Computer Modeling in Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2024.056850

ARTICLE

Oversampling-Enhanced Feature Fusion-Based Hybrid ViT-1DCNN Model for Ransomware Cyber Attack Detection

Muhammad Armghan Latif[1], Zohaib Mushtaq[2,*], Saifur Rahman[3], Saad Arif[4], Salim Nasar Faraj Mursal[3], Muhammad Irfan[3] and Haris Aziz[5]

[1]Department of Computer and Information System, Cleveland State University, Ohio, 44115, USA

[2]Department of Electrical, Electronics and Computer Systems, College of Engineering and Technology, University of Sargodha, Sargodha, 40100, Pakistan

[3]Electrical Engineering Department, College of Engineering, Najran University, Najran, 61441, Saudi Arabia

[4]Department of Mechanical Engineering, College of Engineering, King Faisal University, Al Ahsa, 31982, Saudi Arabia

[5]Department of Mechanical, Industrial and Energy System Engineering, University of Sargodha, Sargodha, 40100, Pakistan

*Corresponding Author: Zohaib Mushtaq. Email: zohaib.mushtaq@uos.edu.pk

Received: 01 August 2024 Accepted: 13 December 2024 Published: 27 January 2025

ABSTRACT

Ransomware attacks pose a significant threat to critical infrastructures, demanding robust detection mechanisms. This study introduces a hybrid model that combines vision transformer (ViT) and one-dimensional convolutional neural network (1DCNN) architectures to enhance ransomware detection capabilities. Addressing common challenges in ransomware detection, particularly dataset class imbalance, the synthetic minority oversampling technique (SMOTE) is employed to generate synthetic samples for minority class, thereby improving detection accuracy. The integration of ViT and 1DCNN through feature fusion enables the model to capture both global contextual and local sequential features, resulting in comprehensive ransomware classification. Tested on the UNSW-NB15 dataset, the proposed ViT-1DCNN model achieved 98% detection accuracy with precision, recall, and F1-score metrics surpassing conventional methods. This approach not only reduces false positives and negatives but also offers scalability and robustness for real-world cybersecurity applications. The results demonstrate the model's potential as an effective tool for proactive ransomware detection, especially in environments where evolving threats require adaptable and high-accuracy solutions.

KEYWORDS

Ransomware attacks; cybersecurity; vision transformer; convolutional neural network; feature fusion; encryption; threat detection

1 Introduction

In the last few years, ransomware attacks have become a growing concern and one of the most dangerous cyber threats to organizations and industries including power utilities, financial services, and healthcare. These attacks affect business processes by locking important files, and then demanding ransom payments to unscramble the highly costly information. Thus, current ransomware types are

more complex and challenging to detect since most of the existing methods are based on signature detection which lacks flexibility to adapt to the new threat models. Due to the increased occurrence and sophistication of ransomware attacks, better detection methods have been introduced to identify new patterns in real-time. Substantial work has been done in machine learning (ML) and deep learning (DL) to obtain advanced features in identifying anomalies and likely malicious programs from the data traffic model. However, many of these models are affected by issues like degraded performance on imbalanced datasets, higher computational cost, and unreliability in distinct network environments [1–4], etc. Ransomware is malware that encrypts data files and demands ransom money in exchange for decryption keys, which is a big risk to cybersecurity [5–7]. Numerous studies have addressed the development of effective ransomware detection techniques [8–12]. Due to the higher frequency of these attacks, various methods such as ML, DL, feature engineering, etc. have been employed by researchers to achieve better threat classification results [13–15]. Recent studies indicate that advanced detection measures are indeed critical in the protection of digital systems and data as ransomware threats have been on the rise in the recent past. So, a precise investigation of these studies is required to understand the development of detection approaches and relevant challenges [16–18].

As concluded from the published research review, the existing ransomware detection techniques incur some inherent limitations and face many challenges for their wide applicability to different datasets and varying conditions. A major issue in existing methods is the reliance on a single detection approach, e.g., ML-based models. They have proved to be efficient in detecting known threat signatures but lack adaptability to new and polymorphic attacks. Another significant shortcoming is the class imbalance in typical ransomware datasets which affects the detection accuracy. This aspect is partially addressed by the researchers using oversampling techniques, but a broader investigation is still needed in this area. In pursuit of designing a fast and adaptable approach, this study builds upon the early work by proposing a well-generalized hybrid model incorporating feature fusion, oversampling, and integration of vision transformer (ViT) with one-dimensional convolutional neural network (1DCNN). The key contributions of this research work are as follows:

- Dataset class imbalance incurs data sampling bias during model training resulting in biased classification results. To overcome this issue, the synthetic minority oversampling technique (SMOTE) is introduced in this study. It creates artificial samples to balance the classes and thus helps the model to identify ransomware attacks more effectively. The oversampling process is incorporated into the data preparation stage which optimizes model learning from the balanced data.
- This model uses a state-of-the-art feature fusion and low-rank approximation method for dimensionality reduction. ViT and 1DCNN layer features are combined to extract the inherent properties of network traffic data. Dimensionality reduction is incorporated to reduce computational cost and to identify important features that improve the model performance.
- This novel multimodal framework takes advantage of ViT's capacity for contextual learning while 1DCNN captures the temporal data patterns making it suitable for adaptable classification. Therefore, it improves ransomware detection performance over existing threats as well as for new attack patterns.

Extensive experimentation is carried out to evaluate the performance of the proposed hybrid model by testing it on various datasets, different types of ransomware attacks, and under varying network traffic conditions. The achieved results demonstrate the model's capability by showing higher accuracy, precision, recall, and F1-score metrics confirming its robustness and generalizability for

real-world scenarios. By addressing these key aspects, this study significantly advances the field of ransomware detection, offering a highly effective and scalable solution for cybersecurity applications.

The remaining sections are distributed as follows. Section 2 presents a detailed literature review performed for this study. Numerous studies regarding ransomware detection are reviewed and strengths and limitations of the existing methods are highlighted. The detailed methodology of the proposed hybrid model, feature fusion, oversampling strategy, and dimensionality reduction are presented in Section 3. Section 4 presents achieved results and a discussion about the findings of this work. The research summary, effectiveness of the proposed approach, and future research directions are concluded in Section 5.

2 Literature Review

Over recent years, researchers have made substantial contributions to ransomware detection through various approaches including ML, behavior analysis, and feature engineering. Alvee et al. [1] explored artificial intelligence-based techniques that prevent ransomware attacks on critical infrastructures, underscoring the importance of timely threat detection. However, their approach faced limitations with class imbalances affecting detection accuracy. Lee et al. [2] proposed systematic methods for identifying ransomware using ML but observed difficulty in adapting to novel and polymorphic threats, highlighting the need for well-generalized models. Manavi et al. [4] introduced a long short-term memory (LSTM) network to focus on temporal features. Despite the effectiveness of LSTM in specific scenarios, this approach is non-generalized and limited in diverse network environments. Jayanthi et al. [6] advanced the detection capability by proposing both identification and decryption mechanisms. However, their methodology lacked adaptability to complex encryption patterns, which limits its application against evolving ransomware.

Almomani et al. [7] introduced an Android-based ransomware detection method that targets potential breach points in mobile operating systems. Although significant performance was achieved the need for creating cross-platform models is desirable across different environments. Ashraf et al. [8] provided an elaborative framework for categorizing the approaches towards ransomware detection but it lacks testing on a variety of datasets. Wang et al. [12] proposed a behavior-based detection technique named RanPAS which underscores the significance of distinguishing ransomware features characteristics. This approach helps enhance detection performance, though it would be more effective with the incorporation of feature fusion. Zhang et al. [14] presented dual generative adversarial networks (GAN) to identify unknown encryption ransomware attacks. This model is useful in detecting new and random nature threats; however, it has a limitation of direct application to the network traffic for real-time detection. Numerous studies have been published that employed several methods including ML and DL to address the challenges of ransomware detection [19–23]. Table 1 presents an overview of a few previous studies in terms of their key findings and limitations.

Table 1: Overview of studies presenting ransomware detection approaches

Ref.	Technique	Dataset type	Accuracy (%)	Key findings	Limitations
[3]	Multiple supervised ML models	Ransomware behavior dataset	99.18	Detects ransomware by analyzing malicious API calls.	Relies heavily on API call patterns for detection.

(Continued)

Table 1 (continued)

Ref.	Technique	Dataset type	Accuracy (%)	Key findings	Limitations
[4]	LSTM network with executable file headers	Ransomware samples	93.25	Effective use of LSTM for ransomware detection from file headers.	Focuses on portable executable file headers and may miss other indicators.
[5]	Multiple supervised ML models	Imbalanced ransomware dataset	94.7	Improved detection through balanced datasets.	Less effective with highly imbalanced datasets.
[6]	Detection and decryption of ransomware	Ransomware samples	96	Detects and mitigates ransomware impact.	May not be suitable for real-time detection.
[7]	Multiple supervised ML models	Android ransomware samples	98.3	Effective for detecting ransomware on Android v.11.	Focusing on Android platforms only, not applicable for general ransomware detection.
[9]	Content-based ransomware detection	Ransomware samples	90	Focuses content of ransomware for detection.	May not be effective for encryption-aware ransomware.
[13]	Analysis and detection of bait file characteristics changes	Android ransomware samples	98.24	Detects Android-encrypted ransomware characteristics.	Limited to Android-based ransomware detection.
[14]	Deep convolutional GAN	Mixed dataset from multiple ransomware datasets	98	Performed well for unencrypted ransomware attack detection in comparison to other DL methods.	Requires extensive computing resources for GAN.
[17]	File entropy analysis using SVM	Encrypted ransomware files	92	Improves ransomware detection through file entropy analysis.	May not address behavior-based detection.
[19]	Ransomware clustering and classification using the Jaccard similarity index	Ransomware samples	88	Helps categorize ransomware variants based on similarity index.	Limited to offline classification only, and no real-time detection.
[20]	ML and DL-based methods	Smaller ransomware dataset	96	File behavior-based malware and ransomware detection.	Generic approach used without details of specific methods.
[21]	Fast selective hashing techniques	Limited ransomware samples	86	Behavioral ransomware detection with reduced detection time.	Limited explanation of selective hashing techniques.

(Continued)

Table 1 (continued)

Ref.	Technique	Dataset type	Accuracy (%)	Key findings	Limitations
[22]	LR with feature selection and data preprocessing	Dataset of executable binaries	93.86	Ransomware detection improved using feature selection and data preprocessing.	Information on specific features and model parameters is not given.

Note: API, application programming interface; LR, logistic regression; SVM, support vector machines.

In the review of recent research works, many studies focused on ransomware detection from cloud-encrypted data using advanced variants of DL and transfer learning. Singh et al. [24] presented the RANSOMNET+ model which hybridized CNN with pre-trained transformers and obtained 99.1% detection accuracy with a unique feature set comprising hierarchical features and local patterns. A study [25] implemented an ensemble of multilayer perceptron (MLP) which outperformed the performance of individual MLP models with an accuracy of 98.79% in real-time detection of RaaS attacks in cloud-encrypted data. Urooj et al. [26] introduced a weighted GAN model to address behavioral drift in ransomware detection. By focusing on early detection, the weighted GAN architecture showed robust results in dynamic threat environments. However, the model's complexity raises concerns about deployment in resource-constrained scenarios. Ispahany et al. [27] provided a comprehensive review of ML-based ransomware detection detailing current limitations and suggesting future research directions. Despite insightful findings, this study lacks practical guidelines for implementing proposed strategies in real-world systems which limits its direct utility for practitioners.

Ferdous et al. [28] developed a systematic framework for artificial intelligence-based ransomware detection with data collection, preprocessing, feature extraction, model training, and evaluation. This structured approach enhances detection consistency and robustness. However, the framework may benefit from additional considerations regarding dataset diversity which may improve generalizability across different ransomware types. Hernandez-Jaimes et al. [29] presented an ML model based on Nilsimsa fingerprinting which is specifically designed for ransomware detection on the internet of medical things (IoMT). As its application is specific to IoMT, it limits broader applicability to non-IoMT settings despite its effectiveness in generating unique fingerprints for ransomware. Marcinkowski et al. [30] developed a technique named method for interpretable ransomware attack detection (MIRAD) which prioritizes interpretability by using a simplified additive model for ransomware detection. Interpretability is crucial in high-risk environments but reliance on simplified models may compromise detection accuracy, particularly for novel or highly sophisticated ransomware. Hill et al. [31] explored ransomware classification through hardware performance counters on non-virtualized systems and achieved over 95% accuracy with limited hardware event features. This approach is innovative but may not transform well to virtualized or cloud-based environments which limits its scalability.

Rana et al. [32] focused on countermeasures against ransomware in cyber-physical systems emphasizing web-based automated defense. This method offers a layered security approach, however, its reliance on web defense could be bypassed by sophisticated attackers using encrypted channels. Liu et al. [33] proposed an image-based CNN framework for multi-class malware detection which demonstrated high accuracy through balanced sampling and data augmentation techniques. However, the reliance on visual features may limit its applicability to text-based malware such as script-based ransomware. Lee et al. [34] presented the CENSor model which is a graph-based method for detecting

illicit Bitcoin operations and is significant for tracking ransomware payments. This framework is effective in identifying illegal transactions; however, it does not directly address ransomware detection and thus may serve as a complementary tool only rather than a primary detection mechanism. Finally, Khanan et al. [35] conducted a systematic literature review on intrusion detection system datasets aiming to enhance cybersecurity awareness. The review provides a comprehensive overview but could be further strengthened by including emerging datasets for ransomware detection, thus making it further relevant to current cybersecurity needs.

3 Methodology

3.1 Dataset Description

In this study, an open-source dataset UNSW-NB15 is employed. It includes network traffic data for ransomware detection which contains features describing distinct aspects and patterns of network communication. These features are important to detect potential ransomware activities because they represent sufficient detail and context of network parameters that are required to classify attack behavior. Selected dataset attributes and their descriptions are presented in Table 2.

Table 2: Dataset features and descriptions

Feature	Description
id	Unique identifier for each data point which serves as an index for referencing and organization.
dur	The duration of network traffic flow in seconds represents the active time for a connection.
spkts	Number of source packets transmitted during the network flow.
dpkts	Number of destination packets transmitted during the network flow.
sbytes	Number of source bytes transmitted during the network flow.
dbytes	Number of destination bytes transmitted during the network flow.
rate	The rate of data transmission measured in packets per second.
sttl	Source time-to-live (TTL) indicates the number of hops a packet can make before it is discarded.
dttl	Destination TTL reflects remaining hops allowed for a packet at its destination.
sload	Source load represents data load from the source.
dload	Destination load reflects data load at the destination.
sloss	Number of source packets lost during transmission.
dloss	Number of destination packets lost during transmission.
smean	Mean packet size for source packets.
dmean	Mean packet size for destination packets.
Ransomware	A binary class label indicating the network traffic flow associated with ransomware samples (as 1) and benign samples (as 0).

The distribution of binary class among the dataset samples is shown in Fig. 1. Fig. 2 shows the occurrence frequency of various network services and protocols in the dataset. The extracted features

cover both the communication aspect of ransomware attacks as well as their behavioral aspects. As this model does not rely on traditional features only like payload, command, and control, or malicious IP addresses, it can recognize slight changes in network traffic that might indicate ransomware activity, such as changes in packet size, flow duration, protocol types, and file entropy, etc. These features are most efficient in detecting phases of ransomware activity including lateral movement, data extraction, and encryption. In addition, the inclusion of IP addresses and service types enhances the ability of the model to identify communications with blacklisted IPs and services that are typically used by ransomware. This combination allows the temporal and structural approach to detection as it is necessary for the identification of ransomware.

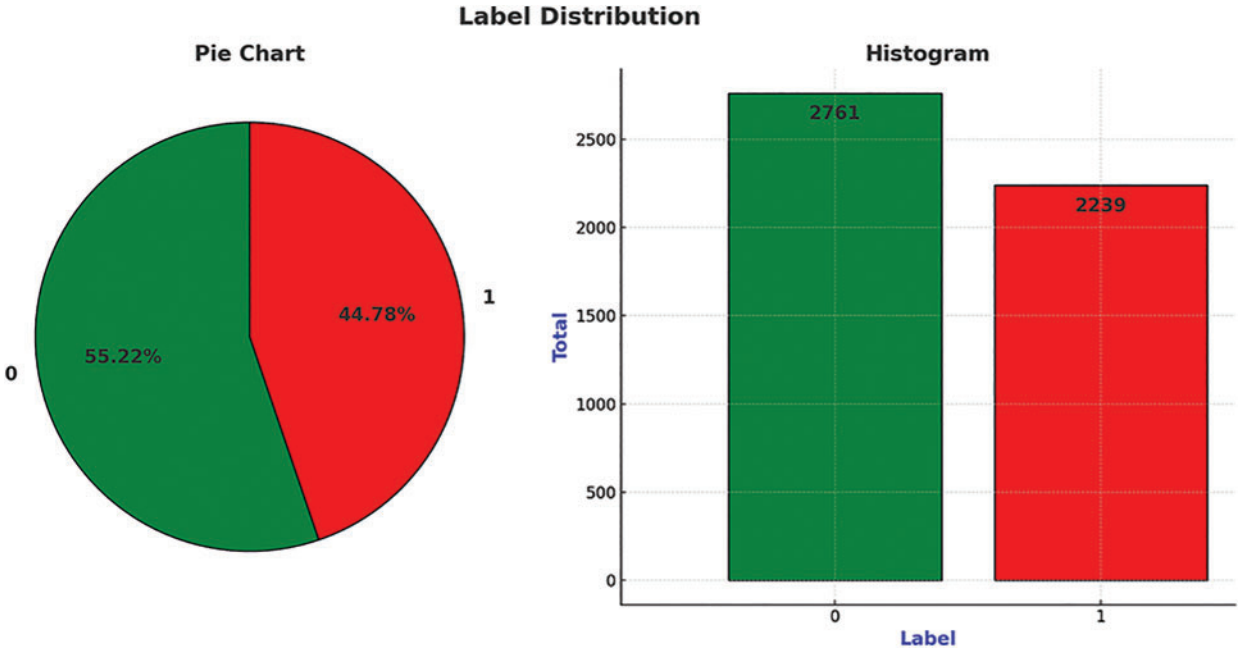

Figure 1: Dataset class distribution (1: ransomware, 0: benign) samples

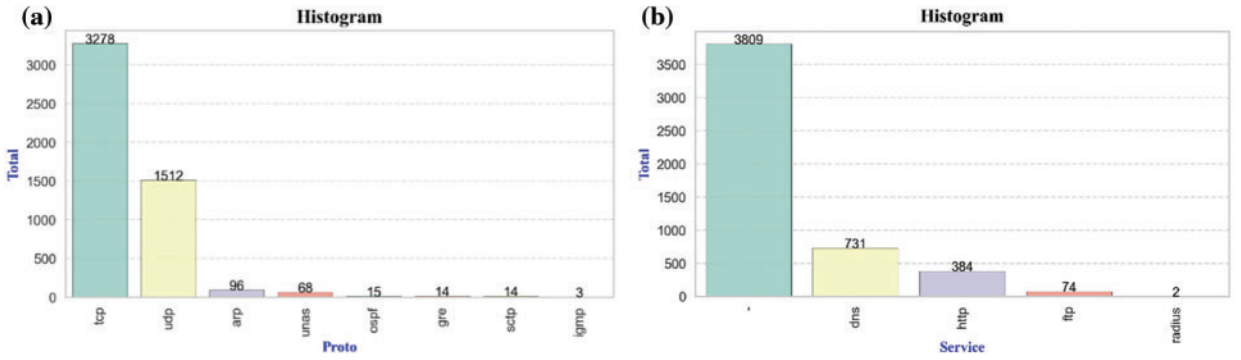

Figure 2: Categorical variable distribution in the dataset: (a) protocol distribution, (b) service distribution

3.2 Feature Fusion

Feature fusion has been incorporated in this study to enhance the representational capability of the dataset for better classification. Some patterns and dependencies between the features could not be surfaced when employed individually. Dataset quality is enhanced with feature fusion by modeling the inter-dependency of individual features and hence generating new combinations that are more representative of the data. This increase in feature space dimension leads to an enhanced model's discriminative ability, generalization, and detection performance. Furthermore, it can also be useful in reducing the data dimension which is important for the model performance and its interpretability. This fusion is crucial for achieving the best performance of ransomware identification from the characteristics of both the ViT and 1DCNN. Thus, this study uses global contextual information as well as localized sequential patterns of network traffic data. Those features are fused which either have a higher correlation or are hypothesized to yield effective insights upon fusion. For example,

- X_{total} captures the total data volume by adding the number of bytes transmitted from the source and destination (sbytes + dbytes).
- Y_{total} represents the overall packet exchange within a connection and is obtained by fusing the number of packets transmitted from the source and destination (spkts + dpkts).
- Z_{total} reflects the total load within a network connection and is computed by adding the load data from the source and destination (sload + dload).
- W_{total} indicates the overall packet loss within a connection obtained by combining the number of packets lost during the transmission from both the source and destination (sloss + dloss).

Features like total bytes and total packets represent total activity at the source and destination of the network connection which helps the model to capture fine details and correlations which are useful in ransomware identification. Fig. 3 shows the occurrence frequency of features among the dataset samples. The spread follows the Gaussian distribution over the entire range of feature values. Fig. 4 shows the two-dimensional mapping of binary feature space before and after feature fusion among dataset samples. Related attributes are fused into composite features which reduce the complexity of data transfer from network connection to the classification model, as it decreases the number of input features instead of employing higher dimensional feature space resulting from feature union.

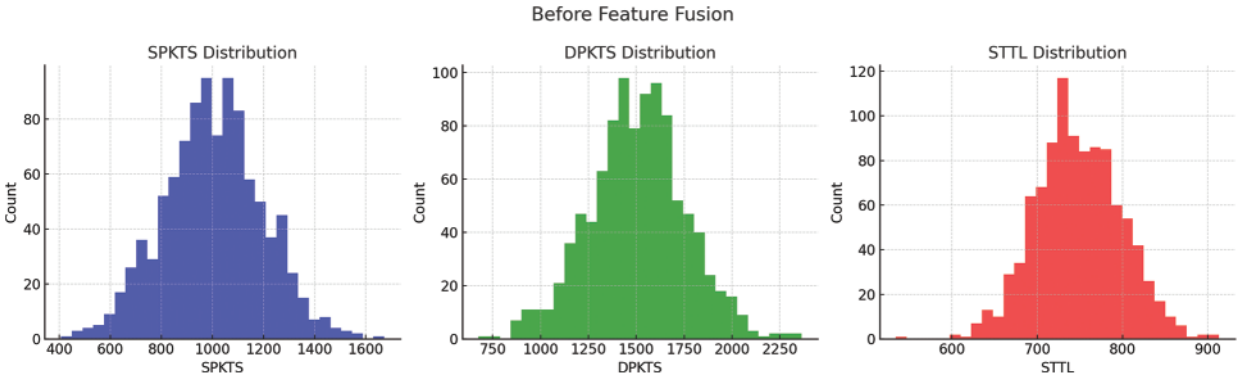

Figure 3: Histogram of individual features of the dataset

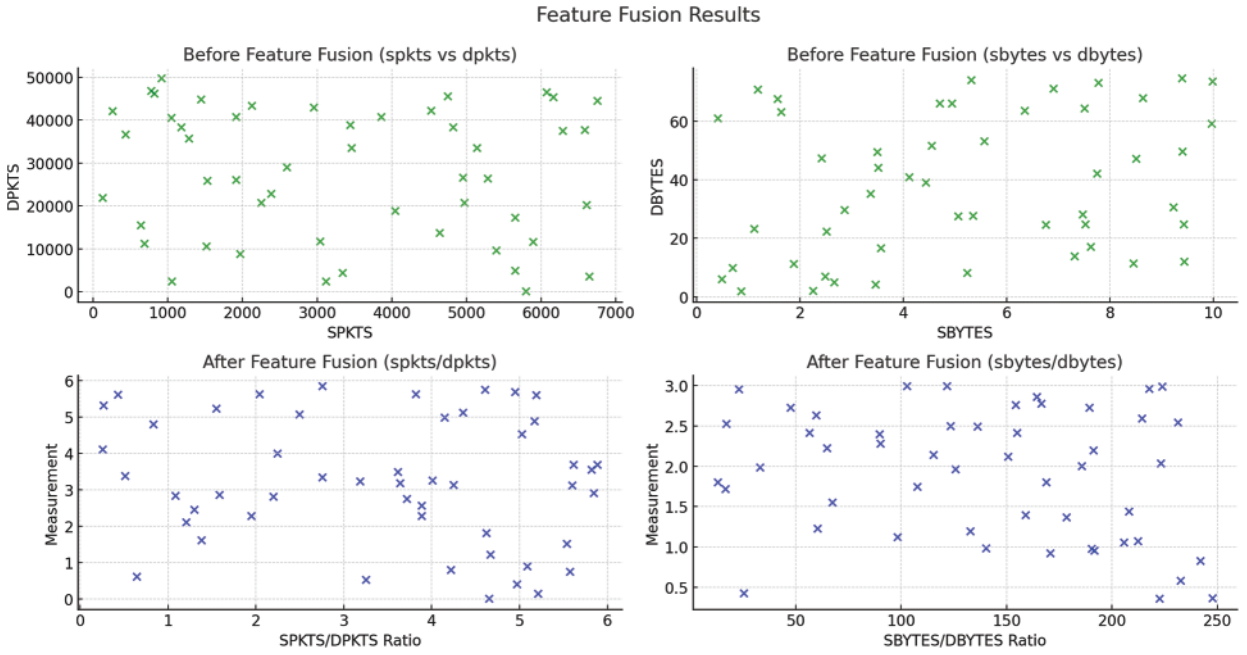

Figure 4: Two-dimensional feature space mapping before and after the feature fusion

3.3 Oversampling

After data cleaning, the resulting dataset had a critical class imbalance with a higher number of benign network samples which affects classification accuracy by decreasing ransomware detection probability. This issue of decreased performance for underrepresented classes is addressed by oversampling the minority class in ML classification problems. In this study, the SMOTE oversampling technique is applied to generate new samples of the minority class by synthesizing them between actual instances of the cleaned dataset. It helps avoid bias toward the dominant class and increases the chances of having a good sample size with a balanced class selection for model training and testing. Among various oversampling methods, SMOTE was chosen because it provides increased model performance by reducing overfitting, better distinction of minority class, enhancing the model's sensitivity to features, and balanced model training for both classes. It was applied for all the features by maintaining the original distribution of feature space for synthetic samples. Fig. 5 shows the distribution of cleaned dataset samples between target variables before oversampling, as well as the resulting distribution with the application of SMOTE oversampling. The resulting dataset is comparatively fair to prevent class imbalance-related issues and hence leads to more accurate ransomware classification.

3.4 Feature Selection

In ML and data analysis, feature selection is used to narrow down a large feature space of candidate features to a manageable subspace containing important attributes only. Its objective is to reduce the model's complexity by eliminating the least important attributes. It is an important factor for data preparation and various approaches are available to choose the best-suited method. In this study, correlation analysis, and random forest (RF) feature selection techniques are employed.

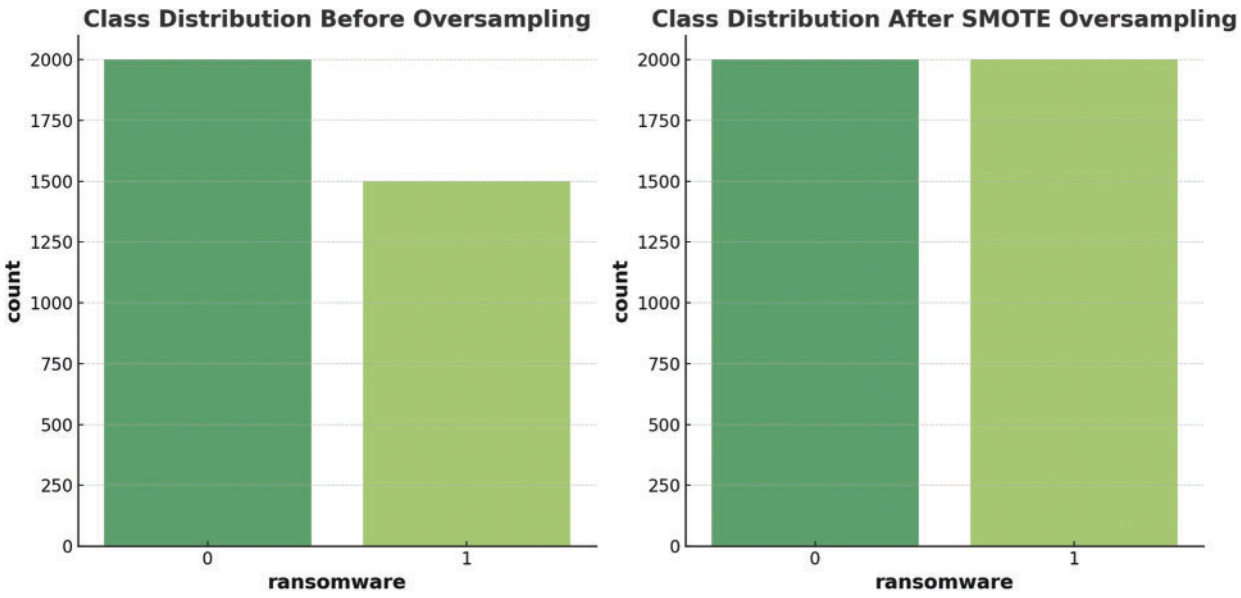

Figure 5: Class distribution of cleaned dataset before and after oversampling

3.4.1 Correlation Analysis

Correlation analysis is used to find interdependent relationships between features. To find correlated features, the Pearson correlation coefficient (r) is computed for all the binary feature pairs. The following formula is used to measure this linear relationship between two variables.

$$2r = \frac{\sum (X - X^{-})(Y - Y^{-})}{\sqrt{\left(\sum (Y - Y^{-})^{2} \sum (X - X^{-})^{2}\right)}} \tag{1}$$

where X and Y are two feature variables under consideration, X^{-} and Y^{-} are their mean values, respectively.

Fig. 6 shows the feature correlation heatmap showing the correlation between all the binary feature pairs. The correlation coefficient ranges from -1 (perfect negative correlation) to 1 (perfect positive correlation). Values close to 0 indicate little or no correlation at all. Highly correlated features are identified from these results and further considered for ransomware detection.

3.4.2 Random Forest

An embedded-type feature selection approach comprising the RF method is employed in this study. This ensemble learner uses decision trees in the RF to find the output prediction power of each attribute. RF measures the significance of each feature for model prediction and rates them based on their importance score. High-importance features are selected for further model evaluation while low-importance features are discarded. Using the Gini impurity index and the total number of decision trees in RF, significance scores for features are calculated with the following relationship:

$$Importance\,(F_i) = \frac{1}{N} \sum_{j} Gini\ Decrease_j \tag{2}$$

where *Importance*(*Fi*) is the importance score for feature F_i, N is the number of decision trees in RF, *Gini Decrease*$_j$ measures reduction in Gini impurity when feature F_i is used in j tree.

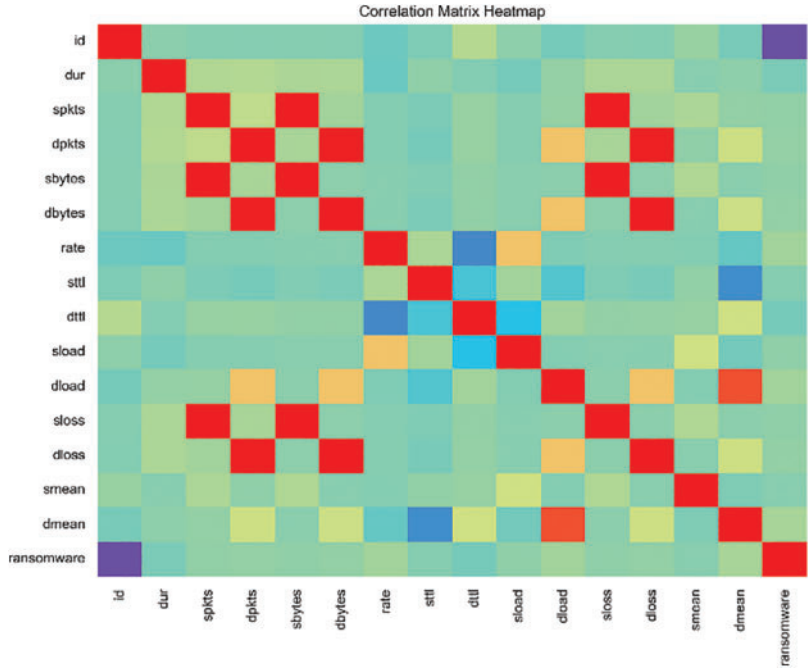

Figure 6: Feature correlation matrix for all binary feature pairs

By ranking features based on their importance scores, the most relevant attributes are selected for ML models.

Fig. 7 shows the top ten important features in feature ranking based on their importance scores. This reduction in the number of features makes the dataset more manageable and hence improves the model performance. Irrelevant or redundant features can introduce noise and adversely affect the model's accuracy. Introducing feature selection in this methodology ensures working with the most relevant attributes which leads to effective ransomware prediction by the classification model.

3.5 Dimensionality Reduction

To preserve the underlying structure and relationships between data points, dimensionality reduction methods are used to map high-dimensional data onto a lower-dimensional feature space. t-distributed stochastic neighbors embedding (t-SNE) based dimensionality reduction approach is employed in this study to comprehend relationships between features of cleaned and balanced datasets. It is a nonlinear method for reducing dataset dimensions which focuses on maintaining the pairwise similarities between data points. In contrast to linear methods like principal component analysis, it attempts to preserve the local structure of the data in the reduced dimension. It is achieved by simulating the joint probability distribution of pairwise similarities between high-dimensional and low-dimensional data points. Given a dataset X containing n data points, t-SNE works as follows.

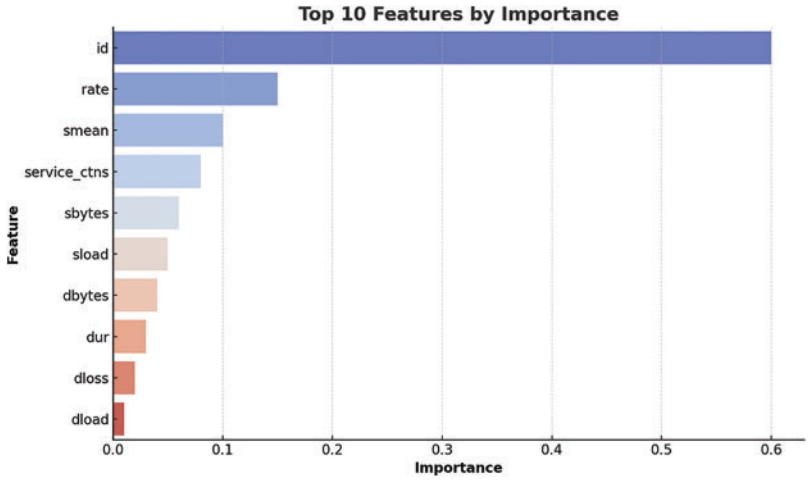

Figure 7: Feature ranking based on feature importance scores

High-dimensional similarities (P): Define a conditional probability distribution to measure similarity between two data points in high-dimensional space. This similarity is computed using a Gaussian distribution over a pairwise Euclidean distance.

$$P_{i,j} = \frac{e^{-\frac{||x_i - x_j||^2}{2\sigma^2}}}{\sum e^{-\frac{||x_i - x_j||^2}{2\sigma^2}}} \tag{3}$$

where $P_{i,j}$ is the conditional probability of similarity between data points x_i and x_j, σ^2 is the variance of the Gaussian distribution which is determined through a binary search to match a given perplexity value.

Low-dimensional similarities (Q): Define a conditional probability distribution to measure the similarity between two data points in the low-dimensional space. This similarity is computed using a t-distribution over a pairwise Euclidean distance.

$$Q_{i,j} = \frac{\left(1 + ||y_i - y_j||^2\right)^{-1}}{\sum_k \left(1 + ||y_i - y_j||^2\right)^{-1}} \tag{4}$$

where $Q_{i,j}$ is the conditional probability of similarity between data points y_i and y_j.

The objective function (cost function): t-SNE aims to minimize the dissimilarity between high-dimensional and low-dimensional pairwise similarities and is achieved by minimizing the Kullback-Leibler divergence between them using the following relationship:

$$C = KL\,(P||Q) = \sum_{i,j} P_{i,j} \log\left(\frac{P_{i,j}}{Q_{i,j}}\right) \tag{5}$$

where C is the cost function, $KL(P||Q)$ represents the Kullback-Leibler divergence between P and Q.

Fig. 8 shows the data distribution before and after the t-SNE application. This interactive technique is used for exploring high-dimensional data to reveal visual patterns and data groups. It

processed the data by compressing data size while retaining the local relationships which can be observed in this data distribution plot. Thus, introducing t-SNE in the proposed methodology helps in a better understanding of the underlying data structure and allows to select most informative features to enhance the model's performance.

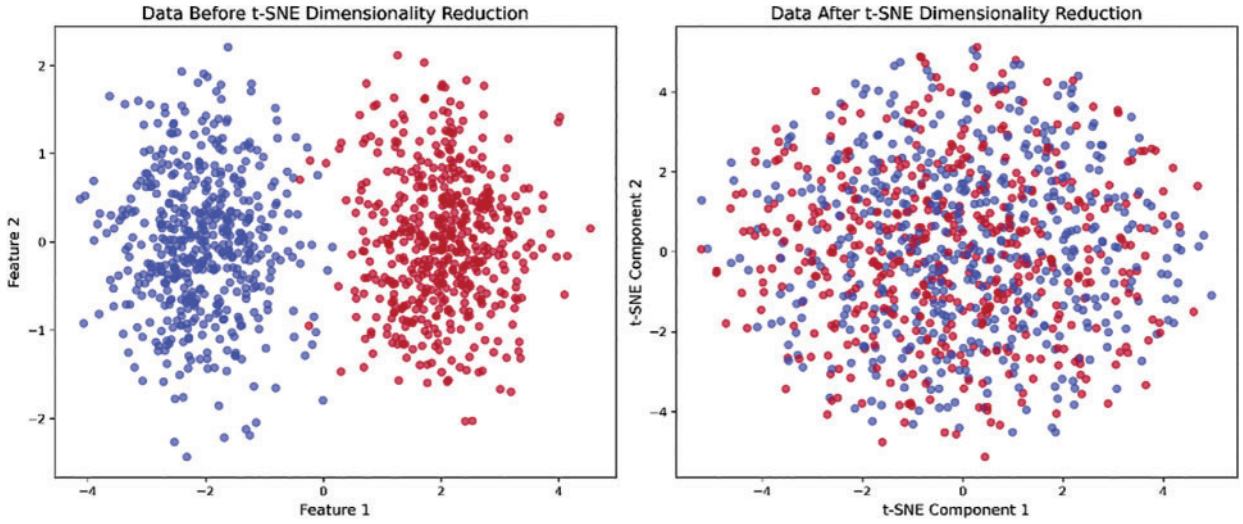

Figure 8: Dataset before and after t-SNE application

3.6 Hybrid Classification Model

The proposed ViT-1DCNN model leverages the strengths of two powerful DL architectures namely vision transformer and deep convolutional neural network in parallel. This hybridization provides an efficient and effective ransomware detection approach. Details on individual DL models and their components are given below.

3.6.1 Vision Transformer

ViT is a DL architecture mostly used for computer vision applications. It presents a unique method to address the challenges of sequential data by using self-attention mechanisms. Fig. 9 shows detailed ViT architecture with representation of its subcomponents. **Input patch embeddings** receive sequential patches as input. These patches are linearly embedded into lower-dimensional representations, usually denoted as $x \in R^N \times C$, where N represents the number of patches, and C represents the dimensions of each patch embedding. **Positional encodings** are added to the patch embeddings to capture the spatial information. It helps models to identify the relative positions of different patches. **Multi-head self-attention mechanism** is the core of ViT which allows the model to focus on different parts of the sequential data when making predictions, thus enabling it to learn complex patterns and relationships. After the self-attention stage, **feed-forward neural networks** process the attention-weighted features to create significant descriptions for downstream tasks. The inclusion of ViT in the hybrid model exploits its powerful capabilities in capturing intricate patterns and dependencies within the dataset.

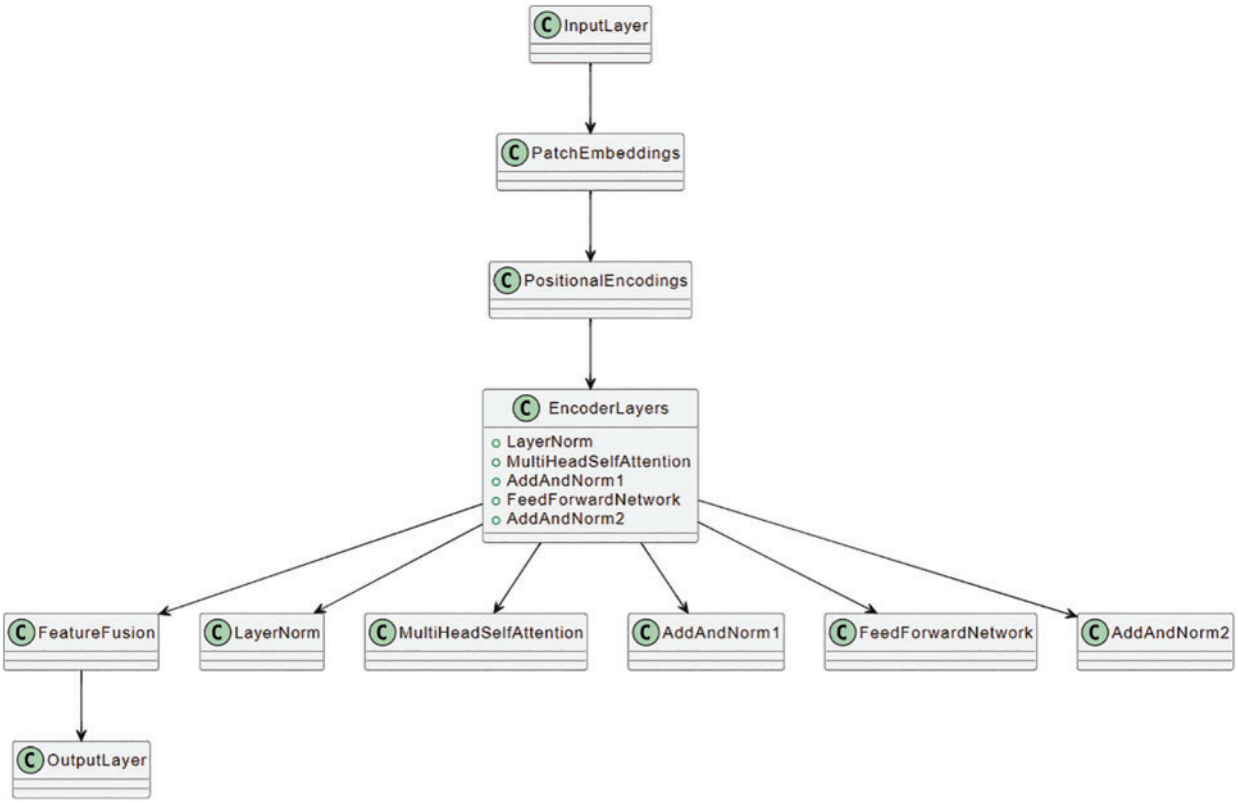

Figure 9: Architecture of vision transformer model

3.6.2 One-Dimensional Convolutional Neural Network

1DCNN are designed for sequential data analysis which makes them suitable for detecting patterns in sequential data such as time series or one-dimensional data vectors. Fig. 10 shows detailed 1DCNN architecture with representation of its subcomponents. **Convolutional layers** use convolutional filters to identify local patterns or features within the data. The sliding convolution operations over the input data extract the relevant features. **Pooling layers** down-sample the spatial dimensions of feature maps to reduce computational complexity and focus on the essential features. After feature extraction, **fully connected layers** are employed for weighted linear transformation on input vectors and to provide resulting data to the output layer for classification or regression tasks.

3.6.3 ViT-1DCNN Hybrid Architecture

The final model hybridizes both the ViT and 1DCNN in parallel after getting data from the input layer. In the end, the outputs of these parallel models are fused in the feature fusion layer before the final output layer as shown in the ViT-1DCNN model architecture in Fig. 11.

The input layer receives the network traffic data as a sequence of feature vectors representing the ransomware dataset denoted as $X = [x_1, x_2, \ldots, x_n]$, where each x_i is a feature vector of dimension m. The dataset contains n number of samples for m number of features. These inputs are further supplied to ViT and 1DCNN branches simultaneously for their unique data processing.

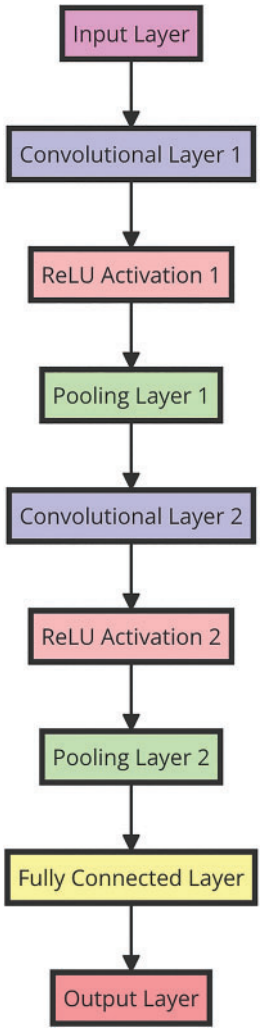

Figure 10: Architecture of one-dimensional convolutional neural network

In the ViT branch, firstly the positional encodings are added to the input features as follows:

$$X' = X + P \tag{6}$$

where positional encodings are represented as a matrix P of dimensions $n \times m$, where each element $P_{\{i,j\}}$ encodes the position of feature j within sample i, and X' is the position-encoded input data. Then X' is processed by further layers to give the ViT output $H_{VIT} = [h_1, h_2, \ldots, h_n]$, where each h_i represents the learned visual representation of the corresponding input feature vector x_i.

In the 1DCNN branch, all the model layers operate on feature vector X and give the output $Y_{CNN} = [y_1, y_2, \ldots, y_n]$ which represents the learned temporal feature vector matrix, where y_i is the learned feature representation. Y_{CNN} is intended to detect temporal patterns in the network traffic and treat the data as a sequence, for example, the order of the packets. It is powerful to identify localized behavior like several consecutive data transfers or bursts of traffic that suggest network attack or data exfiltration.

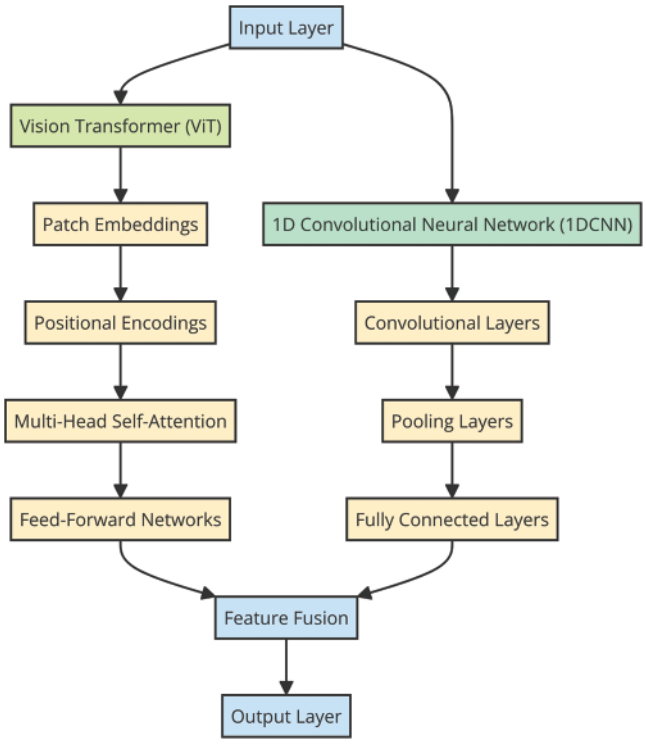

Figure 11: Architecture of ViT-1DCNN hybrid model

Then a feature fusion layer is used to combine the learned visual and sequential features. This layer fuses the features obtained from the ViT layer (H_{VIT}) and the 1DCNN layer (Y_{CNN}). The fused features are denoted as $F = [f_1, f_2, \ldots, f_n]$ where each f_i represents the combined feature representation. This layer can be mathematically represented as:

$$F = Fusion\,(H_{VIT}, Y_{CNN}) \tag{7}$$

The specific fusion operation depends on the architecture and model goals. Common approaches include concatenation, weighted summation, element-wise addition, or other fusion techniques.

Finally, the fused features F are fed to a fully connected layer and then the output layer to perform ransomware detection. $O = [o_1, o_2, \ldots, o_n]$ denotes the output of the model where each o_i represents the prediction for sample i. The output layer can be expressed as:

$$O = Output\,(F) \tag{8}$$

The *Output* is the mathematical function to produce the final binary classification result such as sigmoid, etc.

This hybrid model has shown remarkable promise in ransomware detection due to its ability to adapt to evolving threats by learning from both visual and sequential complex patterns of the data. Its effectiveness in fusing information from different data representations results in improved ransomware detection accuracy. The model is trained to optimize its parameters during the training process, ensuring effective data pattern capture.

To address the potential risk of overfitting with direct oversampling, focal loss is incorporated into this model. Focal loss emphasizes training on minority classes by down-weighting the loss assigned to well-classified examples, thus reducing the impact of the majority class and focusing the model on hard-to-classified examples.

3.7 Performance Metrics

Confusion matrix-based performance assessment metrics are used to evaluate the binary classification performance of the proposed model. Table 3 presents the list of performance metrics employed in this study with their formula and description.

Table 3: Performance assessment metrics

Metrics	Formula	Description
Accuracy	$\frac{(TP+TN)}{(TP+TN+FP+FN)}$	Measures the overall correct predictions.
Precision (positive predictive value)	$\frac{TP}{(TP+FP)}$	Measures the accuracy of positive predictions.
Recall (sensitivity, true positive rate)	$\frac{TP}{(TP+FN)}$	Measures the ability to correctly detect positive instances.
F1-score	$2 \times \frac{(Precision \times Recall)}{(Precision + Recall)}$	Balances precision and recall into a single metric.
AUROC	Area under the receiver operating characteristic curve	Measures the model's ability to distinguish between classes.

4 Results and Discussion

In this section, the performance measure metrics results are presented to evaluate the overall performance of the proposed hybrid model in comparison to the individual models and published research models. The results are also compared for various datasets, oversampling techniques, detection algorithms, and computational costs.

Fig. 12 shows the classification results and decision boundary of the proposed hybrid model ViT-1DCNN for the detection of ransomware attacks. The proposed model was able to classify the available network traffic data with a high accuracy of 98%. It achieved 97% precision suggesting that the model is efficient at generating a positive result with a limited number of false positive outcomes. 97.5% recall shows the model's excellent capability to detect true positive cases. The F1-measure of 98% means that both precision and recall were equally well achieved. The high classification ability of the model was confirmed by the AUROC score of 0.96 which shows a higher detection rate and effectiveness of the approach for cybersecurity applications.

In the decision boundary plot, red circles denote the data samples labeled as ransomware and purple circles denote the samples labeled as normal network traffic. The shades around the decision boundary show the levels of decisions made by the model about the data points. The decision boundary

itself is important because it shows how well the model can distinguish ransomware from other data points. The graph depicts the two classes are well separated by this boundary. It shows that the proposed model can effectively distinguish between most of the data points. Nevertheless, the fact that the classes are not entirely disjoint indicates that there may be cases where the model fails to correctly classify ransomware traffic from normal traffic, especially in regions with a high density of data points. It provides further evidence for the model's good performance but also reveals areas where the performance can be fine-tuned for improvement.

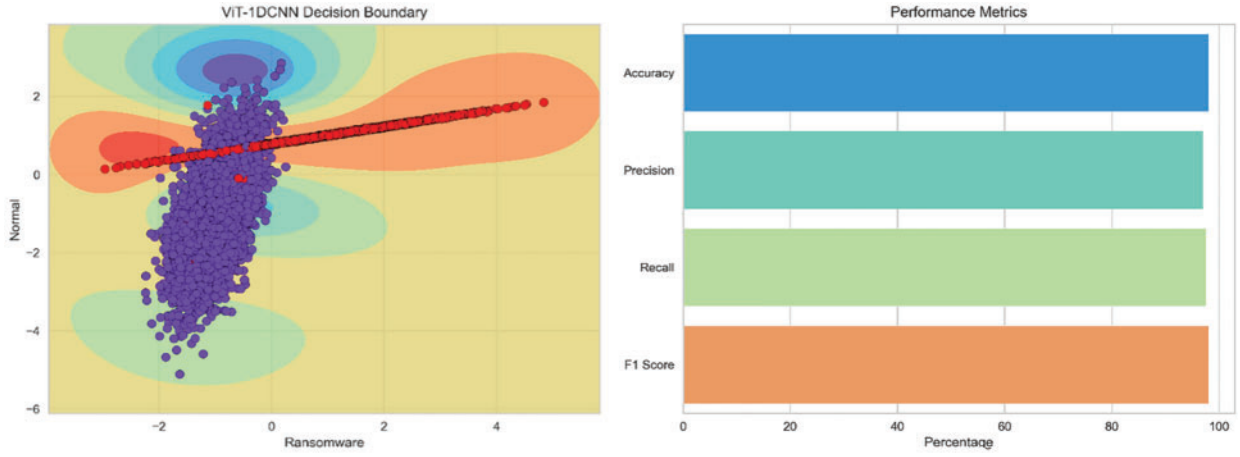

Figure 12: ViT-1DCNN decision boundary (left) and performance assessment metrics (right)

4.1 Comparison of Hybrid and Individual Models

In comparison with individual models of vision transformer and convolutional neural network, the ViT-1DCNN model outperformed with significantly better results of all the metrics as shown in Table 4.

Table 4: Performance comparison between hybrid and individual models

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
ViT-1DCNN	98	97	97.5	98
ViT	95	94	93.5	95
1DCNN	92	91.5	90.2	91.8

Fig. 13 shows the training and validation curves for the number of training instances for all the models. The hybrid model also outperformed here in terms of model training and validation scores. To avoid overfitting, k-fold cross-validation was used to assess the effectiveness of each model. It is a very reliable and common method in which data is divided into k subsets. Each of the k subsets is used as the validation set, while the other k-1 subsets are used as the training set. This process is repeated k times or k-folds so that each subset is used as a validation set once. In each fold, the data sampling is randomized to form different training and validation sets. This shuffling helps in adding the variability to each training and testing instance which can be noticed by the fluctuating training and validation performance as the model is trained on different data in each fold. Hence, the model's learning experience differs across folds leading to variation in training accuracy rates. The

purpose of cross-validation is to establish how well the given model performs on data that has not been encountered before and hence can give a better estimate of the model's performance for real-life applications.

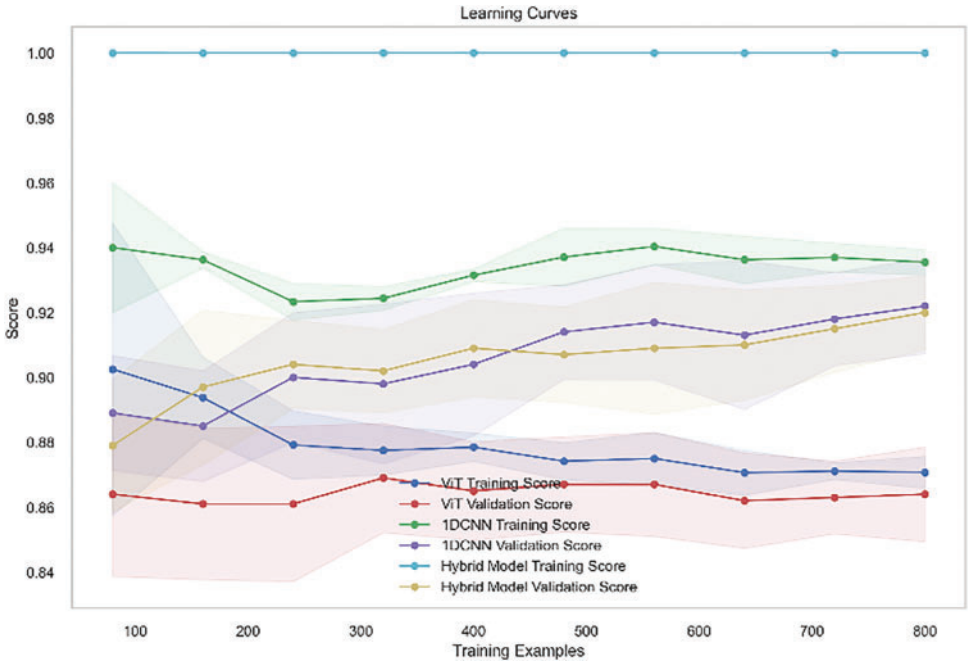

Figure 13: Learning curves for all models (training and validation scores)

In Fig. 14, ROC curves for all the models are presented which show true positive rate against false positive rate for each classifier. It measures the model's discriminatory power for identifying correct ransomware predictions against misclassified instances and can be quantified by AUROC values. The highest AUROC value of 0.97 for ViT-1DCNN validates its best performance in comparison to the individual ViT and 1DCNN models.

Figs. 15 and 16 represent the confusion and precision matrices for hybrid and individual models, respectively. Insights into a classification model's efficiency can be determined from the confusion matrix by counting the number of correct (TP), incorrect (TN), false positive (FP), and false negative (FN) predictions. All the performance assessment metrics results are generated from these confusion matrices according to the formulas given in Table 3. Higher values of true predictions show higher accuracy and precision of the ViT-1DCNN model as compared to individual models as it benefits from both the spatial and sequential nature of network traffic data. The findings of this comparison provide evidence for the effectiveness of the proposed approach in enhancing network security and identifying ransomware attacks. The results support the idea that hybrid architecture is more beneficial than ViT or 1DCNN individually.

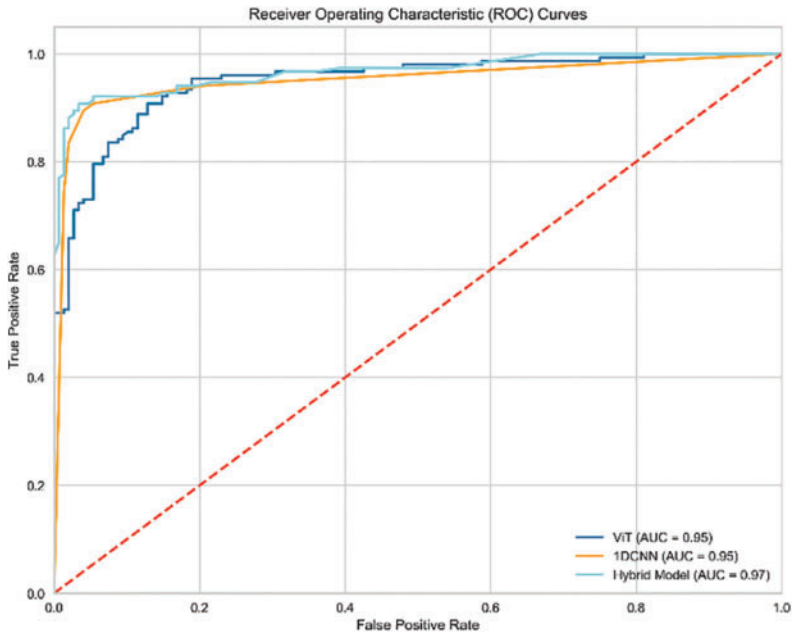

Figure 14: Receiver operating characteristics curves for hybrid and individual models

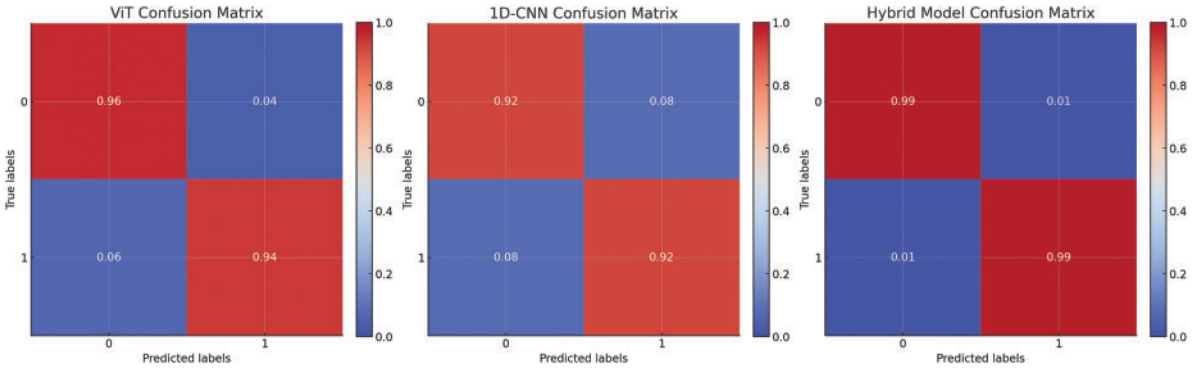

Figure 15: Comparative confusion matrices for hybrid and individual models

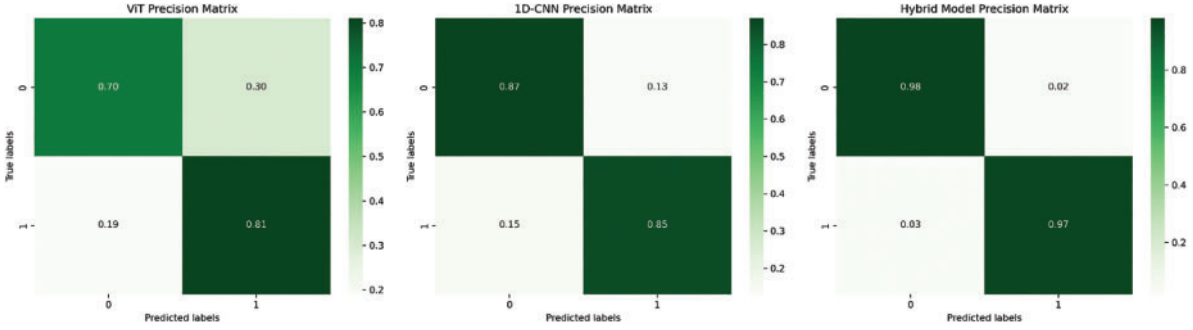

Figure 16: Comparative precision matrices for hybrid and individual models

4.2 Comparison with Previous Studies

Apart from standalone models, ViT-1DCNN is also compared against previous research works conducted in ransomware detection. Its performance is validated against other models such as RF, SVM, LR, DL, etc. The proposed hybrid model provided higher accuracy, precision, recall, and F1-score than the other models in terms of recognizing the ransomware and managing sequential data. The comparison in Table 5 establishes that the proposed model is more efficient and flexible than the other research for ransomware detection.

Table 5: Comparison of the proposed model with published research works

Ref.	Dataset	Technique	Accuracy	Advantages
[11]	Ransomware	RF	92%	Feature importance, generalization
[14]	Ransomware	DL	94.5%	Feature learning, adaptability
[15]	Ransomware	SVM with RBF	89%	Kernel flexibility, margin maximization
[18]	Ransomware	LR	88.5%	Simplicity, interpretability
Proposed model	Ransomware	ViT-1DCNN	98%	Sequential data handling, high accuracy

Note: RBF, radial basis function.

4.3 Computational Cost and Resource Comparison

To further strengthen the generalization of the findings, it is crucial to compare the computational resources and the overall resources required by the proposed hybrid model with standalone and other best practices models. This comparison will thus help in determining the feasibility level of the proposed approach for real-life applications. The computational cost and resource comparison of ViT-1DCNN with others is shown in Table 6.

Table 6: Computational cost and resource comparison among different models

Model	Training time (h)	Inference time (ms/sample)	Memory usage (GB)	Number of parameters (million)
ViT-1DCNN	20	50	8.5	30
ViT	12	40	6.5	18
1DCNN	10	30	5.0	15
RF [11]	2	5	1.0	0.1
DL [14]	15	35	7.0	25
SVM [15]	5	10	2.0	0.5
LR [18]	1	5	0.5	0.05

ViT-1DCNN takes the longest training time of 20 h as compared to ViT and 1DCNN with 12 and 10 h, respectively. As it integrates both architectures and other steps like feature fusion and dimensionality reduction, it makes it a challenging task. The time taken to make an inference for each sample is 50 ms for the hybrid model which is slower than ViT at 40 ms and 1DCNN at 30 ms. The increase in inference time is due to more steps involved in the processing of both the sequential and spatial features at the same time. The proposed model utilizes 8.5 GB of memory for training and inference while ViT and 1DCNN take 6.5 and 5.0 GB, respectively. This rise in memory utilization is attributed to the requirements of storing and processing both feature sets. The proposed model has 30 million parameters which is the highest in comparison to ViT and 1DCNN which have 18 million and 15 million parameters, respectively. It happened because two DL architectures have been integrated which affected the model size and its parameters. LR model [18] is computationally efficient with the least training time (1 h), inference time (5 ms/sample), and memory usage (0.5 GB). However, it has significantly fewer parameters (0.05 million) than the other models which may affect its capacity to learn the complexities involved in the application and prevent it from solving many tasks.

As ViT-1DCNN gave the best performance and results for ransomware detection as compared to the other models, it also resulted in a significantly increased overhead of computational resources and was time-consuming for both the training and inference. These factors of model complexity and resource utilization should not be overlooked in real-world applications so tradeoff can be done with the significantly best model performance. Future work may concern the fine-tuning of this hybrid model to make it computationally less expensive without significant compromise on the performance. This may include strategies like model pruning, quantization, or better methods of feature fusion and dimensionality reduction. Furthermore, the development of other architectures as potential candidates to achieve a balance between advanced ransomware detection and reduction of computational cost will be a significant step for advanced ransomware detection systems in real-world scenarios.

4.4 Performance Comparison with Oversampling Technique

To overcome the class imbalance problem, it is crucial to compare ViT-1DCNN with ViT, 1DCNN, and other models. The comparison in Table 7 shows how various models approach the problem of class imbalance and how performance increases with oversampling methods.

Table 7: Comparison of various models in handling data imbalance

Model	Oversampling technique	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Comments
ViT-1DCNN	SMOTE	98	97	97.5	98	Improved performance but synthetic samples may not fully represent real samples.
ViT	None	95	94	93.5	95	Struggles with minority class detection without oversampling.
1DCNN	None	92	91.5	90.2	91.8	Performs well in the majority class but has a poor recall in the minority class.

(Continued)

Table 7 (continued)

Model	Oversampling technique	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Comments
RF [11]	Random oversampling	92	90	88	89	Random oversampling leads to overfitting.
DL [14]	SMOTE	94.5	93	92.5	93	Better than no oversampling but still limited by synthetic data quality.
SVM with RBF [15]	Adaptive synthetic sampling	89	87	85	86	Adaptive sampling improves recall but increases complexity.
LR [18]	None	88.5	87	86	86.5	Limited by data imbalance, leading to poor minority class detection.

The proposed model used SMOTE which enhanced the performance metrics, however, the synthetic samples generated by it may not mimic real samples very well which may impact the model's performance. ViT-1DCNN outperformed all the models with the highest results of performance assessment metrics. ViT and 1DCNN models achieved good metrics results in identifying the majority class, however, lack of oversampling impacts their capability of identifying the minority class which comes with a low recall value. RF over-samples randomly to achieve an accuracy of 92%, precision of 90%, recall of 88%, and F1-score of 89% [11]. Random oversampling may cause overfitting, and it impairs the model's capability of being generalized. Adaptive synthetic sampling with RBF kernel-based SVM [15] enhances the recall and other metrics but at the expense of model complexity and computational requirements.

On the baseline model without any oversampling technique, LR [18] attains an accuracy of 88.5%, 87% precision, 86% recall, and 86.5% F1-score. The model's accuracy is constrained by the data skewness resulting in a lower prediction rate of minor class. For ransomware detection, by using SMOTE for class imbalance, the proposed hybrid model ViT-1DCNN achieves better performance metrics. Other techniques like adaptive sampling, ensemble methods, and sophisticated oversampling strategies may be explored to enhance the performance of the proposed model. Using realistic datasets with balanced samples that represent the target populations may also improve the model's training data, and hence its overall performance.

4.5 Robustness Testing

Robustness testing of ViT-1DCNN was conducted to enhance the credibility of the proposed model. It involved the model's performance evaluation under different scenarios, datasets, and types of ransomware attacks to assess its generalizability and reliability for real-world applications. The robustness testing was carried out using the following variations:

- UNSW-NB15 dataset: The primary dataset used in the study.

- CICIDS 2017 dataset: A dataset comprising a mix of normal and malicious traffic, including ransomware.
- CTU-13 dataset: A dataset focused on botnet traffic, including ransomware samples.

Types of ransomware attacks include the following:

- File-encrypting ransomware: Ransomware that encrypts files on the affected system.
- Locker ransomware: Ransomware that locks the system's interface to prevent user access.
- Scareware: Fake software that claims to detect issues and demands payment to fix them.

Different scenarios that were tested are mentioned below:

- Normal traffic volume: Standard volume of network traffic.
- High traffic volume: Increased volume of network traffic to simulate peak usage times.
- Low traffic volume: Reduced volume of network traffic to simulate off-peak times.

Table 8 presents the performance metrics-based comparison among the above-mentioned scenarios and dataset in robustness testing for ViT-1DCNN.

Table 8: Robustness testing results for multiple datasets and scenarios

Scenario/Dataset	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	AUROC
UNSW-NB15	98	97	97.5	98	0.97
CICIDS 2017	96	95	95.2	95.1	0.96
CTU-13	94	93	92.8	92.9	0.94
File-encrypting	97	96	96.5	96.2	0.97
Locker ransomware	95	94	94.5	94.2	0.95
Scareware	96	95	95.3	95.1	0.96
Normal traffic volume	98	97	97.5	98	0.97
High traffic volume	95	94	94.8	94.4	0.95
Low traffic volume	97	96	96.4	96.2	0.97

ViT-1DCNN achieved the highest performance results of all the metrics for the UNSW-NB15 dataset as compared to the CICIDS 2017 and CTU-13 datasets. Although the metric results are lower for the other two datasets, the proposed model achieved significant and reliable results on them with an average accuracy of 95% and almost 94% average results of other metrics which proved the model's efficiency and versatility. Despite the different difficulty levels of these datasets, the excellent performance of the proposed model shows its high discriminating power for ransomware identification and viability for application to real-world network security domains. Fig. 17 illustrates the confusion matrices for ViT-1DCNN tested on three datasets. TP and TN rates are high while the FP and FN rates are low which is evidence of the model's validity and transferability in identifying ransomware in different networks.

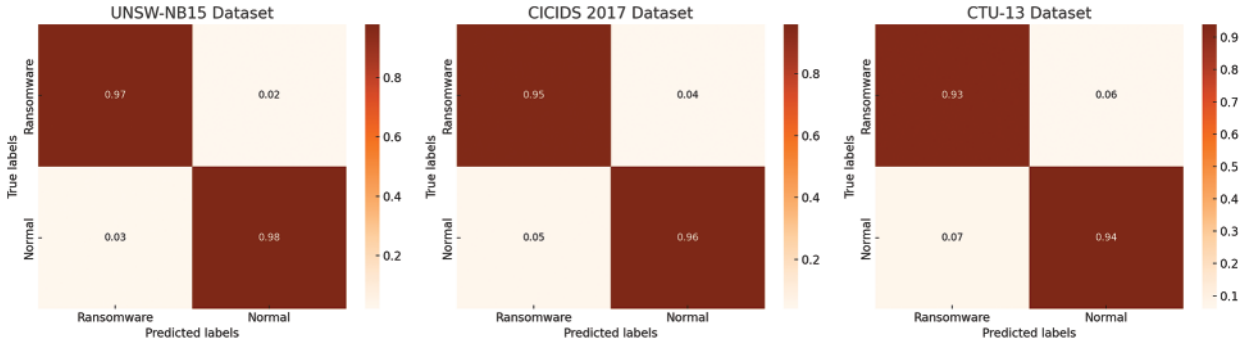

Figure 17: Confusion matrices for ViT-1DCNN performance over multiple datasets

For file-encrypting ransomware, the model achieved 97% accuracy which is good for identifying this attack. For locker ransomware and scareware, the model attained 95% and 96% accuracy, respectively which ascertains the outperformance of the proposed hybrid model. The comparative analysis of ViT-1DCNN testing at eight different traffic load levels concluded that it showed good performance at low and medium traffic while the long-term performance was unsatisfactory at full load. Under normal data traffic conditions, the model achieved an accuracy of 98%, however, the performance dropped to 95% upon an increase in data traffic over the network. This may be due to exposure of the model to more noise and/or error. The generalization and robustness of the ViT-1DCNN model are thoroughly illustrated and proved. The model responses were found to be stable even for small changes in the input environment. To enhance the explanation and credibility of the model, its working mechanism is required to be understood with essential features that play a role in identifying ransomware.

4.6 Computational Efficiency vs. Performance

Although the hybrid ViT-1DCNN model achieved higher results of accuracy, precision, recall, and F1-score for ransomware detection in comparison to the other models, it has higher model complexity with reduced computational efficiency which might pose challenges in real-world applications particularly in limited resource environments. The computational demands of the hybrid model primarily factored upon the high dimensionality and complexity of ViT. Its ability to capture global patterns also leads to a huge increase in both memory and time consumption. In contrast to 1DCNN which operates on sequential data, ViT uses multi-headed self-attention mechanisms which results in higher GPU and memory usage especially when dealing with big data in network traffic.

4.7 Impact on Practical Implementation

Due to the computational complexity of ViT, real-time ransomware detection in large network environments or on edge devices becomes a problem. For instance, in applications that need quick and instant detection like IoT networks, devices with limited resources cannot be able to support the GPU or the memory required to implement the full hybrid model. This hybrid model also takes a longer time during the training process as compared to other simple models. The combination of ViT and 1DCNN in conjunction with the feature fusion process increases training time and computational complexity. This increases the model's efficiency, but at the same time makes it difficult to retrain which may be necessary to account for new ransomware variants. It is especially important when operating in environments where ransomware variants are being released very often. The longer training times related to the proposed hybrid model may be a drawback for firms that need to deploy new models

faster. This is rather disadvantageous for the hybrid approach since such threat landscapes require frequent changes and updates to counter the attackers.

4.8 Computational Efficiency vs. Performance Trade-off

The major disadvantage of the hybrid model is the tradeoff between computational costs and improved results. In large-scale, corporate networks or cloud environments where resources are not a major concern the higher computational overhead of the hybrid model is more easily justifiable given the high stakes involved in the detection of ransomware attacks. The advantages of such models are more important in these situations because the organization may avoid the dangerous and expensive consequences of ransomware attacks. In limited resource conditions including the IoT networks, edge devices, or small-scale systems, the computational cost of the hybrid model may be excessive. The high memory and processing requirements can be a major drawback to deployment. Even if the system is deployed, the latency could affect real-time threat detection and make it less usable. These are situations where lightweight models or possibilities to decrease the hybrid model's resource utilization will be essential. In real-life scenarios, high accuracy and adaptability come with a tradeoff for low latency, limited resources, and a dynamic environment having high network traffic and varying threats. The feature set can be modified to include features relevant to distributed denial-of-service (DDoS) attacks. These modifications may increase the feasibility of the model for deployment in other areas of cybersecurity.

5 Conclusions

In this work, a new model is introduced that integrates a vision transformer (ViT) and a one-dimensional convolutional neural network (1DCNN) for the early and accurate identification of ransomware threats in the network traffic. This proposed hybrid model ViT-1DCNN specializes in capturing global as well as local patterns of network data. Experimental results of the study show that this model is statistically significant and performs better than the standalone ViT and 1DCNN models in terms of accuracy, precision, recall, and F1-score. ViT-1DCNN achieved 98% accuracy, 97% precision, 97.5% recall, and 98% F1-score and outperformed the baseline models in more than one testing environment. These findings support the use of deep learning architectures in the identification of ransomware based on the features investigated while working with big network data. A major concern in this area is the class imbalance between benign and ransomware network traffic which can lead to faulty predictions. Synthetic minority oversampling technique (SMOTE) was used to balance the dataset during data preparation. The differences in the results before and after the SMOTE application showed that the minority class instances, inclusive of ransomware traffic, were better detected by the model while minimizing biases. Moreover, the employed feature fusion strategy also improved the detection performance by combining different features obtained from ViT and 1DCNN to provide a better representation of the input data. It led to the enhanced stability of the model, and it became possible to make changes in traffic patterns and ransomware types. Since the hybrid model is a combination of multiple models, it had favorable results with higher memory consumption and extended model training times. These questions bring into concern its size and practicality for implementation on edge computing devices or real-time detection systems. In the future, the model architecture can be fine-tuned through pruning or by using lightweight versions of ViT to minimize the computational load with minimal impact on the model's efficiency. The current results are encouraging, but more features can be included, and feature selection methods can be fine-tuned to prevent overfitting and performance degradation when the network topology changes. Overall, the proposed hybrid ViT-1DCNN model is an improvement in ransomware detection due to

its ability to improve performance in imbalanced datasets and the integration of multiple features. This work provides the basis for improved detection in cybersecurity and opens the door for future work in model optimization for practical implementation in real-time scenarios. As deep learning techniques continue to advance, this hybrid model can help to set the direction for further advancements in network traffic analysis and malware detection systems.

Acknowledgement: Authors acknowledge the support of University of Sargodha and Najran University for the preparation and publication of the manuscript.

Funding Statement: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Author Contributions: The authors confirm their contribution to the paper as follows: conceptualization, methodology, Muhammad Armghan Latif, Zohaib Mushtaq and Saad Arif; validation, formal analysis, investigation, Muhammad Armghan Latif, Saad Arif, Zohaib Mushtaq, Saifur Rahman and Haris Aziz; writing—original draft preparation, Muhammad Armghan Latif, Saad Arif, Zohaib Mushtaq and Salim Nasar Faraj Mursal; writing—review and editing, Saad Arif, Saifur Rahman, Muhammad Irfan and Salim Nasar Faraj Mursal; supervision, resources, funding acquisition, Zohaib Mushtaq, Muhammad Irfan, Salim Nasar Faraj Mursal and Saifur Rahman. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Publicly available datasets were analyzed in this study. This data can be found here: https://research.unsw.edu.au/projects/unsw-nb15-dataset (accessed on 23 September 2024).

Ethical Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Alvee SRB, Ahn B, Kim T, Su Y, Youn YW, Ryu MH. Ransomware attack modeling and artificial intelligence-based ransomware detection for digital substations. In: 2021 6th IEEE Workshop on the Electronic Grid (eGRID), 2021; New Orleans, LA, USA; p. 1–6.
2. Lee SJ, Shim HY, Lee YR, Park TR, Park SH, Lee IG. Study on systematic ransomware detection techniques. In: 2022 24th International Conference on Advanced Communication Technology (ICACT), 2022; Pyeongchang, Republic of Korea; p. 297–301.
3. Almousa M, Basavaraju S, Anwar M. API-based ransomware detection using machine learning-based threat detection models. In: 2021 18th International Conference on Privacy, Security and Trust (PST), 2021; Auckland, New Zealand; p. 1–7.
4. Manavi F, Hamzeh A. Static detection of ransomware using LSTM network and PE header. In: 2021 26th International Computer Conference, Computer Society of Iran (CSICC), 2021; Tehran, Iran; p. 1–5.
5. Usha G, Madhavan P, Cruz MV, Vinoth NAS, Nancy M. Enhanced ransomware detection techniques using machine learning algorithms. In: 2021 4th International Conference on Computing and Communications Technologies (ICCCT), 2021; Chennai, India; p. 52–8. doi:10.1109/ICCCT53315.2021.9711906.
6. Jayanthi MSM, Vijayakumar K. Detection and decryption of ransomware. In: 2023 2nd International Conference on Applied Artificial Intelligence and Computing (ICAAIC), 2023; Salem, India; p. 1264–7. doi:10.1109/ICAAIC56838.2023.10140747.

7. Almomani I, AlKhayer A, Ahmed M. An efficient machine learning-based approach for Android v.11 ransomware detection. In: 2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA), 2021; Riyadh, Saudi Arabia; p. 240–4. doi:10.1109/CAIDA51941.2021.9425059.
8. Ashraf M, Asif M, Ahmad MB, Ayaz A, Nasir A, Ahmad U. Towards classification and analysis of ransomware detection techniques. In: 2023 4th International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), 2023; Sukkur, Pakistan; p. 1–5. doi:10.1109/iCoMET57998.2023.10099204.
9. Min D, Ko Y, Walker R, Lee J, Kim Y. A content-based ransomware detection and backup solid-state drive for ransomware defense. IEEE Trans Comput Aided Des Integr Circuits Syst. 2022;41(12):3290–302.
10. Tsunewaki K, Kimura T, Cheng J. LSTM-based ransomware detection using API call information. In: 2022 IEEE International Conference on Consumer Electronics, 2022; Taiwan; p. 211–2.
11. Baek S, Jung Y, Mohaisen D, Lee S, Nyang D. SSD-assisted ransomware detection and data recovery techniques. IEEE Trans Comput. 2020;70(10):1762–76.
12. Wang B, Liu H, Han X, Xuan D. RanPAS: A behavior-based system for ransomware detection. In: 2021 IEEE Sixth International Conference on Data Science in Cyberspace (DSC), 2021; Shenzhen, China; p. 309–14. doi:10.1109/DSC53577.2021.00049.
13. Jiao J, Zhao H, Liu Y. Analysis and detection of Android ransomware for custom encryption. In: 2021 IEEE 4th International Conference on Computer and Communication Engineering Technology (CCET), 2021; Beijing, China; p. 220–5. doi:10.1109/CCET52649.2021.9544366.
14. Zhang X, Wang J, Zhu S. Dual generative adversarial networks based unknown encryption ransomware attack detection. IEEE Access. 2022;10:105489–98.
15. Zhuravchak D, Ustyianovych T, Dudykevych V, Venny B, Ruda K. Ransomware prevention system design based on file symbolic linking honeypots. In: 2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 2021; Cracow, Poland; p. 284–7. doi:10.1109/IDAACS53288.2021.9660913.
16. Li J, Niu J, Qian X, Liu Y. Real-time ransomware detection method based on TextGCN. In: 2023 6th International Conference on Artificial Intelligence and Big Data (ICAIBD), 2023; Chengdu, China; p. 535–41. doi:10.1109/ICAIBD57115.2023.10206378.
17. Hsu CM, Yang CC, Cheng HH, Setiasabda PE, Leu JS. Enhancing file entropy analysis to improve machine learning detection rate of ransomware. IEEE Access. 2021;9:81994–2006.
18. Alotaibi FM, Vassilakis VG. SDN-based detection of self-propagating ransomware: the case of BadRabbit. IEEE Access. 2021;9:141418–28.
19. Yamany B, Azer MA, Abdelbaki N. Ransomware clustering and classification using similarity matrix. In: 2022 2nd International Mobile, Intelligent, and Ubiquitous Computing Conference (MIUCC), 2022; Cairo, Egypt; p. 41–6. doi:10.1109/MIUCC55081.2022.9781655.
20. Charmilisri A, Harshi I, Madhushalini V, Raja L. A novel ransomware virus detection technique using machine and deep learning methods. In: 2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS), 2023; Madurai, India; p. 8–14. doi:10.1109/ICICCS56967.2023.10142938.
21. AlMajali A, Qaffaf A, Alkayid N, Wadhawan Y. Crypto-ransomware detection using selective hashing. In: 2022 International Conference on Electrical and Computing Technologies and Applications (ICECTA), 2022; Ras Al Khaimah, United Arab Emirates; p. 328–31. doi:10.1109/ICECTA57148.2022.9990424.
22. Gao C, Shahriar H, Lo D, Shi Y, Qian K. Improving the prediction accuracy with feature selection for ransomware detection. In: 2022 IEEE 46th Annual Computers, Software, and Applications Conference (COMPSAC), 2022; Los Alamitos, CA, USA; p. 424–5. doi:10.1109/COMPSAC54236.2022.00072.
23. Rahman S, Mursal SNF, Latif MA, Mushtaq Z, Irfan M, Waqar A. Enhancing network intrusion detection using effective stacking of ensemble classifiers with multi-pronged feature selection technique. In: 2023 2nd International Conference on Emerging Trends in Electrical, Control, and Telecommunication Engineering (ETECTE), 2023; Lahore, Pakistan; p. 1–6. doi:10.1109/ETECTE59617.2023.10396717.

24. Singh A, Mushtaq Z, Abosaq HA, Mursal SNF, Irfan M, Nowakowski G. Enhancing ransomware attack detection using transfer learning and deep learning ensemble models on cloud-encrypted data. Electronics. 2023;12(3899):1–15. doi:10.3390/electronics12183899.
25. Singh A, Abosaq HA, Arif S, Mushtaq Z, Irfan M et al. Securing cloud-encrypted data: detecting ransomware-as-a-service (RaaS) attacks through deep learning ensemble. Comput Mater Contin. 2024;79(1):857–73. doi:10.32604/cmc.2024.048036.
26. Urooj U, Al-Rimy BAS, Zainal AB, Saeed F, Abdelmaboud A, Nagmeldin W. Addressing behavioral drift in ransomware early detection through weighted generative adversarial networks. IEEE Access. 2024;12:3910–25. doi:10.1109/ACCESS.2023.3348451.
27. Ispahany J, Islam MR, Islam MZ, Khan MA. Ransomware detection using machine learning: a review, research limitations and future directions. IEEE Access. 2024;12:68785–813. doi:10.1109/ACCESS.2024.3397921.
28. Ferdous J, Islam R, Mahboubi A, Islam MZ. AI-based ransomware detection: a comprehensive review. IEEE Access. 2024;12(6):136666–95. doi:10.1109/ACCESS.2024.3461965.
29. Hernandez-Jaimes ML, Martínez-Cruz A, Ramírez-Gutiérrez KA, Guevara-Martínez E. Enhancing machine learning approach based on nilsimsa fingerprinting for ransomware detection in IoMT. IEEE Access. 2024;12(2):153886–97. doi:10.1109/ACCESS.2024.3480889.
30. Marcinkowski B, Goschorska M, Wileńska N, Siuta J, Kajdanowicz T. MIRAD: a method for interpretable ransomware attack detection. IEEE Access. 2024;12(10):133810–20. doi:10.1109/ACCESS.2024.3461322.
31. Hill JE, Walker TO, Blanco JA, Ives RW, Rakvic R, Jacob B. Ransomware classification using hardware performance counters on a non-virtualized system. IEEE Access. 2024;12(4):63865–84. doi:10.1109/ACCESS.2024.3395491.
32. Rana MU, Shah MA, Al-Naeem MA, Maple C. Ransomware attacks in cyber-physical systems: countermeasure of attack vectors through automated web defenses. IEEE Access. 2024;12(6):149722–39. doi:10.1109/ACCESS.2024.3477631.
33. Liu Y, Fan H, Zhao J, Zhang J, Yin X. Efficient and generalized image-based CNN algorithm for multi-class malware detection. IEEE Access. 2024;12:104317–32.
34. Lee S, Kim J, Seo M, Na SH, Shin S, Kim J. CENSor: detecting illicit bitcoin operation via GCN-based hyperedge classification. IEEE Access. 2024;12:152330–46.
35. Khanan A, Mohamed YA, Mohamed AHHM, Bashir M. From bytes to insights: a systematic literature review on unraveling IDS datasets for enhanced cybersecurity understanding. IEEE Access. 2024;12(6):59289–317. doi:10.1109/ACCESS.2024.3392338.