Quantum Transfer Learning for Cross-Domain Cybersecurity Threat Detection and Categorization

Shtwai Alsubai[1], Mohamed Ayari[2*], Natalia Kryvinska[3], Ahmad Almadhor[4], Jamel Baili[5], Abdullah Al Hejaili[6], Sidra Abbas[7*]

[1]College of Computer Engineering and Sciences, Prince Sattam bin Abdulaziz University, AlKharj 16273, Saudi Arabia.
[2]Department of Information Technology, Faculty of Computing and Information Technology, Northern Border University, Saudi Arabia.
[3]Department of Information Management and Business Systems, Faculty of Management, Comenius University Bratislava, Odbojarov 10, 82005 Bratislava 25, Slovakia.
[4]Department of Computer Engineering and Networks, College of Computer and Information Sciences, Jouf University, Sakaka 72388, Saudi Arabia.
[5]Department of Computer Engineering, College of Computer Science, King Khalid University, Abha 61413, Saudi Arabia.
[6]Faculty of Computers & Information Technology, Computer Science Department, University of Tabuk, Tabuk 71491, Saudi Arabia.
[7*]Department of Computer Science, COMSATS University Islamabad, Sahiwal, Pakistan.

*Corresponding author(s). E-mail(s): mohamed.ayari@nbu.edu.sa; sidraabbas@ieee.org;
Contributing authors: sa.alsubai@psau.edu.sa; natalia.kryvinska@uniba.sk; aaalmadhor@ju.edu.sa; Jabaili@kku.edu.sa; a.alhejaili@ut.edu.sa;

Abstract

Cybersecurity challenges have become increasingly complex and widespread, and the risks associated with these problems are substantial, affecting thousands of individuals and organisations and being crucial to national security.

As cybercriminals have become increasingly adept at utilizing advanced methods to exploit system vulnerabilities, there has never been a more pressing need for reliable threat detection and response systems. This study proposes a framework that uses quantum transfer learning to enhance cybersecurity threat detection by leveraging multiple datasets, including UNSW-NB15, CICIDS2017, CSE-CIC-IDS2018, and TON_IoT. The framework focuses on improving the accuracy and efficiency of existing machine learning methods for cyber threat detection by employing quantum computing techniques for feature extraction and analysis. The pre-processing of the UNSW-NB15 dataset, the extraction of quantum features using PennyLane, and the training of the deep learning model with TensorFlow are the steps in the workflow of this study. Finally, the model is fine-tuned through transfer learning on other datasets, resulting in improvements in detection accuracy. This study shows that our quantum-enhanced model attains an accuracy of 83% on UNSW-NB15, 91% on the combined CICIDS2017 and CSE-CIC-IDS2018 datasets, and 86% on the TON_IoT dataset, demonstrating the potential of quantum computing and its use in the field of cybersecurity. Unlike fully quantum classifiers, our approach applies quantum transformations only at the feature-extraction stage, thereby creating a hybrid classical-quantum workflow that enhances transfer-learning performance across multiple cybersecurity datasets.

Keywords: Quantum Transfer Learning, Cybersecurity, Threat Detection, UNSW-NB15, CICIDS2017, CSE-CIC-IDS2018, TON_IoT, Quantum Feature Extraction, Deep Learning, Imbalanced Data.

1 Introduction

In the digital age, cybersecurity has become a critical issue due to the rapid expansion of interconnected systems and digital infrastructures, which has increased the attack surface and enabled sophisticated threats such as malware, phishing, Advanced Persistent Threats (APTs), and distributed denial-of-service (DDoS) attacks [1–3]. These incidents can result in severe consequences, including financial losses, data breaches, and reputational damage [4, 5]. Recent reports estimate that global Cybercrime costs may reach $10.5 trillion annually by 2025 [6], highlighting the urgent need for effective and scalable cybersecurity solutions. Additionally, the growing number of Internet-connected devices, projected to exceed 30 billion by 2030 [7], further challenges traditional centralized security mechanisms.

Traditional cybersecurity solutions, including rule-based intrusion detection systems (IDS) and signature-based methods, have long been used to mitigate cyber threats [8]. However, their reliance on static rules and known attack signatures limits their effectiveness against zero-day attacks, evolving adversaries, and highly imbalanced network traffic [9, 10]. To address these limitations, artificial intelligence (AI) and machine learning (ML) techniques have been widely adopted in modern IDS, enabling automated, adaptive, and data-driven threat detection [11, 12]. Despite these

advances, classical AI-based models often struggle with high-dimensional cybersecurity datasets, heterogeneous feature spaces, and generalization across different network environments [13, 14].

Quantum computing has recently emerged as a promising paradigm for addressing these challenges by exploiting quantum-mechanical principles such as superposition and entanglement. In particular, quantum machine learning (QML) techniques have the potential to encode complex data structures into high-dimensional Hilbert spaces, enabling richer feature representations. Within this context, quantum transfer learning (QTL) has gained attention as an approach that combines classical deep learning with quantum-enhanced feature extraction, offering a practical pathway for deploying quantum models in the noisy intermediate-scale quantum (NISQ) era. However, existing quantum-enhanced intrusion detection systems typically employ quantum models only at the classification stage or rely on standalone quantum architectures, while transfer learning-based IDS remain limited to classical-to-classical knowledge reuse. In contrast, the proposed approach fundamentally differs by enabling structured classical-to-quantum transfer learning, where pretrained classical feature representations are transferred into variational quantum circuits for quantum-enhanced feature transformation and cross-dataset intrusion detection. Quantum feature extraction is expected to outperform classical methods because variational quantum circuits can encode complex, high-dimensional correlations among features using superposition and entanglement. This enables richer and more expressive representations of cybersecurity data, potentially improving separability between normal and malicious traffic and enhancing generalization across heterogeneous datasets.

Quantum Transfer Learning (QTL) for cybersecurity threat detection is illustrated in Fig. 1. The proposed workflow consists of three key stages. First, a classical neural network is pre-trained on source-domain cybersecurity data to learn robust feature representations. Second, selected features are encoded into a variational quantum circuit, where quantum entanglement and parameterized gates are used to transform the data. Finally, the quantum-enhanced features are integrated with a classical classifier to perform intrusion detection on target-domain datasets. This hybrid quantum–classical design enables knowledge transfer across datasets while leveraging the representational capabilities of quantum circuits.

This study explores quantum transfer learning for intrusion detection by applying quantum-enhanced feature extraction and transfer learning across multiple benchmark cybersecurity datasets, including UNSW-NB15, CICIDS2017, CSE-CIC-IDS2018, and TON_IoT. Quantum transformations are implemented in PennyLane using AngleEmbedding and BasicEntanglerLayers, whereas classical deep learning models are developed in TensorFlow. By pre-training on UNSW-NB15 and fine-tuning on other datasets, the proposed framework evaluates the adaptability and generalization capability of quantum transfer learning across heterogeneous cybersecurity environments.

The objective of this research is to investigate the feasibility and potential benefits of integrating quantum transfer learning into intrusion detection systems, thereby bridging the gap between classical machine learning limitations and the emerging capabilities of quantum computing.

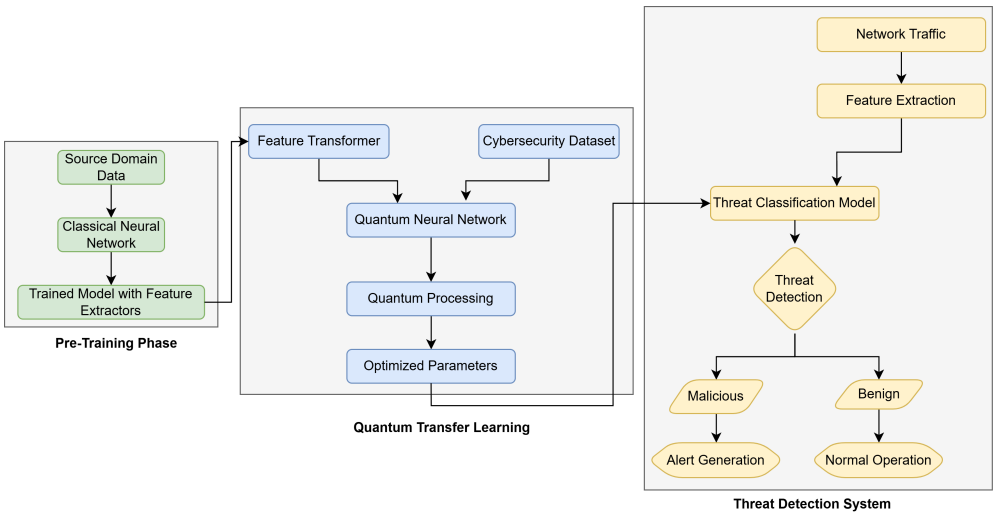


Fig. 1: Overview of Quantum Transfer Learning approach

The major contributions of this research are as follows:

- This study proposes a novel hybrid quantum transfer learning framework that integrates classical pre-trained models with variational quantum circuits for intrusion detection.
- Progressive transfer learning is demonstrated across multiple benchmark cybersecurity datasets (UNSW-NB15 → CICIDS2017/CSE-CIC-IDS2018 → TON_IoT), improving model generalization without full retraining. A standardized preprocessing strategy is introduced to align heterogeneous cybersecurity datasets, enabling effective transfer learning across diverse feature spaces and attack scenarios.
- Experimental results show that the proposed quantum-enhanced model achieves competitive performance, including an accuracy of 83% on UNSW-NB15 and up to 91% on CICIDS2017 and CSE-CIC-IDS2018, demonstrating the feasibility of quantum transfer learning for cybersecurity applications.

The remainder of this paper is organized as follows. Section 2 presents the related work on intrusion detection systems and quantum computing applications. Section 3 introduces the proposed quantum transfer learning framework. Section 4 describes the experimental setup, evaluation, and results. Section 5 presents a detailed discussion of the findings. Finally, Section 6 concludes the paper and outlines directions for future work.

2 Related Work

This section reviews existing research on intrusion detection systems, with a focus on deep learning approaches and their effectiveness in identifying cyber threats.

Thaljaoui et al. [15] proposed a hybrid model combining Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks to detect cyberattacks in the IoT environment. Bayesian optimization was used to optimize the detection performance of the model based on its hyperparameters. The model's effectiveness was assessed by training and testing it on the UNSW-NB15 dataset. Using key performance metrics such as accuracy, precision, recall, and F1 score, the system was further evaluated and found to perform intrusion detection. Amiri et al. [16] introduced FocalCA, a hybrid convolutional-attention encoder model designed to detect cyber threats using the UNSW-NB15 dataset. FocalCA achieved high accuracy without requiring data oversampling or data balancing, preserving the dataset's original trait properties. To effectively capture both global and local traffic patterns, the model uses a feature tokeniser, an attention mechanism as an encoder, a CNN layer, and a fully connected layer. The model addressed class imbalance using weighted sampling and focal loss, achieving a test accuracy of 99.47% and strong precision, recall, and F1 scores across all attack classes.

Hassanin et al. [17] proposed a pre-trained Large Language Model for Cyber Security (PLLM-CS) to enhance the detection and analysis of security threats. This Transformer-based method introduces a custom module that transforms raw network data into structured contextual inputs. Model-based encoding of cyber-specific information became more effective as a result of this transformation. Finally, the method's performance was evaluated on two publicly available IoT-based traffic datasets, UNSW-NB15 and TON_IoT. Experimental results showed that the performance of PLLM-CS outperformed BiLSTM, GRU, CNN, and other state-of-the-art models in the intrusion detection tasks. Shoukat et al. [18] presented XDLTDS, an Explainable Artificial Intelligence (XAI)-based deep learning framework designed to enable transparent and trusted cyber threat detection. Within the framework, the encoded Industrial IoT (IIoT) data was processed by a Long Short-Term Memory Autoencoder (LSTM-AE) to mitigate inference attacks. Multi-class threat classification in the IIoT network was also conducted using an Attention Recurrent Unit (AGRU) with a softmax layer. To make it interpretable for security analysts, the Shapley Additive Explanations (SHAP) method was employed as an additional remedy to the black-box nature of deep learning models. To improve network security, the framework was deployed in an architecture based on Software-Defined Networking (SDN). It is experimentally validated on the N-BaIoT, Edge-IIoTset, and CIC-IDS2017 datasets and demonstrates superior performance in detecting cyber threats in industrial networks compared with other frameworks.

Abdelaziz et al. [19] investigated the effectiveness of machine learning techniques for enhancing Network Intrusion Detection System (NIDS) performance. The study performed data quality enhancement and removed redundancy in the CICIDS-2017 dataset. Feature selection and permutation importance techniques were used to select the most relevant features for intrusion detection. Across multiple attack categories, the Random Forest classifier performed best among conventional machine learning intrusion detection techniques, achieving a 99.8% weighted F1 score and a 93.31% macro F1 score improvement with class-weighting and a custom prediction function. Luqman et al. [20] proposed an intelligent intrusion detection system for IoT

networks that utilizes hybrid deep learning architectures. Random Forest (RF) and Support Vector Machine (SVM) were applied to detect intrusions in the BoT-IoT dataset, achieving 99.60% accuracy for binary and 98.31% for multi-class classification. To address the class imbalance, UNSW BoT-IoT and UNSW-NB15 datasets were merged through feature engineering and used to train a Long Short-Term Memory (LSTM) model. After pre-processing, feature extraction, and multiple training iterations with 10 and 16 epochs, the LSTM-based approach achieved outstanding accuracy rates of 99.89% and 99.97% for multi-class classification, demonstrating its effectiveness in IoT network intrusion detection. Thiagarajan et al. [21] presented an Enhanced Convolutional Neural Network (ECNN) model for anomaly detection that integrates deep learning with advanced clustering and optimization techniques. Using the UNSW-NB15 dataset, KMeans++ clustering was applied, dimensionality was reduced with a deep autoencoder, and classification was performed using an ECNN containing convolutional, pooling, and fully connected layers. To compensate for class imbalance, cross-entropy or targeted loss functions were employed, and hyperparameters were optimized using a Genetic Algorithm. The model was evaluated using anomaly-detection metrics, achieving an F1 score of 99.75%.

Rawat et al. [22] conducted a scientometric analysis of quantum computing research, mapping its evolution from theoretical foundations to practical realization. The study analyzed publication trends, citation patterns, collaboration networks, and thematic structures through author co-citation and keyword co-occurrence analyses, providing insights into emerging research directions in QC. However, the work remains at a high level and does not address application-specific challenges. Similarly, Rawat et al. [23] presented a domain-oriented scientometric and visual analytics study of quantum computing applications across seven scientific fields, using 9,919 Scopus-indexed articles from 1996 to 2025. By analysing publication trends, collaboration patterns, and thematic relationships, the study identified key research hotspots and emerging frontiers. Despite its broad scope, the study is primarily descriptive and does not examine the integration of quantum computing techniques into domain-specific applications such as cybersecurity or intrusion detection. In addition to traditional intrusion detection research, recent studies in related cybersecurity domains highlighted the need for cross-domain adaptability. Sahu et al. [24] emphasized the necessity of secure and interoperable data exchange within telemedicine platforms to manage diverse healthcare datasets effectively. Similarly, Sahu et al. [25] focused on securing decentralized finance (DeFi) frameworks by protecting smart contracts and decentralized identities across heterogeneous financial ecosystems. To address digital integrity, Kumar et al. [26] highlighted strategies for mitigating deepfake disinformation through advanced detection algorithms and secure architectures. While these studies originate from diverse domains and do not directly address intrusion detection, they collectively illustrate the persistent challenges posed by artifact-induced noise, domain-specific data characteristics, and rapidly evolving threat patterns. These cross-disciplinary insights provide strong motivation for the proposed quantum transfer learning framework for robust threat detection in cybersecurity.

3 Proposed Framework

This section explains the proposed workflow, which begins with training on the UNSW-NB15 dataset. Data pre-processing includes removing irrelevant columns, encoding categorical features, scaling numerical values, and applying PCA for dimensionality reduction. Quantum feature extraction is performed using PennyLane's AngleEmbedding and BasicEntanglerLayers. A deep learning model with dense layers and ReLU activation is trained using TensorFlow, incorporating class weighting and early stopping. For transfer learning, the model is fine-tuned on CICIDS2017 and CSE-CIC-IDS2018 datasets after feature mapping and PCA-based quantum encoding. Finally, the model is adapted to the TON_IoT dataset by aligning feature structures and applying the same transformation and fine-tuning approach. Fig. 2 illustrates the proposed framework for Quantum transfer learning and the pre-processing steps across all three Datasets.

Algorithm 1 presents a three-phase quantum transfer learning (QTL) approach for cybersecurity threat detection across various network datasets. In Phase 1, PCA, feature selection, encoding, scaling, and dimensionality reduction are performed on the UNSW NB15 dataset. In particular, AngleEmbedding and entanglement layers are applied to perform quantum feature extraction through a parameterized quantum circuit. Training is performed using a deep learning model with ReLU activations and softmax output, employing class balancing and early stopping to mitigate the risk of poor generalization. In Phase 2, UNSW-NB15 and CSE-CIC-IDS2018 are used to create a dataset with the same formatting as UNSW-NB15, ensuring that features are compatible across the datasets. Then, the model is fine-tuned on the new set with the learned weights from Phase 1 after applying PCA and quantum encoding. This knowledge is transferred from the model to capture changing network traffic patterns. In Phase 3, the TON_IoT dataset is also pre-processed, and the features are aligned with the UNSW structure. The Phase 2 model is further fine-tuned after the quantum feature transformation (QFT). The last phase of our process utilizes the cumulative knowledge from both the previously mentioned datasets to improve anomaly detection in the IoT-based environment. In Phase 3, the generalization and detection performance of the trained model (M_f) improves the transfer learning performance in handling heterogeneous cybersecurity data, and the final detection performance is improved on disparate weaponized threat landscapes.

3.1 Training on the UNSW-NB15 Dataset

This subsection describes the workflow for training on the UNSW-NB15 Dataset.

3.1.1 Dataset Description

The UNSW-NB15 dataset was downloaded from Kaggle https://www.kaggle.com/datasets/mrwellsdavid/unsw-nb15/. This dataset contains a variety of network traffic data, including normal and malicious activities. It was created by the Australian Centre for Cyber Security (ACCS) and contains more than 2.5 million records, with features extracted from real-world network traffic. The dataset comprises nine attack

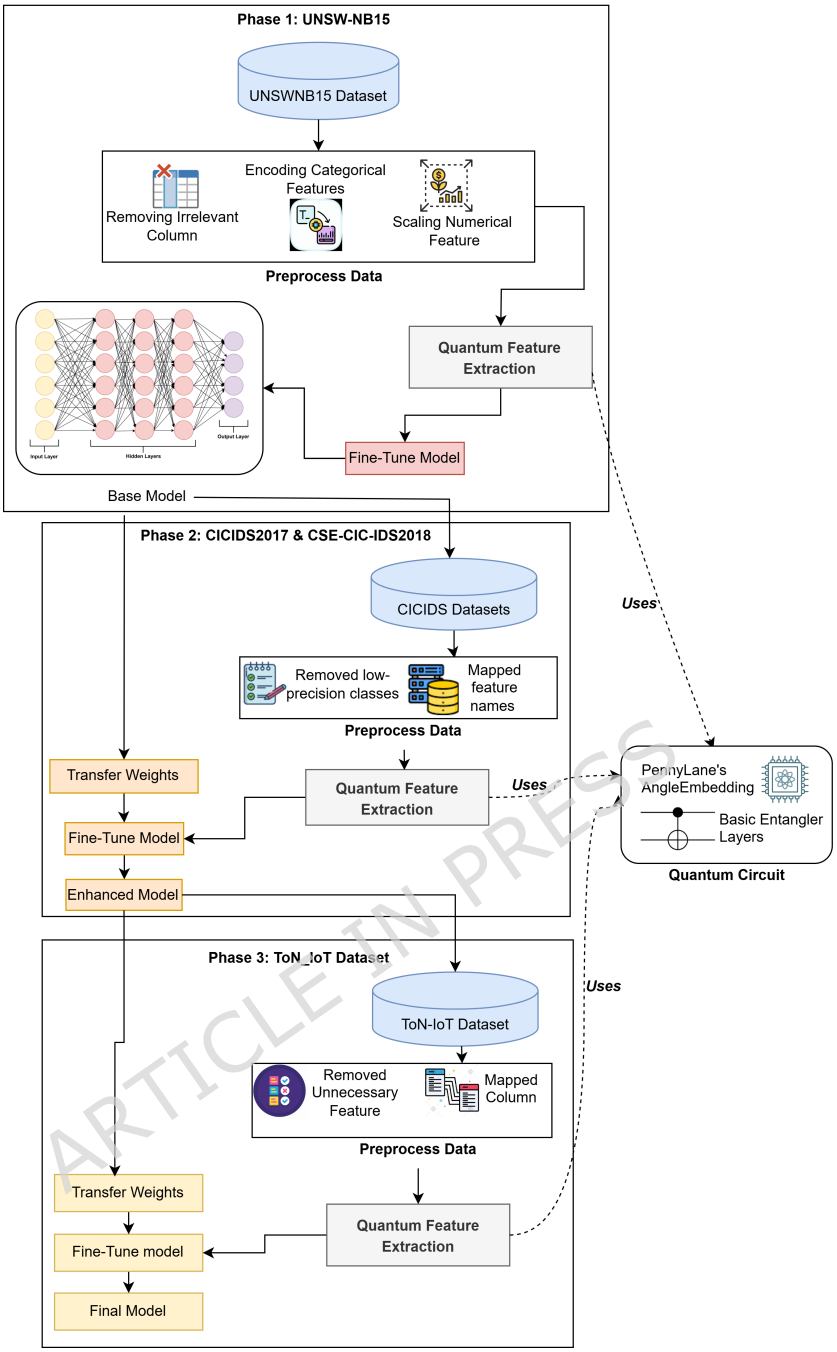


Fig. 2: Proposed Framework For Training on all Datasets

Algorithm 1 Quantum Transfer Learning for Cybersecurity Threat Detection

Require: Datasets: D_{UNSW}, D_{CICIDS}, D_{TON_IoT}

function PREPROCESS(D, $refCols$)
    Remove irrelevant columns using $refCols$
    Encode categorical, scale numerical features
    Apply PCA for dimensionality reduction
    return preprocessed data D'
end function
function QUANTUMEXTRACT(D')
    Initialize quantum circuit with n qubits
    Encode data using AngleEmbedding
    Apply entanglement layers (e.g., BasicEntanglerLayers)
    Measure and extract quantum features $Q(D')$
    return $Q(D')$
end function
function TRAINMODEL(Q, y, w)
    if w is null **then**
        Initialize neural model M with random weights
    else
        Initialize M with weights w
    end if
    Configure input, hidden (ReLU), and softmax output layers
    Handle imbalance, apply early stopping
    Train and **return** M
end function
procedure QTL_CYBERSECURITY
    Phase 1: Train on UNSW
    $D'_1 \leftarrow$ PREPROCESS(D_{UNSW}, null)
    $Q_1 \leftarrow$ QUANTUMEXTRACT(D'_1)
    $M_1 \leftarrow$ TRAINMODEL(Q_1, y_1, null)
    Phase 2: Transfer to CICIDS
    $D'_2 \leftarrow$ PREPROCESS(D_{CICIDS}, cols(D_{UNSW}))
    $Q_2 \leftarrow$ QUANTUMEXTRACT(D'_2)
    $M_2 \leftarrow$ TRAINMODEL(Q_2, y_2, weights(M_1))
    Phase 3: Transfer to TON_IoT
    $D'_3 \leftarrow$ PREPROCESS(D_{TON_IoT}, cols(D_{UNSW}))
    $Q_3 \leftarrow$ QUANTUMEXTRACT(D'_3)
    $M_f \leftarrow$ TRAINMODEL(Q_3, y_3, weights(M_2))
    return M_f
end procedure

types, including DoS, Shellcode, and Exploits, making it a comprehensive benchmark for intrusion detection.

This work improves the machine learning workflow by addressing class imbalance, optimizing training, and refining evaluation. It computes class weights to ensure

the model focuses on underrepresented classes. Class weights were computed separately for each dataset used in the experiments, adapting to each dataset's class distribution to maintain fair representation. An early-stopping callback in TensorFlow's Keras API prevents overfitting by halting training after five epochs without a decrease in validation loss. The quantum circuit is updated to support weighted transformations, enhancing feature extraction with BasicEntanglerLayers. A refined deep learning model, comprising a dense layer with 64 units, ReLU activation, dropout, and softmax output, is trained with the Adam optimizer (learning rate 0.001) for up to 100 epochs (batch size 16), using class weights and early stopping. Performance is assessed via a classification report and confusion matrix. Metrics, ensuring a clear evaluation of strengths and weaknesses in network traffic classification. While balancing class weights helped improve the model's attention to underrepresented classes, further analysis revealed that certain attack categories contributed little to classification performance. To enhance model generalization and reduce noise, some low-impact classes, such as 'Analysis', 'Backdoor', 'Shellcode', 'Worms', 'Fuzzers', 'DoS', and 'Reconnaissance', were removed to improve training stability. These classes contributed little to classification performance and could destabilize the quantum feature extraction workflow. To mitigate potential bias, class weighting was applied to the remaining minority classes, ensuring the model remains sensitive to underrepresented attack types.

3.1.2 Data Preprocessing

Data pre-processing is crucial for preparing the dataset for model training. Initially, the dataset is loaded into a DataFrame, and the target variable, attack_cat, is separated from the feature set. During preprocessing, certain attack classes, such as 'Analysis', 'Backdoor', 'Shellcode', 'Worms', 'Fuzzers', 'DoS', and 'Reconnaissance', were removed. These classes either contained extremely few samples or consistently yielded zero precision and recall in preliminary experiments, making reliable learning and evaluation statistically infeasible. Retaining them caused severe class imbalance, unstable training, and degenerate predictions that could obscure meaningful performance comparisons. Their removal was therefore a deliberate methodological decision aimed at ensuring fair, reproducible, and interpretable evaluation, rather than inflating model performance. To mitigate potential bias introduced by this removal, class weights were applied to the remaining minority classes, thereby ensuring that the model remains sensitive to underrepresented attack types. The remaining class distributions for each dataset are reported to enhance transparency and allow readers to assess the coverage of attack categories.

The target labels are then re-encoded as numeric values using LabelEncoder, which is essential for multiclass classification tasks. Next, the code identifies categorical and numerical columns within the feature set. Categorical features are transformed using one-hot encoding, while numerical features are standardized using StandardScaler. This pre-processing step ensures that the model can effectively learn from the data. The dataset is then split into training and testing sets using an 80-20 split. The following steps were performed:

Removing Irrelevant Columns

Columns that did not contribute to the model's predictive power (e.g., timestamps, IP addresses) were removed. Let D be the original dataset, and D' be the dataset after removing irrelevant columns:

$$D' = D - \{\text{irrelevant columns}\} \tag{1}$$

Encoding Categorical Features

Categorical features were transformed into numerical representations using techniques such as one-hot encoding. If C is a categorical feature with n unique values, the encoding can be represented as:

$$C' = \text{OneHotEncode}(C) \tag{2}$$

Scaling Numerical Features

Numerical features were scaled to a standard range (e.g., [0, 1] or standardized to mean 0 and variance 1). For a feature x, min-max scaling can be expressed as:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{3}$$

Applying Principal Component Analysis (PCA)

To manage data dimensionality and avoid exceeding quantum device limits, Principal Component Analysis (PCA) is applied, reducing the number of features to a maximum of 16 components. This step is crucial for quantum computing, as it directly limits the number of qubits used in the quantum circuit. PCA identifies the principal components that capture the most variance in the dataset, thereby reducing redundancy and improving model efficiency. Given a dataset X with n samples and m features, PCA transforms X into a lower-dimensional space Z:

$$Z = XW \tag{4}$$

where W is the matrix of eigenvectors corresponding to the top k eigenvalues. The advantage of PCA is that it helps to remove noise and computational complexity while preserving essential data patterns, making the reduced features suitable for quantum processing.

3.1.3 Quantum Feature Extraction

Quantum feature extraction encodes classical features into quantum states to leverage the representational properties of quantum circuits. In this work, quantum feature extraction is employed to transform classical cybersecurity features into a quantum-enhanced representation before classification. A quantum device is initialized using PennyLane, and a variational quantum circuit is constructed. Classical input features are encoded into qubits using AngleEmbedding, after which entanglement layers

are applied to capture complex feature relationships. Quantum feature extraction is expected to outperform classical methods because variational quantum circuits can encode high-dimensional correlations through superposition and entanglement, capturing complex nonlinear relationships that classical feature extraction techniques, including PCA and standard neural network embeddings, may struggle to model. It is important to note that our approach differs from existing quantum-enhanced classifiers in that the classifier itself remains classical, while quantum circuits are used solely for feature transformation. Similarly, unlike traditional classical transfer learning, which transfers pre-trained models across datasets without altering the feature space, our approach transfers models trained on quantum-transformed features. This enables richer representations of the data, better class separability, and improved generalization across heterogeneous cybersecurity datasets. The circuit outputs expectation values of Pauli-Z operators, which are treated as quantum-transformed features and passed to the classical learning model for training and evaluation. As a result, the quantum circuit operates on 16 qubits corresponding to the 16 PCA-reduced features, and produces a 16-dimensional quantum feature vector. This choice of embedding and entangling ensures a balance between expressive power and compatibility with near-term quantum hardware constraints.

Encoding Features into Quantum States

Features are encoded using PennyLane's AngleEmbedding technique. A shallow circuit configuration with two entangling layers is employed to capture feature interactions while avoiding excessive depth, which can adversely affect training stability on near-term quantum devices. AngleEmbedding maps each classical feature value to the rotation angle of a quantum gate applied to an individual qubit, allowing classical data to be represented as a quantum state. In this work, each of the m PCA-reduced features is mapped to a separate qubit, resulting in a quantum circuit with $m = 16$ qubits. This embedding enables the quantum circuit to process classical data in the quantum Hilbert space, thereby facilitating nonlinear transformations via subsequent variational operations. AngleEmbedding was chosen to map each PCA-reduced feature to a qubit rotation because of its simplicity, linear scaling with the number of features, and ease of implementation on near-term quantum devices. A shallow circuit with two entangling layers is employed to preserve trainability and avoid excessive depth, which could introduce noise and instability. While alternative embeddings, such as AmplitudeEmbedding, were considered, they require normalization and deeper circuits, which could reduce scalability on NISQ devices.

Quantum Transformations

After embedding, the encoded quantum states are processed using BasicEntanglerLayers, which introduce entanglement among qubits. BasicEntanglerLayers were selected to introduce qubit entanglement in a shallow, parameterized manner, sufficient to capture nonlinear correlations. Alternative entanglers such as StronglyEntanglingLayers may be explored in future work, but BasicEntanglerLayers provided a good balance of expressivity and trainability for this study. These layers form a parameterized quantum circuit (PQC) that applies a sequence of rotation and entangling gates. The

purpose of these transformations is to model nonlinear correlations among features that are difficult to capture with classical linear mappings. The resulting quantum transformation can be described by a unitary operation U:

$$|\psi'\rangle = U|\psi\rangle \tag{5}$$

where U represents the variational quantum circuit that entangles and rotates the quantum states, producing a richer and more expressive feature representation.

3.1.4 Deep Learning Model Training

The model consists of an input layer, a hidden dense layer with ReLU activation, and an output layer with softmax activation for multiclass classification. The model is compiled with the Adam optimizer and sparse categorical cross-entropy loss function. Fig. 3 illustrates the architecture of the deep learning model.

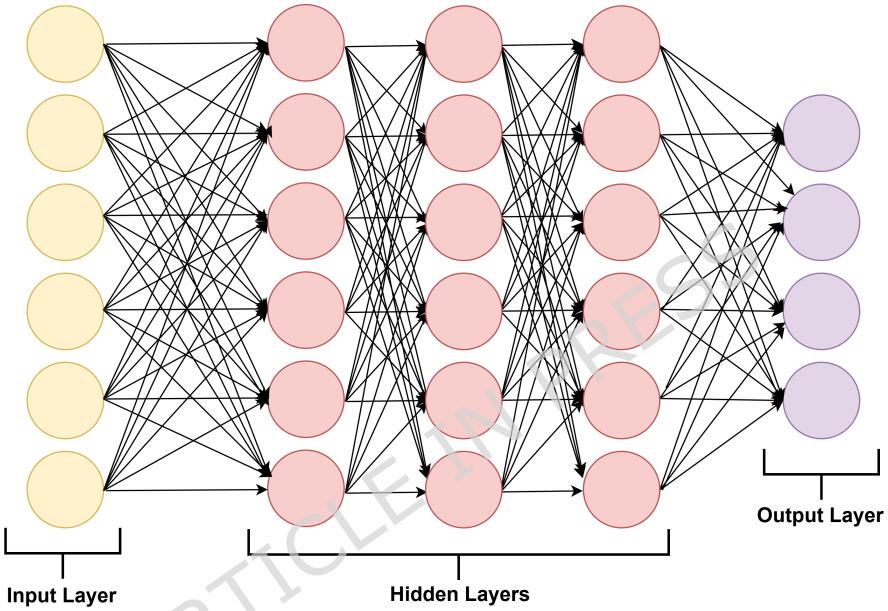


Fig. 3: Deep Learning model Architecture

The deep learning model was trained with the following architecture:

Input Layer

The input layer consisted of quantum-extracted features, denoted as **X**.

Hidden Layers

The model comprised several dense layers with ReLU activation. ReLU introduces nonlinearity, thereby preventing vanishing gradients and improving learning. For a

hidden layer h with weights W_h and biases b_h, the output can be expressed as:

$$h = \text{ReLU}(W_h \cdot \mathbf{X} + b_h) \tag{6}$$

Output Layer

The output layer utilized a softmax activation function for multi-class classification. If z is the output from the last hidden layer, the softmax function is defined as:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{7}$$

where K is the number of classes.

Class Weighting

To address class imbalance, class weights were applied during training. The loss function L can be modified to include class weights w_i:

$$L = -\sum_{i=1}^{K} w_i y_i \log(\sigma(z)_i) \tag{8}$$

Early Stopping

Early stopping was implemented to prevent overfitting by monitoring validation loss and halting training when performance no longer improves. The model is then trained on the quantum features extracted from the training set for 50 epochs with a batch size of 8, while validation is performed on the test set. Finally, the model's performance is evaluated by predicting the classes of the test set, and a classification report is generated to display precision, recall, and F1-score metrics for each class. This comprehensive approach integrates classical and quantum techniques to improve the classification of network traffic data.

3.2 Transfer Learning to CICIDS2017 & CSE-CIC-IDS2018

This subsection explains the workflow for the Transfer Learning to CICIDS2017 & CSE-CIC-IDS2018 Dataset.

Dataset Description

The CICIDS2017 dataset was generated by the Canadian Institute for Cybersecurity (CIC) and includes various attack types, such as Brute-Force, DDoS, and Infiltration. The CSE-CIC-IDS2018 dataset extends the CICIDS2017 dataset by incorporating more sophisticated attack patterns. These datasets are crucial for evaluating intrusion detection models. This dataset was downloaded from Kaggle https://www.kaggle.com/datasets/ernie55ernie/improved-cicids2017-and-csecicids2018.

3.2.1 Dataset Loading and Pre-processing

This work fine-tunes a pretrained deep learning model to classify network traffic using CICIDS2017 and CSE-CIC-IDS2018 datasets. To improve model performance, specific low-precision classes such as Botnet and Web Attacks are removed from the dataset. This filtering ensures that the model focuses on the most relevant traffic types, improving classification accuracy. To ensure consistency with the model originally trained on UNSW-NB15, feature names are mapped accordingly. A predefined column mapping dictionary renames dataset columns to match the original feature set. The dataset also identifies numerical and categorical features, filling missing values with zeros for numerical data and 'unknown' for categorical ones.

A data pre-processing pipeline is applied to the dataset. Numerical features are standardized using StandardScaler, while categorical features are encoded with OneHotEncoder. Dimensionality reduction is performed using Principal Component Analysis (PCA), reducing the number of features to a maximum of 20 while retaining essential information. To prepare for quantum-inspired feature encoding, the reduced feature vectors are normalized using L2 normalization, ensuring all feature vectors have unit magnitude.

The deep learning model is initialized with pretrained weights from the UNSW-NB15 model to facilitate transfer learning. The model architecture is reconstructed to align with the new dataset. It includes an input layer with the same dimensionality as the pre-processed features, a fully connected hidden layer with 32 neurons using ReLU activation, and a final output layer with the number of classes in the new dataset.

3.3 Fine-Tuning on the New Dataset

To train and evaluate the model, the dataset is split into training (80%) and testing (20%) sets using train_test_split. The model is compiled with the Adam optimizer (learning rate: 1×10^{-5}) and sparse categorical crossentropy loss. Fine-tuning is conducted over 20 epochs with a batch size of 16, utilizing validation data to monitor performance. If a pretrained model exists, the approach attempts to load its weights; otherwise, training starts from scratch.

Once training is complete, the model generates predictions on the test set. The model ensures that only valid classes are included in the evaluation. The final results demonstrate the effectiveness of transfer learning in network intrusion detection.

This structured workflow highlights how transfer learning was successfully applied to CICIDS2017 and CSE-CIC-IDS2018 datasets, enhancing intrusion detection accuracy by leveraging knowledge from a pretrained model.

3.4 Transfer Learning to TON_IoT Dataset

This subsection describes the workflow for transfer learning to the TON_IoT Dataset.

Dataset Description

The TON_IoT dataset was developed to simulate IoT network attacks, including DDoS, ransomware, and backdoors. It comprises telemetry data from IoT

devices and network traffic logs, making it particularly useful for cybersecurity research. This dataset was downloaded from Kaggle https://www.kaggle.com/datasets/fadiabuzwayed/ton-iot-train-test-network.

3.4.1 Dataset Processing and Feature Selection

The process begins by reading the data and then removing Unnecessary features, such as timestamps (`ts`) and IP addresses (`src_ip`, `dst_ip`), and retaining only relevant attributes. The dataset is then split into features (X_{new}) and the target variable (y_{new}), where `label` denotes the classification target. To maintain consistency with previous datasets, the script prints the dataset's feature names for verification.

3.4.2 Feature Mapping to Match UNSW-NB15

To align the dataset with the feature structure of UNSW-NB15, a predefined mapping dictionary is used to rename the columns. This ensures compatibility with the previously trained model. Numerical and categorical features are identified separately, with missing numerical values replaced by zero and missing categorical values substituted with `"unknown"`. The dataset is then reconstructed to match the expected structure, where any missing columns are assigned default values.

3.4.3 Feature Transformation Using PCA and Quantum Encoding

A pre-processing pipeline is applied to normalise and encode the features. Numerical features undergo standardisation using `StandardScaler`, while categorical features are encoded using `OneHotEncoder`. To manage high-dimensional data, Principal Component Analysis (PCA) is performed, reducing the feature set to a maximum of 20 components while preserving essential information. The reduced feature vectors are then normalised using L2 normalisation to ensure unit magnitude. This step is particularly important for quantum encoding, which requires unit-normalised inputs.

3.4.4 Model Initialization with Pretrained Weights

The deep learning model is initialized using pretrained weights from the UNSW-NB15 and CICIDS2017 models. To ensure compatibility, the model architecture is reconstructed with an input layer matching the dimensionality of the transformed dataset. A fully connected hidden layer with 32 neurons (using ReLU activation) is added, followed by an output layer adapted to the number of classes in the TON_IoT dataset. The model attempts to load pretrained weights from a checkpoint file. If a mismatch occurs, an error message prompts users to verify the architecture.

3.4.5 Fine-Tuning on the TON_IoT Dataset

Before training, the dataset is split into training (80%) and testing (20%) sets. The model is trained with the Adam optimizer at a learning rate of 1×10^{-5}, and sparse categorical crossentropy is used as the loss function. Fine-tuning is performed over 20 epochs with a batch size of 16, leveraging validation data to monitor performance.

If the pretrained weights are successfully loaded, the model refines its parameters to adapt to the new dataset; otherwise, it starts training from scratch.

3.4.6 Model Evaluation and Performance Analysis

After training, the model generates predictions on the test set, and predicted class probabilities are converted into discrete class labels. A classification report is generated detailing precision, recall, and F1-score for each class. Since the target labels were numerically encoded using `LabelEncoder`, the work retrieves their original string representations to provide interpretable performance metrics. The final evaluation results confirm the effectiveness of transfer learning, demonstrating that knowledge from previous datasets enhances the classification performance on TON_IoT.

4 Experimental Analysis and Results

In this section, the accuracy, precision, recall, and F1 scores are used to evaluate model performance. More specifically, it describes systematic experimental outcomes. This subsection defines all performance measurements, such as accuracy, precision, recall, and F1-score, and indicates how these measurements have to be used. Accuracy is the number of correctly classified instances (TP+TN) divided by the total number of instances in the dataset. By applying equation of 9, the value is computed as:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{9}$$

Precision is the ratio of the number of times the model accurately predicted a product to the total number of times it has predicted it positively. Applying equation 10 in this way will provide this result:

$$Precision = \frac{TP}{TP + FP} \tag{10}$$

Recall is the ratio of true positive predictions to the actual number of positive instances in the data. It reflects the model's ability to capture all positive instances. Use equation 11 in the following manner to find this value:

$$Recall = \frac{TP}{TP + FN} \tag{11}$$

F1-score is the harmonic mean of precision and recall, providing a single metric to balance both. It is particularly useful when there is an imbalance between classes. Use equation 12 in the following manner to find this value:

$$F1 - score = 2 \times \frac{Precision + Recall}{Precision + Recall} \tag{12}$$

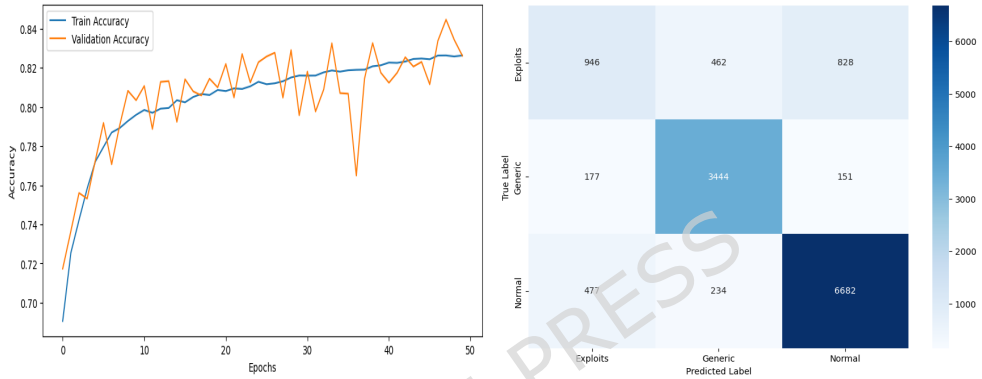
The performance of the proposed quantum-enhanced transfer learning model is evaluated across three different cybersecurity datasets, as detailed in the Tables.

Table 1 presents the classification performance on the UNSW_NB15 dataset. The model achieves an overall accuracy of 83% across 13,401 samples. The highest recall is observed for the "Generic" and "Normal" classes, indicating that the model effectively

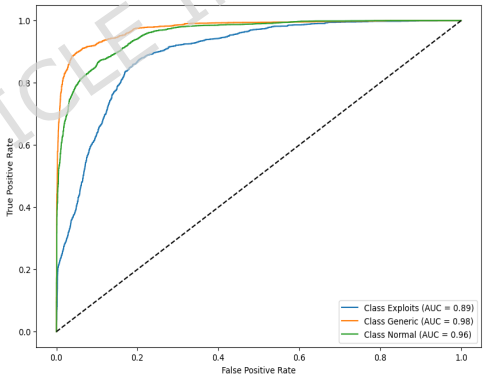
identifies normal traffic and general attack patterns. However, the "Exploits" class exhibits relatively low recall and F1-score, suggesting difficulty in detecting this attack type.

Class	Precision	Recall	F1-Score	Support
Exploits	0.64	0.36	0.46	2236
Generic	0.81	0.93	0.87	3772
Normal	0.87	0.92	0.90	7393
Accuracy	0.83			
Macro Avg	0.77	0.74	0.74	13401
Weighted Avg	0.82	0.83	0.82	13401

Table 1: Final Result - Transfer Learning on UNSW_NB15 Dataset



(a) Accuracy Graph (b) Confusion Matrix



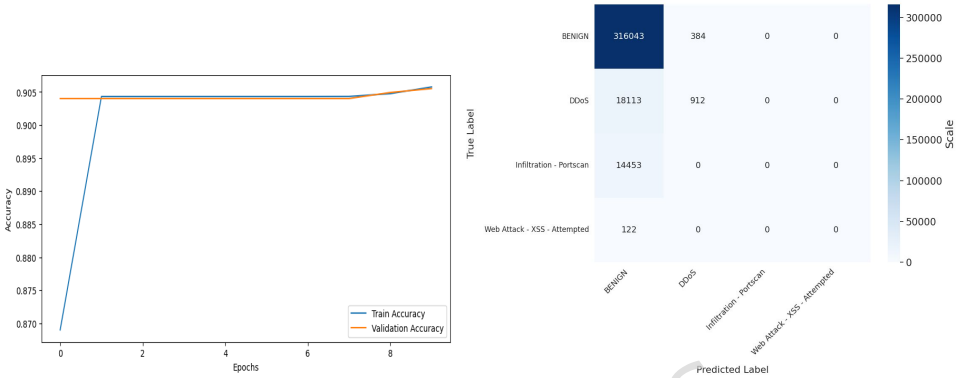
(c) ROC

Fig. 4: Graphical Representation of model for UNSW-NB15 Dataset

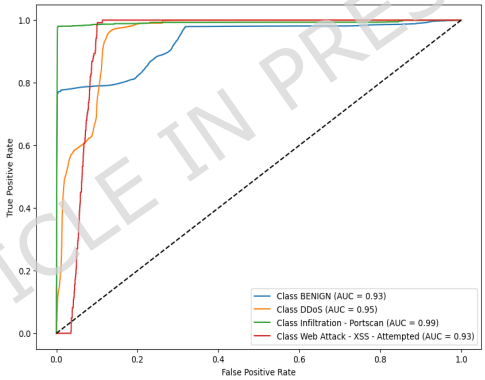
Fig. 4 presents the graphical representation of the model for the UNSW_NB15 dataset, including the accuracy graph, confusion matrix, and ROC curve. Fig. 4a illustrates the training and validation accuracy of a model over 50 epochs. The training accuracy, depicted by the blue line, starts at approximately 0.70 at the 0_{th} epoch and increases steadily, reaching about 0.80 by the 20_{th} epoch. After this point, the training accuracy continues to increase, albeit more slowly, ultimately reaching approximately 0.82 by the 50_{th} epoch. The validation accuracy, depicted by the orange line, starts slightly higher at approximately 0.72 at 0_{th}. It exhibits fluctuations in both upward and downward directions throughout the epochs. By the end of the training, the validation accuracy stabilizes at about 0.81. Fig.4b represents the confusion matrix of a model distinguishing between three categories: Exploits, Generic, and Normal. From the confusion matrix, the model correctly classified 946 Exploits instances, while misclassifying 462 as Generic and 828 as Normal. For the Generic category, the model achieved a high number of correct predictions, labelling 3444 instances accurately, but it misclassified 177 as Exploits and 151 as Normal. In the case of Normal traffic, the model demonstrated strong performance, with 6682 correct predictions; however, 477 Normal instances were misclassified as Exploits and 234 as Generic. Overall, the classifier shows good accuracy in identifying Generic and Normal traffic. Fig.4c illustrates the ROC curve in which the x-axis represents the False Positive Rate (FPR), while the y-axis indicates the True Positive Rate (TPR). The curve for the Class Exploits, shown in blue, achieves an Area Under the Curve (AUC) of 0.89, indicating strong predictive performance. The Class Generic curve, shown in orange, has an AUC of 0.98, indicating excellent classification performance. Lastly, the Class Normal, depicted in green, has an AUC of 0.96, also demonstrating strong predictive power. The dashed diagonal line serves as a reference for a random classifier, where a model performing equally on both classes would fall along this line. Overall, the graph suggests that the model is particularly effective at distinguishing between the Generic and Normal classes, while still performing reasonably well on the Exploits class.

Fig. 5 presents the graphical representation of the model for the CICIDS2017 & CSE-CIC-IDS2018 dataset, including the accuracy graph, confusion matrix, and ROC curve. Fig. 5a displays the training and validation accuracy of a model over 10 epochs. The training accuracy starts at approximately 0.870 in the 0_{th} epoch and rises sharply to around 0.905 by the 1_{st} epoch. Following this initial increase, the training accuracy plateaus, maintaining a consistent value of about 0.905 from the 1_{st} epoch through to the 9_{th} epoch. Conversely, the validation accuracy starts at 0.905 in the 0_{th} epoch and then stabilizes for the remaining epochs. Fig.5b illustrates the confusion matrix of a model trained on the CIC dataset, focusing on distinguishing between BENIGN traffic and various cyberattacks, including DDoS, Infiltration - Portscan, and Web Attack - XSS - Attempted. The model demonstrates strong accuracy in detecting benign traffic, correctly classifying 316,043 instances while making only 384 misclassifications. However, the classification performance declines substantially for attack detection. For DDoS attacks, only 912 were correctly identified, whereas 18,113 were incorrectly labelled as BENIGN. Similarly, all 14,453 instances of Infiltration - Portscan and all 122 instances of Web Attack - XSS - Attempted were misclassified as BENIGN, showing a complete failure to detect these attack types. This result highlights a significant

imbalance in the model's performance, favouring BENIGN predictions and severely underperforming in detecting malicious traffic. Fig.5c illustrates the ROC curve in which class BENIGN is shown in blue, with an AUC value of 0.93, indicating that it performs well. Class DDoS, depicted in orange, has a slightly better AUC of 0.95. The Class Infiltration - Portscan, shown in green, achieves the highest performance, with an AUC of 0.99, approaching perfect classification. Lastly, the Class Web Attack - XSS - Attempted, marked in red, has an AUC of 0.93, comparable to the benign class but less effective than the DDoS and Portscan classes. Curves that lie closer to the top-left corner of the plot indicate better performance, demonstrating that the Infiltration-Portscan class is the most effective.



(a) Accuracy Graph (b) Confusion Matrix



(c) ROC

Fig. 5: Graphical Representation of model for CICIDS2017 & CSE-CIC-IDS2018

Fig. 6 presents the graphical representation of the model for the TON_IoT dataset, including the accuracy graph, confusion matrix, and ROC curve. Fig. 6a illustrates the training and validation accuracy over epochs for a model training process. Training accuracy starts at approximately 0.675 in the 0_{th} epoch and gradually increases

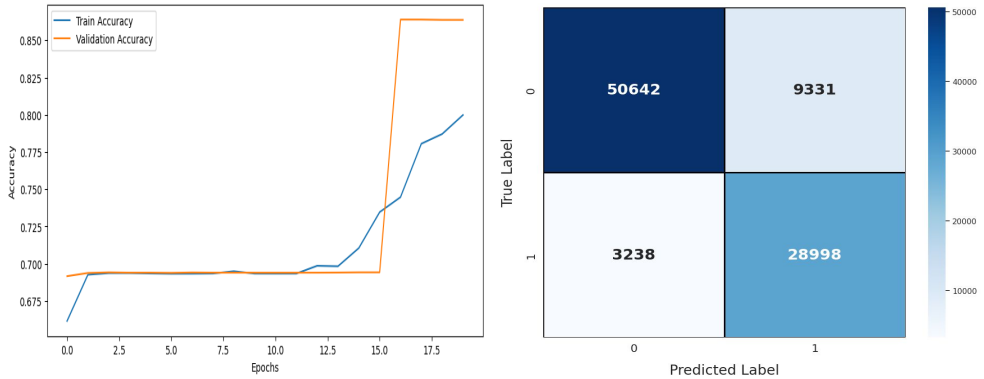
to about 0.825 by the 18_{th} epoch. In contrast, the validation accuracy begins at approximately 0.700 at the 0_{th} epoch and remains relatively stable, with minor fluctuations during the initial epochs. However, a notable surge in validation accuracy occurs around the 15_{th} epoch, peaking at approximately 0.850 by the end of the training. Fig.6b represents the confusion matrix of a model distinguishing between types 0 and 1. The matrix indicates that the model correctly classified 50,643 and 28,998 instances of classes 0 and 1, respectively, while 9,331 instances of class 0 were misclassified as class 1, and 3,238 instances of class 1 were misclassified as class 0. Overall, the classifier achieves high accuracy in classifying both classes. In this particular graph, the AUC (Area Under the Curve) is relatively low at 0.14. This value suggests that the model exhibits poor discriminative ability, indicating that it does not effectively distinguish between the positive and negative classes. The dashed diagonal line represents a random classifier, indicating no discriminative power and highlighting that the model's performance is significantly worse than random guessing. Fig.6c illustrates the ROC curve in which the AUC is relatively low at 0.14. This value indicates that it is not effectively distinguishing between the positive and negative classes. The dashed diagonal line represents a random classifier, indicating no discriminative power and highlighting that the model's performance is significantly worse than random guessing.

Table 2 showcases the results on the CICIDS2017 and CSE-CIC-IDS2018 datasets. The model demonstrates a substantial improvement, achieving an accuracy of 91% on 350,027 samples. The "BENIGN" class exhibits a notably high recall of 99%, ensuring minimal false negatives for normal traffic. However, the "DDoS" class achieves a much lower recall of 16%, indicating that further optimization is needed to improve the detection of distributed denial-of-service attacks. The macro-average scores indicate a significant class imbalance, impacting the detection of minority attack classes.

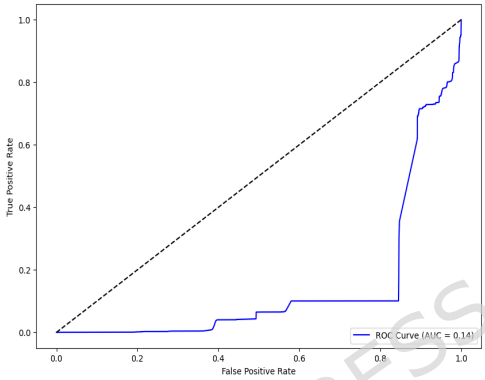
Class	Precision	Recall	F1-Score	Support
BENIGN	0.91	0.99	0.95	316490
DDoS	0.54	0.16	0.25	19112
Accuracy		0.91		
Macro Avg	0.36	0.29	0.30	350027
Weighted Avg	0.85	0.91	0.87	350027

Table 2: Classification Report for Transfer Learning on CICIDS2017 AND CSECICIDS 2018 Dataset

Table 3 presents the classification performance on the TON_IoT dataset. The model achieves an accuracy of 86% across 92,209 samples. Classes "0" (benign) and "1" (attack) exhibit strong recall values of 84% and 90%, respectively, indicating that the model effectively distinguishes between normal and malicious activities. The macro and weighted average F1 scores indicate a well-balanced classification performance across both classes, validating the generalization capability of the quantum-enhanced transfer learning approach.



(a) Accuracy graph (b) CM



(c) ROC

Fig. 6: Graphical Representation of model for TON_IoT Dataset

Class	Precision	Recall	F1-Score	Support
0	0.94	0.84	0.89	59973
1	0.76	0.90	0.82	32236
Accuracy	0.86 (92209 samples)			
Macro Avg	0.85	0.87	0.86	92209
Weighted Avg	0.88	0.86	0.87	92209

Table 3: Classification Report for Transfer Learning on TON_IoT Dataset

5 Discussion and Findings

The findings presented in this work demonstrate that quantum transfer learning is a useful tool for improving cybersecurity threat detection. The most important result is the minimisation of quantum feature extraction. Complex relationships within the data were captured that simple methods might not have captured, using

quantum computing techniques to encode classical features into quantum states via AngleEmbedding or BasicEntanglerLayers. The unique superposition and entanglement properties of quantum mechanics are leveraged by this quantum-enhanced feature representation in the model, thereby improving classification accuracy and (anti)robustness to noise and irrelevant features. A second important aspect of our findings is the use of a transfer-learning approach. Likewise, knowledge from this comprehensive dataset was capitalised on by fine-tuning on the CICIDS2017, CSE-CIC-IDS2018, and TON_IoT datasets, using pre-trained weights from the UNSW-NB15 dataset to initialise the model. More important in cybersecurity is adaptability, as new threats and attack vectors continually emerge. The model achieved 91% and 86% accuracy on the CICIDS2017 and CSE-CIC-IDS2018 datasets, respectively, and 86% accuracy on the ToN-IoT dataset, demonstrating that the theoretically derived knowledge embodied in the UNSW-NB15 dataset was successfully transferred to new settings. Although the UNSW-NB15 accuracy is modest (83%), quantum feature extraction is theoretically expected to yield richer, more expressive representations through superposition and entanglement. This enables the model to capture complex nonlinear correlations that classical feature-extraction methods may miss, thereby supporting improved generalisation and cross-dataset performance, as evidenced by higher accuracies on CICIDS2017, CSE-CIC-IDS2018, and ToN-IoT.

Although several existing deep learning-based intrusion detection approaches reported in the literature achieve higher accuracy on benchmark datasets such as UNSW-NB15, CICIDS2017, and TON_IoT, the proposed quantum transfer learning framework demonstrates competitive performance while offering additional advantages. In contrast to purely classical state-of-the-art methods, the proposed approach emphasises cross-dataset knowledge transfer, reduced retraining requirements, and improved adaptability to new environments, which are critical for evolving cybersecurity scenarios. The bar chart Fig. 7 represents the accuracy of all three datasets.

It suggests that quantum transfer learning could significantly reduce the time and resources required to train models on new datasets, making it a promising approach in fields with changing environments. Additionally, the project addressed another common issue in cybersecurity datasets: data imbalance, in which attacks of a specific type are represented much less frequently than normal traffic. Still, by employing class weighting during training, the model's sensitivity to minority classes was enhanced, which is crucial for accurate threat detection. Despite these measures, the confusion matrix (Fig. 5b) indicates that certain attack classes, such as DDoS, Infiltration - Portscan, and Web Attack - XSS - Attempted, are still poorly detected, with many instances misclassified as BENIGN. This highlights that class imbalance remains a significant challenge and that additional strategies, such as targeted oversampling, anomaly detection techniques, or cost-sensitive learning, may be required to improve the detection of rare yet critical attack types. The model exhibits enhanced capability to detect uncommon yet dangerous threats, thereby strengthening its overall threat-detection capacity.

Quantum computing has demonstrated transformative potential for enhancing cybersecurity detection capabilities when combined with transfer learning, according to this research project. It should be noted that all quantum experiments in this study

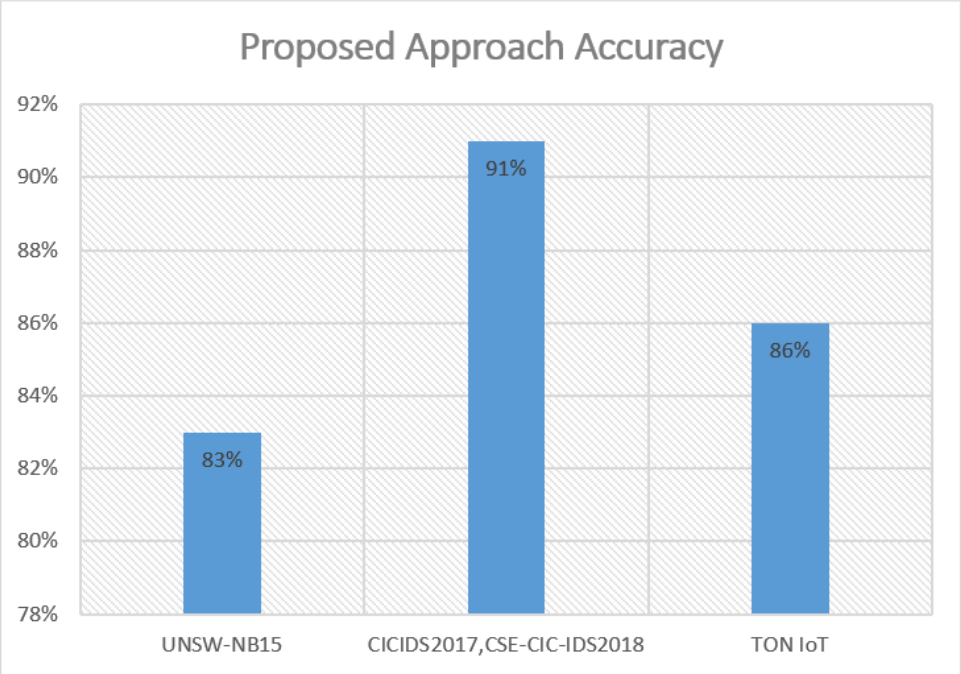


Fig. 7: Proposed Approach accuracy on all datasets

were conducted on a PennyLane simulator under ideal conditions. Deployment on real quantum hardware would be affected by noise, decoherence, limited qubit connectivity, and gate errors. To address these challenges, the proposed quantum circuit is designed to be shallow with only two entangling layers and 16 qubits, making it suitable for near-term NISQ devices. While the proposed quantum feature extraction yields richer representations, it entails greater computational overhead than classical methods. Techniques such as PCA-based dimensionality reduction and shallow circuit design are employed to mitigate this cost, but further Optimization is required for deployment on large-scale, real-time cybersecurity datasets. Future strategies, such as error-mitigation techniques and hybrid quantum-classical approaches, may further enhance robustness and scalability on real quantum hardware. The Residual Joint Antenna Network (RJAN) [27] also utilizes deep neural networks for modeling antenna data. While RJAN effectively captures residual dependencies, integrating our hybrid quantum transfer learning framework with RJAN could further enhance its learning capability, particularly in scenarios with limited data or highly complex interactions. Exploring this integration represents a promising direction for future research.

Future work will evaluate the proposed framework on real network traffic streams to assess its real-time intrusion-detection capability. Additionally, experiments will be extended to larger-scale and more diverse cybersecurity datasets, comparative analysis with other quantum machine learning algorithms, such as quantum support vector

machines and alternative variational quantum circuits, will be conducted, and the inclusion of rare attack classes and data augmentation techniques will be explored to further the model's real-world applicability and robustness in transfer learning scenarios. Additionally, future work will explore integrating Singular Pooling (SP) [28] into the quantum transfer learning framework. SP could provide more informative feature reduction by preserving spectral properties, thereby enhancing cross-domain cybersecurity threat detection and improving feature generalization within hybrid quantum-classical pipelines.

6 Conclusion

Integrating quantum computing with transfer learning in cybersecurity effectively enhances threat detection capabilities. The UNSW-NB15 dataset was used for training quantum feature extraction methods, thereby enhancing the model's pattern recognition capabilities. Transfer learning algorithms achieved higher accuracy on the CICIDS2017, CSE-CIC-IDS2018, and TON_IoT datasets after training. Our quantum-enhanced model achieves 83% accuracy on UNSW-NB15, 91% on the combined CICIDS2017 and CSE-CIC-IDS2018 datasets, and 86% on the TON_IoT dataset, demonstrating the potential of quantum computing and its application in cybersecurity. The experimental results demonstrate that the proposed quantum transfer learning framework effectively captures transferable features across heterogeneous cybersecurity datasets, thereby improving cross-domain threat detection and categorization performance. The results confirm that combining quantum-enhanced feature representation with transfer learning improves robustness against dataset-specific characteristics and variations compared to classical learning approaches. Despite these promising results, deploying such models in real-time cybersecurity applications presents several challenges. Real-time intrusion detection systems must handle high-speed network traffic, strict latency requirements, and limited computational resources. Additionally, integrating quantum-enhanced models into real-world environments requires careful consideration of scalability, inference efficiency, and seamless interaction with existing security infrastructures. Addressing these challenges is essential to the practical adoption of quantum transfer-learning-based cybersecurity solutions. Quantum-enhanced machine learning demonstrates strong potential in addressing complex cybersecurity challenges, and the findings of this work provide a foundation for future research. Future investigations will focus on optimizing computational efficiency, reducing inference latency, and extending the framework toward real-time deployment scenarios, further strengthening its applicability in operational cybersecurity environments.

7 Declarations

Ethics approval and consent to participate

Not applicable.

Consent for Publication

Not applicable.

Availability of data and material

The dataset is available within the article.

Competing Interests

The authors share no conflict of interest.

Funding

The authors extend their appreciation to the Deanship of Scientific Research at Northern Border University, Arar, KSA, for funding this research work through the project number "NBU-FFR-2026-2443-xx" and the Deanship of Research and Graduate Studies at King Khalid University for funding this work through the Large Research Project under grant number RGP.2/379/46.

Authors' contributions

S. A.: Conception and design of study, Writing - original draft, Writing - review & editing, Methodology, Project Administration, Visualisation. M. A.: Acquisition of data, Analysis and/or interpretation of data, Writing - original draft, Writing - review & editing. N. K.: Analysis and/or interpretation of data, Writing - original draft, Writing - review & editing. A.A.: Conception and design of study, Acquisition of data, Analysis and/or interpretation of data, Writing - original draft. J.B.: Writing - original draft, Writing - review & editing, Methodology, Validation, Supervision. A.A.H: Analysis and/or interpretation of data, Writing - original draft, Methodology, Validation. S.A.: Writing - original draft, Acquisition of data, Writing - review & editing, Methodology.

References

[1] Ferrag MA, Alwahedi F, Battah A, et al (2025) Generative ai in cybersecurity: A comprehensive review of llm applications and vulnerabilities. Internet of Things and Cyber-Physical Systems

[2] Al Siam A, Hassan MM, Bhuiyan T (2025) Artificial intelligence for cybersecurity: A state of the art. In: 2025 IEEE 4th International Conference on AI in Cybersecurity (ICAIC), IEEE, pp 1–7

[3] Khanna S (2025) Ai in cybersecurity: A comprehensive review of threat detection and prevention mechanisms. International Journal of Sustainable Devlopment in field of IT 17(17)

[4] Kokaji A, Goto A (2022) An analysis of economic losses from cyberattacks: based on input–output model and production function. Journal of Economic Structures 11(1):34

[5] Li J, Xiao W, Zhang C (2023) Data security crisis in universities: identification of key factors affecting data breach incidents. Humanities and Social Sciences Communications 10(1):1–18

[6] Subramanian G, Chinnadurai M (2024) Hybrid quantum enhanced federated learning for cyber attack detection. Scientific Reports 14(1):32038

[7] Tao X, Liu J, Yu Y, et al (2025) An insider threat detection method based on improved test-time training model. High-Confidence Computing p 100283

[8] Qawasmeh SAD, AlQahtani AAS, Khan MK (2025) Navigating cybersecurity training: A comprehensive review. Computers and Electrical Engineering 123:110097

[9] Sarker IH (2024) Introduction to ai-driven cybersecurity and threat intelligence. In: AI-Driven Cybersecurity and Threat Intelligence: Cyber Automation, Intelligent Decision-Making and Explainability. Springer, p 3–19

[10] Mutalib NHA, Sabri AQM, Wahab AWA, et al (2024) Explainable deep learning approach for advanced persistent threats (apts) detection in cybersecurity: a review. Artificial Intelligence Review 57(11):297

[11] Ejeofobiri C, Fadare AA, Fagbo OO, et al (2024) The role of artificial intelligence in enhancing cybersecurity: A comprehensive review of threat detection, response, and prevention techniques. International Journal of Science and Research Archive 13(02):310–316

[12] Raji A, Olawore A, Mustapha A, et al (2023) Integrating artificial intelligence, machine learning, and data analytics in cybersecurity: A holistic approach to advanced threat detection and response. World Journal of Advanced Research and Reviews 20(3):2005–2024

[13] Dash B, Ansari MF, Sharma P, et al (2022) Threats and opportunities with ai-based cyber security intrusion detection: a review. International Journal of Software Engineering & Applications (IJSEA) 13(5)

[14] Kavitha D, Thejas S (2024) Ai enabled threat detection: Leveraging artificial intelligence for advanced security and cyber threat mitigation. IEEE Access

[15] Thaljaoui A (2025) Intelligent network intrusion detection system using optimized deep cnn-lstm with unsw-nb15. International Journal of Information Technology pp 1–17

[16] Amiri A, Ghaffarnia A, Sakib SK, et al (2025) Focalca: A hybrid-convolutional-attention encoder for intrusion detection on unsw-nb15 achieving high accuracy without data balancing. In: 2025 IEEE 4th International Conference on AI in Cybersecurity (ICAIC), IEEE, pp 1–8

[17] Hassanin M, Keshk M, Salim S, et al (2025) Pllm-cs: Pre-trained large language model (llm) for cyber threat detection in satellite networks. Ad Hoc Networks 166:103645

[18] Shoukat S, Gao T, Javeed D, et al (2025) Trust my ids: An explainable ai integrated deep learning-based transparent threat detection system for industrial networks. Computers & Security 149:104191

[19] Abdelaziz MT, Radwan A, Mamdouh H, et al (2025) Enhancing network threat detection with random forest-based nids and permutation feature importance. Journal of Network and Systems Management 33(1):2

[20] Luqman M, Zeeshan M, Riaz Q, et al (2025) Intelligent parameter-based in-network ids for iot using unsw-nb15 and bot-iot datasets. Journal of the Franklin Institute 362(1):107440

[21] Thiagarajan G, Mahalingam S (2025) Advanced deep learning techniques for anomaly detection in cloud computing traffic: Methods and applications. Available at SSRN 5082090

[22] Rawat KS, Sharma T (2024) A leap from theory to reality: knowledge visualization of quantum computing. IEEE Transactions on Engineering Management

[23] Rawat KS, Yadav M (2025) Analyzing quantum computing applications across key scientific domains using trends and visual analytics. Archives of Computational Methods in Engineering pp 1–32

[24] Sahu K, Kumar R (2025) Telemedicine: how to achieve interoperability without compromising data security. British Journal of Healthcare Management 31(1):1–5

[25] Sahu K, Kumar R (2024) A secure decentralised finance framework. Computer Fraud & Security 2024(3)

[26] Kumar R, Khan SA, Alharbe N, et al (2024) Code of silence: cyber security strategies for combating deepfake disinformation. Computer Fraud & Security 2024(4)

[27] Cai J, Qi Y, Liu S, et al (2025) A residual joint antenna network for joint transmit-receive antenna subset selection in mimo systems. IEEE Transactions on Antennas and Propagation

[28] Zhu S, Cai J, Xiong R, et al (2025) Singular pooling: a spectral pooling paradigm for second-trimester prenatal level ii ultrasound standard fetal plane identification. IEEE Transactions on Circuits and Systems for Video Technology