equation (3) shows the Quantum kernel determined by the overlap between two Quantum states encoding different data points:

$$\kappa(x, x') = |\langle \psi(x') | \psi(x) \rangle|^2 \tag{3}$$

By using this kernel-based approach, we avoid the need for processing and measuring the typical variational circuits, focusing solely on the data encoding (Schuld and Killoran, 2018).

3 METHODOLOGY

(Pandey and Rudra, 2024) proposed deepfake speech detection using Quantum models such as QSVM and QNN to compare the performance with Classical models such as Support Vector Machine (SVM) and Artificial Neural Networks (ANN). To improve the detection performance of the Quantum model, we proposed a hybrid Classical-Quantum model that takes advantage of Classical and Quantum Machine Learning. We also train the classical 1D CNN to compare it with the proposed model. A numerical dataset is utilized to train the models. The dataset contains features from the audio speech. Figure 3 illustrates the proposed methodology, which is broken down into three stages:

- **Input Phase.** In this stage, the dataset is taken as input and undergoes preprocessing.
- **Training and Testing Phase.** This stage takes the Classical data and encodes it into Quantum states using the embedding technique, and the variational form is applied to create the QNN (refer to

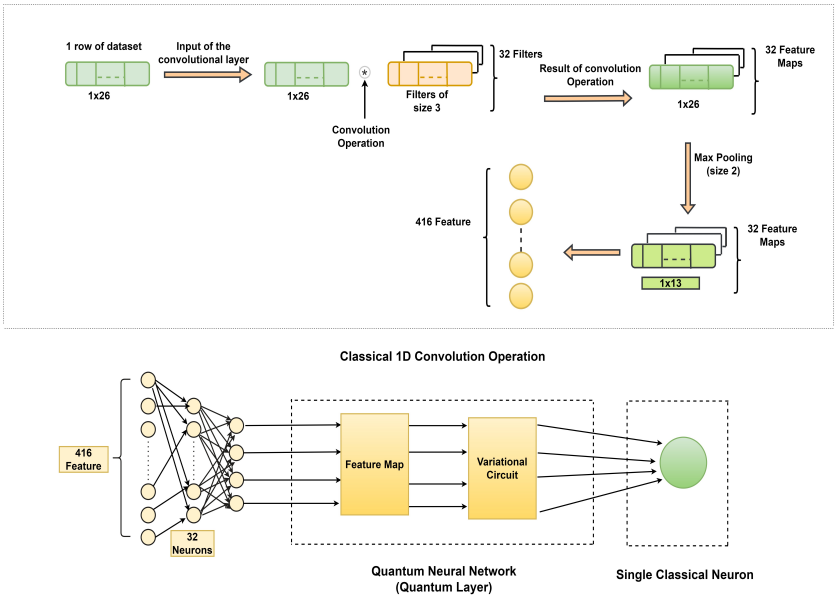

Figure 4: Hybrid Classical 1D Convolution Quantum Neural Network.

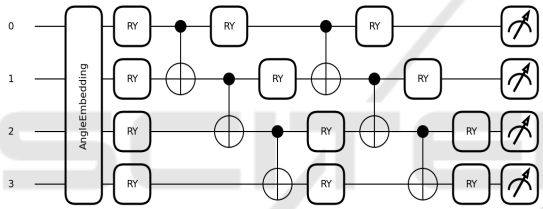

Figure 5: Quantum Layer Circuit of Hybrid Model (HC1CQNN).

section 2.1). The hybrid model is trained, and its performance is evaluated using the test dataset.

- **Output Phase.** This stage compares the results from both the Classical models, Quantum models, and the hybrid model.

3.1 Dataset

We consider the recent audio numerical dataset "Real-time detection of AI-Generated speech for Deepfake", published in 2023. J. J. Bird and A. Lotfi applied the two-stem model (Hennequin et al., 2020) from Spleeter to separate actual speech into natural vocals and accompaniment (background noise). The Spleeter model comprises 12 Layers organized into two sets of 6 Layers each for the encoder-decoder Convolutional Neural Network (CNN) within a U-Net architecture. Following this, the unprocessed vocals were converted into synthesized vocals of distinct individuals utilizing the Retrieval-Based Voice Conversion (RVC) model. Subsequently, the background noise and RVC-generated vocals were amalgamated to produce synthetic speech. The author has employed the Python-based Librosa library (McFee et al., 2015) to extract the 26 different features. The features include the chromagram (chromastft), spectral centroid, spectral bandwidth, spectral rolloff, root mean square (RMS) and twenty Mel-Frequency Cepstral Coefficients (MFCCs) (Bird and Lotfi, 2023).

3.2 Preprocessing

To ensure that the model does not become biased towards any particular class, the dataset should be shuffled to introduce randomness. It is divided into 80% for training and 20% for testing, which helps to prevent data leakage. Since some features in the dataset have varying ranges, feature scaling is applied to bring them to a uniform scale. This scaling improves the algorithm's convergence speed. The Min-Max scaling method normalises the features while preserving the original data range and enhancing interpretability. The scaler function is fitted on the training data and then applied to transform the test data, thereby avoiding data leakage during model evaluation.

3.3 Hybrid Classical Quantum Learning Model

Literature shows that CNN helps the model to learn necessary features from the data, which is extracted using the convolution operation. To take advantage

of the CNN, we propose a Hybrid Classical 1D Convolution Quantum Neural Network (HC1CQNN) as shown in figure 4. This hybrid model combines 4 Layers- 1D Convolution Layer, Classical Neurons Layer, Quantum Layer, and Single Classical Neuron Layer. The Classical 1D Convolution extracts the important feature from the audio dataset and feeds the features as input to the Classical Neurons. This will reduce the dimension of the data, which will be helpful for the Quantum Layer because we can not feed all the extracted features into the Quantum Layer because of the limitations of the current Quantum Simulators. The end Layer (Single Classical Neuron) of the hybrid model will classify the audio speech as real or fake. The hybrid model is trained as a single unit.

The hybrid model in figure 4 performs the convolution operation on the data of size 1x26 with 32 filters of size 3, which results in the 32 features map of size 1x26, and then applies the Max Pooling Layer (size-2) on the feature maps, which produces the 32 feature maps of size 1x13. Each feature map contains 13 features, which generates 416 features after flattening. 416 features cannot directly feed as input to the Quantum Layer because of the restriction of the qubit and the limitation of Quantum simulators. To handle this issue, we have applied 32 Classical Neurons followed by 4 Classical Neurons after flattening the features, and their output will be input to the Quantum Layer with 4 qubits to avoid system crash. The Quantum Layer converts the Classical data into a Quantum state using Angle embedding, which then learns the pattern in the data after embedding using the TwoLocal variational forms algorithm discussed in section 2.1 and then performs measurements on all the qubits. Figure 5 shows the Quantum Layer circuit diagram of HC1CQNN, which includes Angle embedding, TwoLocal variational forms and measurement of the circuit. The measurement result will be the input for the Single Classical Neuron, which later performs the classification of fake and real audio.

4 QUANTUM AND CLASSICAL SYSTEM ANALYSIS

This section analyzes the Quantum system Q and Classical system C to evaluate their computational performance in deep learning tasks. Let us assume that both systems take classical data x consisting of n bits as input.

$$x \leftarrow \text{Classical data}$$
$$n \leftarrow \text{Number of bits}$$

The system Q processes x using the PQC discussed in section 2.1. This system processes all 2^n states simultaneously. Let us assume that Q requires p units to process these 2^n states, from encoding to measurement. The measurement results are then fed into a classical optimizer, which adjusts the parameters used in Q. Assume the optimizer takes q units per optimization iteration. Thus, the PQC requires $(p+q)$ units for a single run. To reach the desired minimum loss, the PQC runs z times.

$$p \leftarrow \text{Units to process Q}$$
$$q \leftarrow \text{Units taken by the classical optimizer}$$
$$p+q \leftarrow \text{Units taken by PQC for a single run}$$
$$z \leftarrow \text{Times to run PQC}$$

The total processing time for system Q is:

$$T_Q = (p+q) \times z \tag{4}$$

Now, let's assume the Classical system C with GPU can handle m states concurrently. With 2^n states to process, C would need to perform approximately $(2^n/m)$ sequential processing steps. Each processing step requires r units, and its result is fed into the classical optimizer, which adjusts the parameters for C. Thus, system C requires $\frac{2^n}{m} \times (r+q)$ units for a single run. To achieve the minimum loss, system C also runs z times.

$$m \leftarrow \text{States handle by the GPU}$$
$$\frac{2^n}{m} \leftarrow \text{Sequential processing steps}$$
$$r \leftarrow \text{Units required per processing step}$$
$$\frac{2^n}{m} \times (r+q) \leftarrow \text{Units taken by system } C \text{ for a single run}$$
$$z \leftarrow \text{Times to run system } C$$

The total processing time for the Classical system C with GPU is:

$$T_C^{GPU} = \frac{2^n}{m} \times (r+q) \times z \tag{5}$$

For the Quantum system Q to outperform the Classical system C with GPU support, we require:

$$(p+q) \times z < \frac{2^n}{m} \times (r+q) \times z$$

Dividing by z (assuming $z \neq 0$) gives:

$$p+q < \frac{2^n}{m} \times (r+q)$$

Based on this analysis, we observe that the Quantum system Q maintains an advantage as n grows larger. As 2^n grows exponentially, $\frac{2^n}{m} \times (r+q)$ becomes large even with significant GPU parallelism. Thus, while adding GPU with Classical system C, it still does not eliminate the exponential scaling challenge faced by Classical processing.

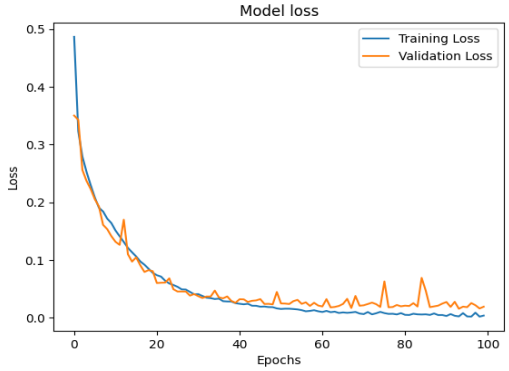

Figure 6: Loss Graph of 1D Convolution Neural Network During Training.

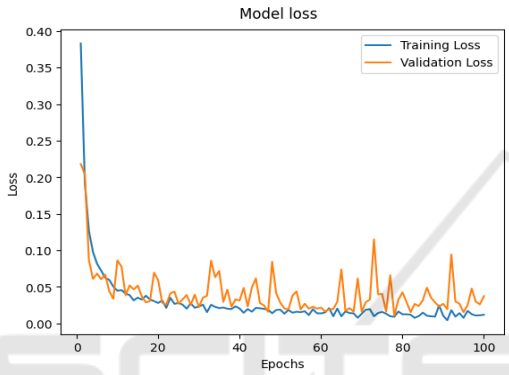

Figure 7: Loss Graph of Hybrid Classical 1D Convolution Quantum Neural Network During Training.

5 IMPLEMENTATION & RESULTS

The authors (Bird and Lotfi, 2023) created a .csv file comprising both real and fake voice samples. This file contains 11,778 data points (rows) and 26 features (columns). The authors (Pandey and Rudra, 2024) subsequently reduced the dataset to 2,000 data points and applied Principal Component Analysis (PCA) to reduce the feature count from 26 to 13 for training the Quantum models (QSVM and QNN) to prevent system crashes. We have utilized the entire dataset to enhance the performance of the Quantum model through a hybrid model approach. We employed an `NVIDIA RTX A4000` GPU to run the classical and hybrid models. Both the hybrid model and the Classical 1D CNN were trained on all 11,778 data points with 26 features to ensure a fair comparison of the models on the same scale. The Classical 1D CNN and the hybrid model were implemented using TensorFlow version 2.15.0. For the hybrid model, we used Pennylane (Bergholm et al., 2022) version 0.36.0 to encode the classical data and construct the Quantum Layer using Pennylane's TensorFlow interface. The hybrid model training was conducted on the `lightning qubit simulator` provided by Pennylane, which offers efficient linear algebra computation and differentiation methods to train the hybrid model effectively. These simulators use Quantum algorithms to leverage Quantum properties to execute QML programs. However, at the hardware level, simulators run on Classical computers, which may require several days or even weeks to complete QML tasks, resulting in more execution time and sometimes system crashes. Table 1 presents the classification metrics for both Classical (SVM and ANN) and Quantum (QSVM and QNN) models, as discussed (Pandey and Rudra, 2024). 90.02%, 95.97%, 83.50% and 89.30% are the accuracy, precision, recall and f1-score respectively represent the performance metric of the QSVM. 70.07%, 72.47%, 64.50% and 68.25% are the accuracy, precision, recall, and f1-score respectively represent the performance metric of the QNN. These results indicate that the reduced dataset and simulator limitation leads to performance degradation for QSVM and QNN. Our hybrid model implementation overcomes these issues and yields improved results. Table 2 provides the classification metrics for the models—1D CNN and hybrid model (HC1CQNN) on the test dataset. We obtain the training loss graph as shown in figure 6 and 7. We observe that all the parameters range from 98-99% for both 1D CNN and hybrid model (HC1CQNN). we observe that the hybrid model is trained perfectly and is almost similar to the training loss graph of the Classical 1D CNN model. This shows that the hybrid model has improved its performance over the Quantum model.

6 DISCUSSION

In this study, we evaluate both Classical and Quantum Machine Learning for deepfake audio detection. To leverage the Quantum advantage, the author (Pandey and Rudra, 2024) uses the Quantum models to perform deepfake audio detection. However, the performance of the Quantum model lags behind that of the classical model, as in table 1. This likely happened because of the use of fewer features and data (rows) from dataset (Bird and Lotfi, 2023) as well as the limitation of simulators. Therefore to improve the detection performance, we have proposed the hybrid model. This model leverages the advantage of the Classical and Quantum models. The results in table 2 show that the hybrid model achieved almost similar results compared to Classical 1D CNN. This indicates that the performance of the Quantum model

Table 1: Classification Metric Results of Classical and Quantum Model.

Model Type	**Models**	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-score (%)**
Classical Model	Classical SVM (Pandey and Rudra, 2024)	83.79	86.43	81.90	84.10
	Classical ANN (Pandey and Rudra, 2024)	95.00	96.27	93.29	94.76
Quantum Model	QSVM (Pandey and Rudra, 2024)	90.02	95.97	83.50	89.30
	QNN (Pandey and Rudra, 2024)	70.07	72.47	64.50	68.25

Table 2: Classification Metric Results of Classical and Hybrid Model.

Model Type	**Models**	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-score (%)**
Classical Model	Classical 1D CNN	99.19	98.89	99.48	99.19
Quantum Model	Hybrid Model (HC1CQNN)	98.81	98.39	99.23	98.80

(QSVM and QNN) can be enhanced through a hybrid approach. Our analysis in section 4 suggests that Quantum systems offer faster computation than Classical systems. However, in practice, this advantage is not purely realized due to the current limitations of Quantum simulators.

7 CONCLUSION AND FUTURE WORK

This paper demonstrates the application of Quantum models for deepfake audio detection, utilizing the computational advantages offered by Quantum processing. Quantum approaches were considered, as Classical computers encounter significant computational challenges, particularly with the extensive resources required for training deep learning models. However, literature shows that Quantum models are not up to when compared with Classical models due to the use of fewer features and data from the dataset as well as the limitation of Quantum simulators. Therefore, to improve the performance of Quantum models (QSVM and QNN), we propose a hybrid approach that leverages the strengths of both Classical and Quantum models. The results indicate that the hybrid model performs almost similar to the Classical 1D CNN model. Our analysis shows that Quantum systems have the potential to perform faster computations than Classical systems. However, this advantage remains constrained in practice due to the limitations of current Quantum simulators. Deploying deepfake audio detection using the Quantum model effectively requires large datasets and improved Quantum simulators to prevent systems from crashing. With the existing technology, our hybrid model demonstrates improved performance with a combination of Classical and Quantum Machine Learning techniques. However, achieving optimal performance with purely Quantum models will require further development in Quantum simulators. In the future, we will explore a hybrid Classical-Quantum Model approach for other areas of deepfake detection, such as video deepfake detection.

REFERENCES

Aïmeur, E., Brassard, G., and Gambs, S. (2006). Machine learning in a quantum world. In Lamontagne, L. and Marchand, M., editors, *Advances in Artificial Intelligence*, pages 431–442, Berlin, Heidelberg. Springer Berlin Heidelberg.

Benedetti, M., Lloyd, E., Sack, S., and Fiorentini, M. (2019). Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4(4):043001.

Bergholm, V., Izaac, J., Schuld, M., Gogolin, C., Ahmed,

S., and *et.al.*, V. A. (2022). Pennylane: Automatic differentiation of hybrid quantum-classical computations.

Bird, J. J. and Lotfi, A. (2023). Real-time detection of ai-generated speech for deepfake voice conversion.

Chintha, A., Thai, B., and *et.al.*, S. (2020). Recurrent convolutional structures for audio spoof and video deepfake detection. *IEEE Journal of Selected Topics in Signal Processing*, 14:1024–1037.

Dagar, D. and Vishwakarma, D. (2022). A literature review and perspectives in deepfakes: generation, detection, and applications. *International Journal of Multimedia Information Retrieval*, 11.

Doan, T.-P., Nguyen-Vu, L., Jung, S., and Hong, K. (2023). Bts-e: Audio deepfake detection using breathing-talking-silence encoder. *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5.

Elías Fernández, Combarro Álvarez, S. G. C. (2023). *A Practical Guide to Quantum Machine Learning and Quantum Optimization*. Packt, 1st edition.

Hamza, A., Javed, A. R. R., Iqbal, F., Kryvinska, N., Almadhor, A. S., Jalil, Z., and Borghol, R. t. (2022). Deepfake audio detection via mfcc features using machine learning. *IEEE Access*, 10:134018–134028.

Hennequin, R., Khlif, A., Voituret, F., and Moussallam, M. (2020). Spleeter: a fast and efficient music source separation tool with pre-trained models. *Journal of Open Source Software*, 5(50):2154.

Khochare, J., Joshi, C., Yenarkar, B., Suratkar, S., and Kazi, F. t. (2021). A deep learning framework for audio deepfake detection. *Arabian Journal for Science and Engineering*, 47.

Li, X., Li, K., Zheng, Y., Yan, C., Ji, X., and *et.al*, W. X. (2024). Safeear: Content privacy-preserving audio deepfake detection.

McFee, B., Raffel, C., and *et.al.*, L. (2015). librosa: Audio and music signal analysis in python. In *Proceedings of the 14th python in science conference*, volume 8, pages 18–25.

Mcuba, M., Singh, A., Ikuesan, R. A., and *et.al.*, H. V. (2023). The effect of deep learning methods on deepfake audio detection for digital investigation. *Procedia Computer Science*, 219:211–219. CENTERIS – International Conference on ENTERprise Information Systems / ProjMAN – International Conference on Project MANagement / HCist – International Conference on Health and Social Care Information Systems and Technologies 2022.

Mishra, B. and Samanta, A. (2022). Quantum transfer learning approach for deepfake detection. *Sparklinglight Transactions on Artificial Intelligence and Quantum Computing*.

Mittal, H., Saraswat, M., Bansal, J. C., and Nagar, A. (2020). Fake-face image classification using improved quantum-inspired evolutionary-based feature selection method. In *2020 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 989–995.

Nguyen, T. T., Nguyen, Q. V. H., Nguyen, D. T., Nguyen, D. T., Huynh-The, T., Nahavandi, S., Nguyen, T. T., Pham, Q.-V., and Nguyen, C. M. t. (2022). Deep learning for deepfakes creation and detection: A survey. *Computer Vision and Image Understanding*, 223:103525.

Pandey, A. and Rudra, B. (2024). Deepfake audio detection using quantum learning models. In *Proceedings of the IEEE Middle East Conference on Communications and Networking*.

Pham, L., Lam, P., Nguyen, T., Nguyen, H., and Schindler, A. (2024). Deepfake audio detection using spectrogram-based feature and ensemble of deep learning models.

Saha, S., Sahidullah, M., and Das, S. (2024). Exploring green ai for audio deepfake detection.

Schuld, M. and Killoran, N. (2018). Quantum machine learning in feature hilbert spaces. *Physical review letters*, 122 4:040504.

Schuld, M. and Petruccione, F. (2018). *Supervised Learning with Quantum Computers*. Springer Publishing Company, Incorporated, 1st edition.

The Wall Street Journal (2019). Fraudsters use ai to mimic ceo's voice in unusual cybercrime case. Accessed on May 28, 2024.

Wu, H., Chen, J., Du, R., Wu, C., He, K., Shang, X., Ren, H., and Xu, G. (2024). Clad: Robust audio deepfake detection against manipulation attacks with contrastive learning.

Wu, X., He, R., and *et.al.*, S. (2018). A light cnn for deep face representation with noisy labels. *IEEE Transactions on Information Forensics and Security*, 13(11):2884–2896.

Yi, J., Wang, C., Tao, J., Zhang, X., Zhang, C. Y., and *et.al*, Y. Z. (2023). Audio deepfake detection: A survey.

Zaman, K., Marchisio, A., Hanif, M. A., and *et.al.*, M. S. (2024). A survey on quantum machine learning: Current trends, challenges, opportunities, and the road ahead.

Zhang, Y., Wang, W., and *et.al.*, P. Z. (2021). The effect of silence and dual-band fusion in anti-spoofing system. In *Interspeech*.