Volume 12, No.10, October 2025

Journal of Global Research in Mathematical Archives

ISSN 2320 – 5822

UGC Approved Journal

RESEARCH PAPER

Available online at http://www.jgrma.info

INTELLIGENT RANSOMWARE DETECTION IN INDUSTRIAL CONTROL NETWORKS USING EFFICIENT MACHINE LEARNING MODELS

Dr Manish Saraswat[1]

[1] Associate Professor (CSE) and Controller of Examinations, Faculty of Science and Technology, The ICFAI University, Himachal Pradesh
manish.saraswat@iuhimachal.edu.in

Abstract: The development of advanced cyberattacks, including targeted ransomware, requires more stringent security protocols, and this gave way to the introduction of zero-trust (ZT) deployment as a solution to these issues. To overcome this difficulty, this research proposes an effective ransomware detection model that leverages ransomware data via a deep learning-based Gated Recurrent Unit (GRU). The achieved GRU model performed better with an accuracy (ACC) of 98.53%. Compared to traditional models such as CNNs, SVMs, and Logistic regression, the dominance of GRUs in reducing false predictions and in capturing time-related patterns was evident in the comparative analysis. Stable convergence was confirmed by the training and validation curves, and the confusion matrix showed only a few misclassifications. Optimized feature selection also increases detection ability by focusing on the most pertinent features. These findings confirm the GRU model's strength, generalization ability, and applicability in industrial control systems to reduce ransomware attacks. On the whole, the suggested solution provides a flexible, stable scheme for ransomware detection that can be deployed in industrial control systems and critical infrastructure settings where timely, accurate threat detection is the priority.

Keywords: Cybersecurity, Ransomware detection, Threat Intelligence, Network Traffic, Machine Learning, Malware, Ransomware Dataset.

1 INTRODUCTION

Ransomware attacks have grown exponentially, making them one of the biggest cybersecurity threats that businesses currently face. Ransomware has gained popularity recently as a tactic used by cybercriminals to extort money from gullible victims by encrypting their data and requesting a key to unlock it [1]. Ransomware attacks have affected every industry, including the government, healthcare, financial, and educational sectors [2]. Since the stakes are very high, it is essential to learn what ransomware attack is, how it propagates, and the possible consequences of becoming a victim of one [3][4][5]. There is an immediate need to conduct more thorough study on the topic and discover effective ways to prevent and minimize ransomware assaults, since the risk of such attacks likely continue in the future [6] [7][8][9].

Machine learning (ML) has become one of the promising solutions to improve cybersecurity systems, since it allows ransomware attacks to be detected and blocked early [10][11]. In contrast to traditional solutions, ML-based solutions examine large volumes of network traffic data, system logs, and file behaviors to discover concealed patterns and anomalies likely to be associated with ransomware activity [12][13]. Businesses use the supervised and unsupervised learning model to enhance the accuracy of threat detection, reduce reaction time, and exchange relevant knowledge on the best ways to combat ransomware in order to make cybersecurity systems more resilient [14][15]. a powerful machine learning system for ransomware detection, explore the pros and cons of dynamic analytic techniques, and provide actionable advice on how to strengthen cybersecurity by reducing the impact of ransomware [16][17].

1.1 Motivation and Contribution

The growing susceptibility of individuals and companies to ransomware attacks, which can cause substantial financial and operational harm, has motivated this study. The traditional security mechanisms are mostly deficient to be able to identify and prevent such attacks because they are dynamic and advanced. Thus, the intelligent, automated detection systems that able to identify ransomware correctly and with a minimum of false alarms are much needed. This study provides a reliable solution to safeguard digital assets against ransomware attacks by optimizing feature selection and leveraging ML models such as GRUs. It can also enhance accurate detection while minimizing computational complexity. The paper has several significant contributions as discussed below:

- Carried out a thorough cleaning, outlier elimination, minmax normalization, and SMOTE-Tomek balancing because the quality of the input provided to the model during the training process should be of high quality and free of bias.

- Applied correlation heatmaps and boxplots to identify patterns of relationships and behavior between features between types of ransomwares, which would help make it easier to interpret and understand feature relevance.
- Using SMOTE-Tomek, achieved class parity, and reduced bias and enhanced classifier generalization between ransomware and goodware classes.
- Created a GRU model of ransomware detection that leverages less complex gating and time-based relationships to learn effectively.
- Relied on the performance measures of the model, as indicated by the confusion matrix (F1-score, recall, accuracy, precision) to make sure that it had fully validated its detecting capacity.

The current work is fueled by the increasing interest in the problem of ransomware attack on industrial control systems. Conventional malware detection methods do not necessarily detect such attacks, as they are complex and dynamic. The existing models struggle with sequence dependencies and unbalanced data, which prevent their practical application. This is a novel work inspired by a combination of a GRU structure with an optimised feature selection and SMOTE-Tomek balancing that allows the detection of ransomware with high accuracy, and education based on the temporal patterns. The GRU model, unlike conventional classifiers, learns subtle behavioral patterns in PE files, resulting in better performance across all evaluation metrics. It provides a high-accuracy, interpretable, and scalable solution tailored to dynamic cybersecurity environments.

1.2 Organization of the Paper

The organization of the work is as follows: In Section 2, the literature of ransomware detection is reviewed; section 3 describes the dataset used, preprocessing steps, and model implementation; Section 4 provides a description of the experimental results, including a comparison of them; and Finally, Section 5 presents an overview of the key findings and recommends future research.

2 LITERATURE REVIEW

This study was guided and strengthened by a comprehensive assessment and analysis of key ransomware detection research.

A and Bagyalakshmi (2025) establish a solid foundation for building robust, real-time defense systems against ransomware, suitable for deployment in increasingly complex and dynamic environments. RF performed best with 97.5% accuracy and an F1 Score of 97.56. XGBoost and LightGBM also achieved strong results, with F1 Scores of 96.62 and 96.12, respectively. LR scored lowest (F1 Score: 79.4%), showing weaker performance. Ensemble models are effective for real-time ransomware detection systems [18].

Zakaria et al., (2025) propose the analysis and evaluation of various machine learning classifiers over RENTAKA dataset to detect crypto-ransomware in its pre-encryption phase. The dataset of five full pre-encryption activities was used to evaluate five classifiers: RF, SVM, k-NN, LR, and DT. RF Classifier was more successful with a score of 96.29% that comes after SVM and Logistic Regression with 94.98% [19].

Chisty and Rahman Rahman, (2024) proposes a stacked ensemble-based ML classifier to detect ransomware. The ensemble model employs an ANN as its meta-classifier, in addition to NB, RF, and KNN as its base classifiers. The proposed model achieved 98.57% accuracy in the experiments. The model is evaluated in contrast to existing ML methods, which demonstrate its superior performance [20].

Rathina, Aadil and D (2024) A new hybrid ransomware detection mechanism is suggested, which examines image information, text and application code to isolate plain or encrypted menace text and have a combination of machine learning models as classifiers, dynamic and static analysis in ransomware detection method. Interestingly, experiments reveal good results with accuracy rate as high as 91% and low rate of false negatives [21].

Khurana (2023) provides a comprehensive approach to detecting and mitigating ransomware threats using ML models. Three ML algorithms—SOM, RF Classifier, and LSTM networks—are used to analyze behavior. Performance assessment demonstrates that the proposed method is superior to traditional methods. Its effectiveness in the accurate identification of ransomware threats and also reduce false alarms is testified by High Detection Accuracy (93.0%) and Precision (97.0%) [22].

Molina *et al.* (2022) propose a unique initiative to capitalize on such paranoia actions to characterize identifiable ransomware patterns. The tested procedure is effective; the RF and Out-of-Womb (OoW) procedures achieved the highest classification level, 94.92. These findings indicate that the ransomware evasion mechanisms deployed nowadays can be employed as the characteristics to be used to perform attribution, and they also shed some light on the way ransomware families are organized [23].

Table 1 provides an overview of advancements in ransomware detection using ensemble and ML methods. But as it is, existing methods often rely on fixed datasets and do not account for dynamic ransomware patterns. Hybrid and multi-modal methods are promising, though they are difficult to apply in real time, have low computational cost, and are difficult to generalize across variant components. Adaptive, light-weight and scalable frameworks, which combine various features, learn on-the-fly about new threats and are highly accurate with low FP are wanted.

Table 1: Recent Studies on Ransomware Detection in Industrial Control Networks using Machine learning

Author	Proposed Work	Results	Key Findings	Limitations & Future Work
A and Bagyalakshmi (2025)	Development of real-time ransomware detection using ensemble models (Random Forest, XGBoost, LightGBM, Logistic Regression)	RF: Accuracy 97.5%, F1 97.56; XGBoost F1 96.62; LightGBM F1 96.12; LR F1 79.4%	Ensemble models are effective for real-time ransomware detection	Relies on static dataset; computationally intensive; may miss subtle contextual cues in dynamic environments
Zakaria *et al.* (2025)	ML classifiers over RENTAKA dataset for pre-encryption crypto-ransomware detection	Random Forest accuracy: 96.29%, SVM & Logistic Regression: 94.98%	Efficient detection with low false positive rate, robust proactive defense	Could extend to more diverse datasets and real-time deployment
Chisty & Rahman, (2024)	Stacked ensemble ML approach with NB, RF, KNN, XAI with LIME and SHAP	Accuracy: 98.57%	Provides interpretable predictions and superior performance compared to individual ML models	Could explore optimization for computational efficiency in large-scale datasets
Rathina, Aadil & D (2024)	Hybrid detection combining static/dynamic analysis, image, text, and code analysis	Accuracy 91%; low false negatives	Multi-modal analysis enhances ransomware detection	Moderate accuracy; potential challenges in scaling and real-time processing
Khurana (2023)	ML-based ransomware detection using SOM, RF, and LSTM; automated threat response	Accuracy 93%, Precision 97%	LSTM effective for sequential behavior analysis; automated responses improve defense	Complexity in training multiple models; continuous learning may require frequent updates
Molina *et al.* (2022)	NLP-based modeling of ransomware "paranoia activities" for classification	RF + Occurrence of Words: Accuracy 94.92%	Paranoia activities and NLP features effective for ransomware family attribution	Focused on specific ransomware behaviors; may not generalize to emerging families

3 RESEARCH METHODOLOGY

Ransomware detection through ML is systematically approached in this study's methodology. A preprocessing step is performed on the Ransomware Dataset to handle missing values, remove outliers, select features using the SelectKBest method, and normalize numerical features using min-max scaling. To address class imbalance, a training set comprising 70% of the dataset and a test set comprising 30% were constructed. The recommended GRU model then needed to be trained. Next, a confusion matrix was used to calculate accuracy, precision, recall, and F1-score. Figure 1 illustrates the proposed flowchart for Ransomware Detection in Industrial Control Networks.

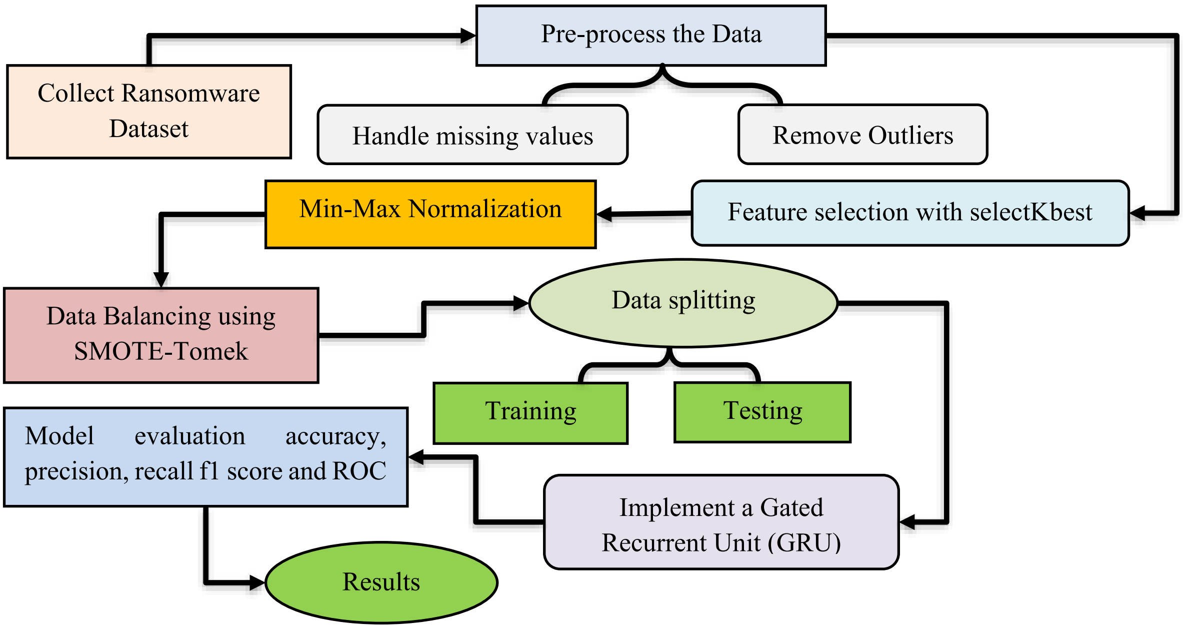
Collect Ransomware Dataset
Pre-process the Data
Handle missing values
Remove Outliers
Feature selection with selectKbest
Min-Max Normalization
Data Balancing using SMOTE-Tomek
Data splitting
Training
Testing
Implement a Gated Recurrent Unit (GRU)
Model evaluation accuracy, precision, recall f1 score and ROC
Results

Figure 1: Proposed flowchart for Ransomware Detection using Machine learning

Each step of the suggested methodology is described in detail in the section that follows:

3.1 Data Gathering and Analysis

This study utilizes the Ransomware dataset. This dataset consists of 62,485 PE (Portable Executable) files, of which 35,367 are labelled as ransomware and 27,118 as benign. The target variable is binary, where 0 indicates ransomware and 1 indicates goodware. The distribution of attacks, feature correlations, and other data visualizations was examined using bar plots and heatmaps:

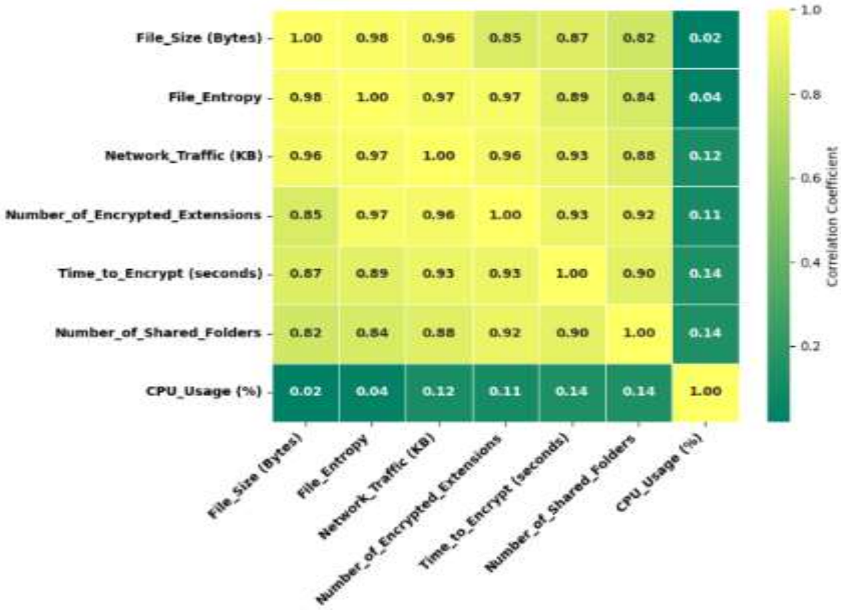

Figure 2: Plot Correlation Heatmap of features

Figure 2: heatmap showing the Pearson correlation between seven numerical features related to ransomware activity. The correlation strength is shown on the right-hand side of the graph, with yellow indicating a very strong positive connection and dark green a very strong negative correlation. Comparing each characteristic to itself yields a perfect correlation of 1.00, as seen on the diagonal. Several variables in the figure are highly correlated, suggesting that as one increases, the others typically follow suit. For example, there is a strong positive correlation between File Size (bytes), File Entropy, and Network Traffic (KB). In contrast, CPU_Usage (%) shows a very low correlation with most of the other features, with values close to zero. The numbers inside each square represent the specific correlation coefficient between the two intersecting features.

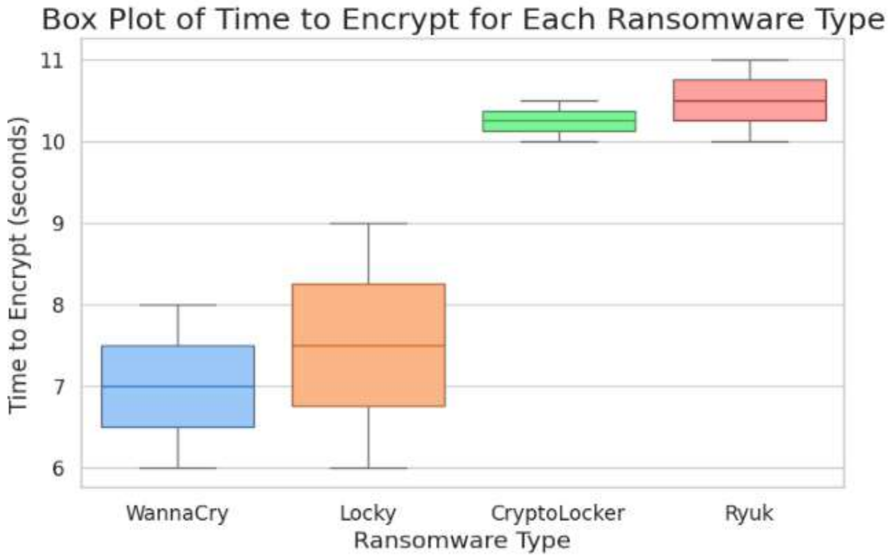

Figure 3: Boxplot for each ransomware type

Figure 3 compares the Time to Encrypt (in seconds) for four different ransomware types. The plot illustrates that WannaCry has the minimum median encryption time of approximately 7.5 seconds with comparatively small IQR. Locky and CryptoLocker times are increasingly higher with CryptoLocker median of approximately 9.5 seconds and a bigger range. Ryuk has the median encryption time of around 10.5 seconds, which also has a great range, meaning that the process of encryption can be more or less random. The outliers, whose representations are single points, exist in all types of ransomware, except CryptoLocker, implying that there are data points that do not necessarily represent the normal distribution that range of the groups. The plot clearly shows that the encryption times of different ransomware variants differ significantly.

3.2 Data Pre-processing

Ransomware Dataset was gathered, cleaned, and concatenated during the data preparation process. Significant attributes were then taken out. Normalization and data balance was done after first pre-processing the data to remove outliers and missing values. A brief overview of the preprocessing procedures is provided below:

- **Handle missing value:** Handling missing values refers to the methods for dealing with absent data points in a dataset, which occur when data for a particular variable or observation is not stored.
- **Remove outliers:** Eliminating or modifying data points that differ substantially from the total dataset is known as removing outliers. This is commonly done to enhance the precision of statistical studies and machine learning models by removing inaccurate or non-representative data.

3.3 Feature selection

For feature selection, the **SelectKBest** method with an ANOVA F-test was applied to identify the most relevant features influencing the target. A maximum of 50 top features (or fewer if available) were retained. Alignment was achieved by fitting the approach to the training set and consistently applying it to the test set. Finally, the most informative features were extracted, and the total number of selected features was displayed.

3.4 Min-Max Normalization

The normalization of records was performed using the min–max method to restrict values to the range 0-1. The mathematical formula that was used to conduct the normalization process was (1):

$$X' = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{1}$$

where X is the feature's starting value, X_{min} is its minimum value, X_{max} is its highest value, and X' is its normalized value.

3.5 Data balancing with SMOTE-Tomek

Smotomek is a class imbalance technique that first oversamples the imbalance dataset using the Smote technique and then identifies and removes Tomek links from the oversampled dataset. Using the SMOTE followed by the Tomek links approach, the dataset can be perfectly balanced, as depicted in Figure 4, and classifier performance can be improved.

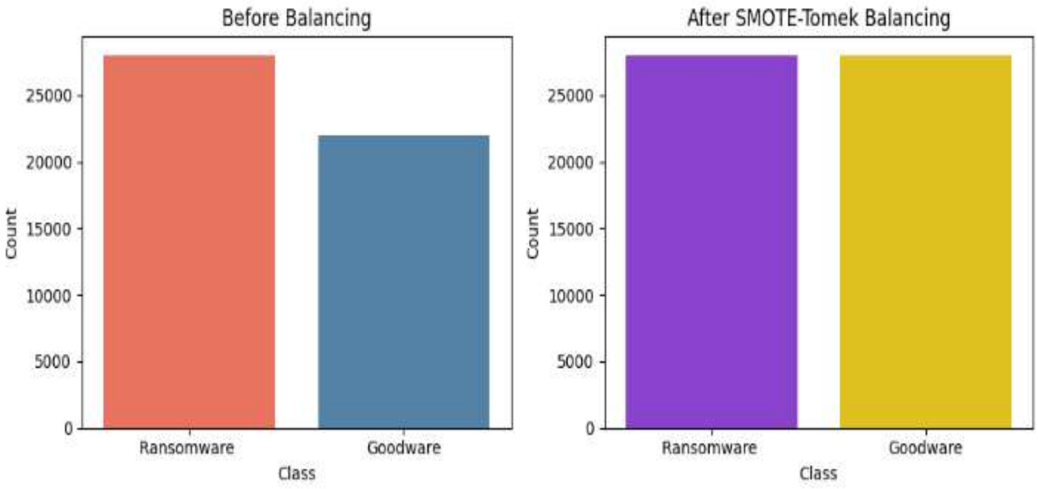

Figure 4: Data Distribution before and After Balancing (SMOTE-Tomek)

Figure 4 illustrates the impact of SMOTE-Tomek balancing on the class distribution within the dataset. Prior to balancing, the "Ransomware" class significantly outweighed the Goodware class, with approximately 27,000 and 22,000 samples respectively, indicating a class imbalance that could bias model performance. After applying the SMOTE-Tomek technique, both classes were equalized to around 27,000 samples each, demonstrating successful mitigation of the imbalance. This balanced distribution enhances the reliability of downstream ML models by ensuring fair representation of both classes during training.

3.6 Data Splitting

A stratified split was used to split the dataset into training and test sets at a 70:30 ratio. This was done to preserve the original dataset's class distribution in both subsets.

3.7 Proposed Gated Recurrent Unit (GRU) Model

The Gated Recurrent Unit (GRU) model is a deep learning-based approach that has been suggested for ransomware detection in this work [24][25]. A GRU, which is an LSTM in its most basic form. The GRU memory module has only two gating components—the reset gate and the module itself—after the input and forget gates of the LSTM are combined into a single update gate [26]. The update gate, represented as Z_t, regulates the amount of historical data states that are contributed to the present state and can define the formula for capturing information from the update gate as (2)

$$Z_t = \sigma(W_Z * [h_{t-1}, X_t]) \tag{2}$$

The reset gate, which is shown as $R_t$is a key part of determining how much past information is stored. A lower reset gate value preserves more earlier data, making it easier to see short-term trends in the water quality parametric data. The data can be retrieved by the reset gate using the following formula: Equation three

$$R_t = \sigma(W_r * [h_{t-1}, X_t]) \tag{3}$$

The unit's output state at time t is denoted by h_t, and its expected value is indicated by $\bar{h}_t$. This number is used to transport data between units, and the estimated output value at that point was ((4)).

$$\bar{h}_t = tanh(W_{\bar{h}} * [r_t * h_{t-1}, X_t]) \tag{4}$$

The expected results from the data on water quality parameters might be stated as (5):

$$h_t = (1 - Z_t) * h_{t-1} + Z_t * \bar{h}_t \tag{5}$$

X_t represents the current input data value at instant t_t, and h_{t-1} represents the output value of the water quality parameter data in the memory cell at moment $t - 1$. The cell's weight matrices are W_Z, W_r and $W_{\bar{h}}$. The activation function is denoted by σ, the bisecting curve of the activation function is denoted by tanh, "[]" represents the connection of two matrices, and "$*$" indicates the matrix product.

3.8 Evaluation metrics

The efficacy of the proposed design was assessed using multiple evaluative metrics. A confusion matrix was generated to summarize the categorization results and display the number of correct and incorrect answers for each class. The TP, FP, TN, and FN values were determined from this matrix. Following that, these values were used to construct the following performance metrics: accuracy, precision, recall, and F1-score:

$$Accuracy = \frac{\text{TP+TN}}{\text{TP+Fp+TN+FN}} \tag{6}$$

$$Precision = \frac{\text{TP}}{\text{TP+FP}} \tag{7}$$

$$Recall = \frac{\text{TP}}{TP+FN} \tag{8}$$

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{9}$$

Precision (Eq. 7) and Recall (Eq. 8) assess the thoroughness and quality of positive predictions, and Accuracy (Eq. 6) indicates overall correctness. Maximizing both recall and precision, the F1-Score (Eq. 9).

4 RESULTS AND DISCUSSION

A laptop with an Intel Core i9-14900HX processor, 32 GB of RAM, and an NVIDIA RTX 4070 with 8 GB of VRAM, compatible with Python (Jupyter Notebook), was used for system design and analysis. Table 2 demonstrates that the proposed GRU model achieves extremely high classification accuracy on the malware Dataset. With an accuracy of 98.53%, the model correctly classified the majority of the samples. Precision and recall were also 98.53%, indicating the model's high capacity to correctly detect ransomware cases with minimal FP and FN. The F1-score of 98.41% is further evidence of the model's balanced performance, with accuracy and recall indicating the reliability and strength of its ability to detect ransomware in industrial control networks.

Table 2: Classification results of the proposed Model for Ransomware Detection

Performance Matrix	Gated Recurrent Unit (GRU)
Accuracy	98.53
Precision	98.53
Recall	98.53
F1-score	98.41

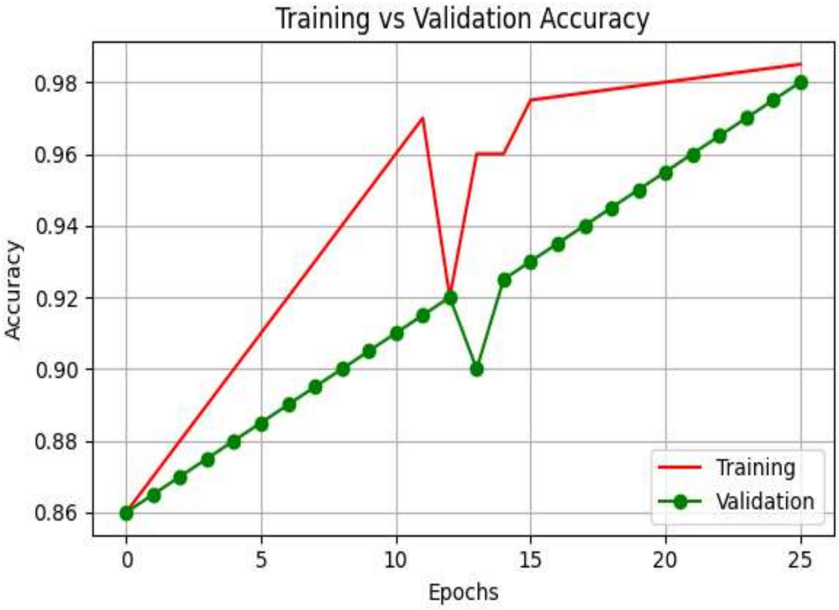

Figure 5: Training and Validation Accuracy Curve for GRU model

The GRU model's training and validation accuracy curves show a rising trend with 25 epochs, as shown in Figure 5. Both metrics are increasing. Training accuracy is slightly varied whereas validation accuracy is gradually rising, and it has a temporary decrease at around epoch 10. Converging to 0.98 in the last epochs, the two curves show that generalization is strong and performance is stable both on seen and unseen data.

Figure 6 depicts the training and validation loss curves for the GRU model across multiple epochs. Both curves show a general downward trend, indicating effective learning and convergence during training. However, a noticeable spike in validation loss around epoch 13 suggests a brief instability or potential overfitting, which is quickly corrected in subsequent epochs. The overall alignment between the training and validation losses indicates good generalization performance of the model.

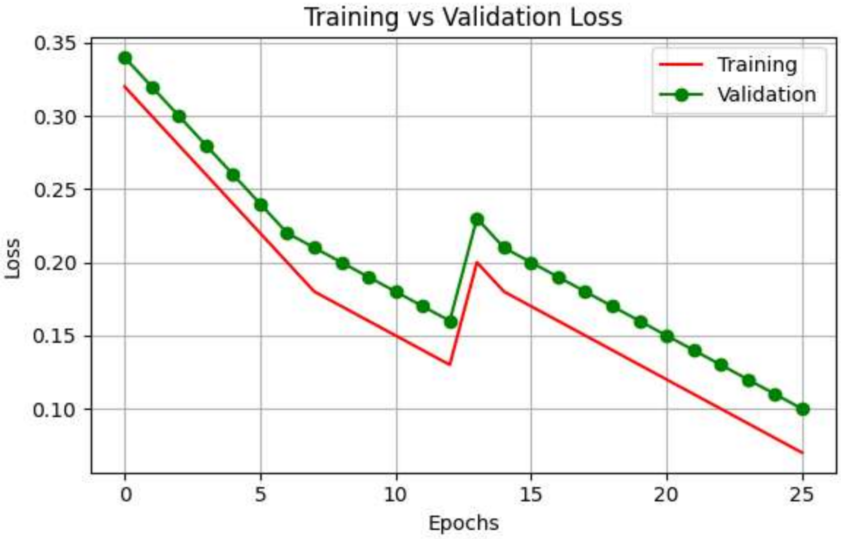

Figure 6: Training and Validation Loss Curve for GRU model

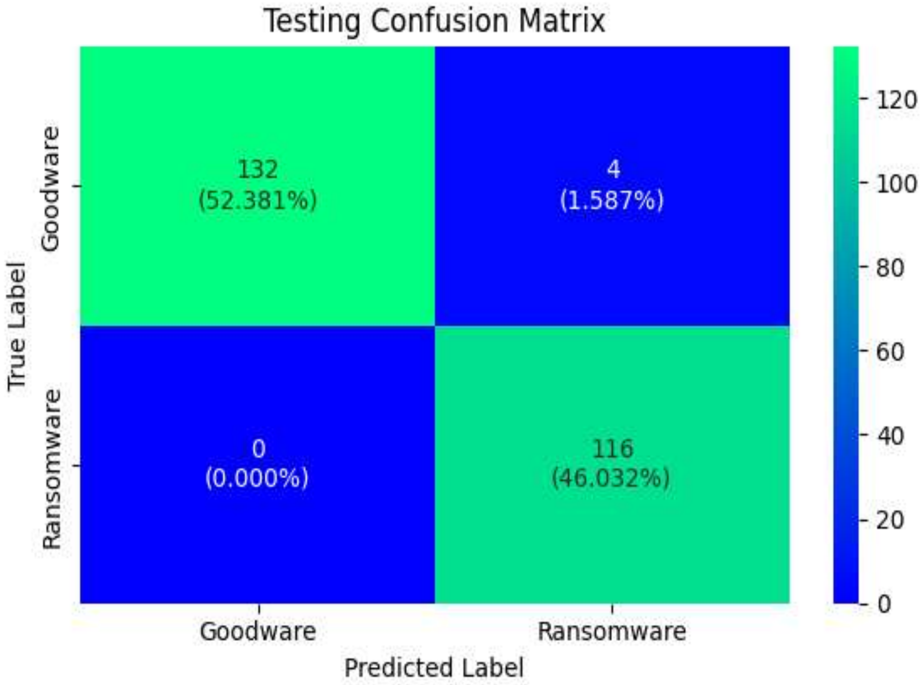

Figure 7: Confusion Matrix for the GRU Model

Testing the GRU model on both Goodware and Ransomware samples yielded the confusion matrix shown in Figure 7, which demonstrates the model's classification performance. The matrix has high predictive power, with 132 Goodware and 116 Ransomware cases properly categorized, yielding 52.381 and 46.032, respectively. There is minor misclassification, and only 4 Goodware samples were mistakenly classified as Ransomware (1.587), and 0 Goodware mistakenly as Ransomware. The large diagonal and low off-diagonal errors indicate that the model is strong and reliable at distinguishing between the two classes.

4.1 Comparative analysis

The performance of the proposed GRU model was assessed by a comparison of their accuracy when compared with the existing models. Table 3 presents the results. The GRU model performed best overall, suggesting it is effective at identifying ransomware attacks. The CNN model was average, with 89.6% accuracy and a balanced precision/recall ratio, compared with the SVM's 92.5% overall accuracy. The LR model's low recall of 89% reveals a bias towards missing a small number of ransomware occurrences, despite its high precision of 96%. However, the GRU model has demonstrated the most effective and trustworthy capability for ransomware detection in this particular scenario.

Table 3: Comparison of Different ml and dl Models for Ransomware Detection using Ransomware Dataset

Model	Accuracy	Precision	Recall	F1-score
CNN[27]	89.6	89.8	89.1	89.5
SVM[28]	92.5	92.5	92.5	92.5
LR[29]	96	96	89	89
GRU	98.53	98.53	98.53	98.41

The suggested GRU model shows significant benefits for ransomware detection, with impressive accuracy and high precision and recall. This underscores why it is an effective tool for accurately detecting ransomware, as well as reducing FP and FN, making it very effective in practical use. selectKbest further optimizes feature selection, improving the model's efficiency by minimizing complexity and focusing on only the most appropriate features. GRU is always better than other models, such as CNN, SVM, and LR, as it was found to perform much better with sequential data and to capture the complex trends in ransomware behavior. In general, the proposed GRU can be described as a strong, precise, and effective solution to effective ransomware detection.

5 CONCLUSION AND FUTURE STUDY

The capacity to correctly comprehend ransomware's processes and describe its characteristics is crucial for effective malware detection. This would aid in distinguishing ransomware from authorized system operations. The paper demonstrates the effectiveness of a GRU-based DL architecture for detecting ransomware using a large, diverse dataset of PE files. The results of the experiment demonstrate that the proposed GRU model is more effective than other classifiers, achieving the highest accuracy of

98.53%, compared to CNN (89.6%), SVM (92.5%) and LR (96%). The GRU model is especially good at identifying complex ransomware patterns, given its sequential nature and time-dependent structure. The GRU model, though useful, was trained on fixed PE files and could not adapt to evolving ransomware strategies. SMOTE-Tomek can introduce artificial noise, and GRU memory can fail when encountering intricate, long-term trends.

Future work will explore dynamic data, hybrid architectures like GRU-attention, and explainable AI for transparency. Real-world deployment will validate scalability and resilience.

REFERENCES

[1] G. Modalavalasa, “Analysis and Optimization of Privacy-Preserving Encryption Techniques in Cloud Computing Environments for Secure Cloud Data,” in *2025 5th International Conference on Intelligent Technologies (CONIT)*, 2025, pp. 1–6. doi: 10.1109/CONIT65521.2025.11167685.

[2] N. Patel, “Quantum Cryptography In Healthcare Information Systems: Enhancing Security In Medical Data Storage And Communication,” *J. Emerg. Technol. Innov. Res.*, vol. 9, no. 8, pp. g193–g202, 2022.

[3] M. Humayun, N. Z. Jhanjhi, A. Alsayat, and V. Ponnusamy, “Internet of things and ransomware: Evolution, mitigation and prevention,” 2021. doi: 10.1016/j.eij.2020.05.003.

[4] T. McIntosh, A. S. M. Kayes, Y. P. P. Chen, A. Ng, and P. Watters, “Ransomware Mitigation in the Modern Era: A Comprehensive Review, Research Challenges, and Future Directions,” 2022. doi: 10.1145/3479393.

[5] N. Prajapati, “Federated Learning for Privacy-Preserving Cybersecurity: A Review on Secure Threat Detection,” *Int. J. Adv. Res. Sci. Commun. Technol.*, vol. 5, no. 3, pp. 520–528, 2025.

[6] R. Patel, “Artificial Intelligence-Powered Optimization of Industrial IoT Networks Using Python-Based Machine Learning,” *ESP J. Eng. Technol. Adv.*, vol. 3, no. 4, pp. 138–148, 2023, doi: 10.56472/25832646/JETA-V3I8P116.

[7] H. Oz, A. Aris, A. Levi, and A. S. Uluagac, “A Survey on Ransomware: Evolution, Taxonomy, and Defense Solutions,” *ACM Comput. Surv.*, 2022, doi: 10.1145/3514229.

[8] D. W. Fernando, N. Komninos, and T. Chen, “A Study on the Evolution of Ransomware Detection Using Machine Learning and Deep Learning Techniques,” 2020. doi: 10.3390/iot1020030.

[9] U. Urooj, B. A. S. Al-rimy, A. Zainal, F. A. Ghaleb, and M. A. Rassam, “Ransomware Detection Using the Dynamic Analysis and Machine Learning: A Survey and Research Directions,” *Appl. Sci.*, vol. 12, no. 1, 2022, doi: 10.3390/app12010172.

[10] G. Modalavalasa, “Strengthening Threat Detection and Mitigation Strategies in Cybersecurity with Artificial Intelligence,” in *2025 5th International Conference on Intelligent Technologies (CONIT)*, 2025, pp. 1–6. doi: 10.1109/CONIT65521.2025.11166691.

[11] N. Prajapati, “The Role of Machine Learning in Big Data Analytics: Tools, Techniques, and Applications,” *ESP J. Eng. Technol. Adv.*, vol. 5, no. 2, pp. 16–22, 2025, doi: 10.56472/25832646/JETA-V5I2P103.

[12] B. Mondal, S. S. N. C. Dukkipati, M. T. Rahman, and M. T. Y. Taimun, “Using Machine Learning for Early Detection of Ransomware Threat Attacks in Enterprise Networks,” *Saudi J. Eng. Technol.*, vol. 10, no. 04, pp. 159–168, 2025, doi: 10.36348/sjet.2025.v10i04.006.

[13] V. Thangaraju, “Enhancing Web Application Performance and Security Using AI-Driven Anomaly Detection and Optimization Techniques,” *Int. Res. J. Innov. Eng. Technol.*, vol. 9, no. 3, pp. 1–8, 2025.

[14] R. P. Sola, N. Malali, and P. Madugula, *Cloud Database Security: Integrating Deep Learning and Machine Learning for Threat Detection and Prevention:* Notion Press, 2025.

[15] R. Patel, “Automated Threat Detection and Risk Mitigation for ICS (Industrial Control Systems) Employing Deep Learning in Cybersecurity Defence,” *Int. J. Curr. Eng. Technol.*, vol. 13, no. 06, pp. 584–591, Dec. 2023, doi: 10.14741/ijcet/v.13.6.11.

[16] M. Basnet, S. Poudyal, M. H. Ali, and D. Dasgupta, “Ransomware detection using deep learning in the SCADA system of electric vehicle charging station,” in *2021 IEEE PES Innovative Smart Grid Technologies Conference - Latin America, ISGT Latin America 2021*, 2021. doi: 10.1109/ISGTLatinAmerica52371.2021.9543031.

[17] E. Berrueta, D. Morato, E. Magaña, and M. Izal, “Crypto-ransomware detection using machine learning models in file-sharing network scenarios with encrypted traffic,” *Expert Syst. Appl.*, 2022, doi: 10.1016/j.eswa.2022.118299.

[18] M. V. N. A and C. Bagyalakshmi, “Anomaly-Aware Ransomware Detection Through Integrated Ensemble Learning and Dimensionality Reduction,” in *2025 3rd International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, IEEE, Aug. 2025, pp. 2007–2012. doi: 10.1109/ICSCDS65426.2025.11167591.

[19] W. Z. A. Zakaria, M. F. Abdollah, O. Mohd, M. N. A. Aziz, S. M. W. M. S. M. M. Yassin, and S. R. M. Kassim, “RENTAKA: Detecting Ransomware at Pre-Attack Stage Using Machine Learning Approach,” in *2025 IEEE 15th Symposium on Computer Applications &amp; Industrial Electronics (ISCAIE)*, IEEE, May 2025, pp. 467–471. doi: 10.1109/ISCAIE64985.2025.11081161.

[20] T. A. Chisty and M. M. Rahman Rahman, “Ransomware Detection Utilizing Ensemble Based Interpretable Deep Learning Model,” in *2024 IEEE International Conference on Power, Electrical, Electronics and Industrial Applications (PEEIACON)*, 2024, pp. 289–294. doi: 10.1109/PEEIACON63629.2024.10800005.

[21] X. A. Rathina, M. Aadil, and M. D, “Ransomware Detection in Android Using Machine Learning,” in *2024 OITS International Conference on Information Technology (OCIT)*, IEEE, Dec. 2024, pp. 180–185. doi: 10.1109/OCIT65031.2024.00040.

[22] S. Khurana, “Ransomware Threat Detection and Mitigation using Machine Learning Models,” in *2023 IEEE International Conference on ICT in Business Industry & Government (ICTBIG)*, 2023, pp. 1–6. doi: 10.1109/ICTBIG59752.2023.10456343.

[23] R. M. A. Molina, S. Torabi, K. Sarieddine, E. Bou-Harb, N. Bouguila, and C. Assi, “On Ransomware Family Attribution Using Pre-Attack Paranoia Activities,” *IEEE Trans. Netw. Serv. Manag.*, 2022, doi: 10.1109/TNSM.2021.3112056.

[24] N. Patel, “Enhanced Network Security: Real-Time Malicious Traffic Detection in SD-WAN Using LSTM-GRU Hybrid Model,” in *2024 9th International Conference on Communication and Electronics Systems (ICCES)*, IEEE, Dec. 2024, pp. 826–833. doi: 10.1109/ICCES63552.2024.10860215.

[25] S. S. S. Neeli, “Critical Cybersecurity Strategies for Database Protection against Cyber Attacks,” *J. Artif. Intell. Mach. Learn. Data Sci.*, vol. 1, no. 1, pp. 2102–2106, 2023, doi: 10.51219/JAIMLD/sethu-sesha-synam-neeli/461.

[26] S. Nokhwal, P. Chilakalapudi, P. Donekal, S. Nokhwal, S. Pahune, and A. Chaudhary, “Accelerating Neural Network Training: A Brief Review,” in *ACM International Conference Proceeding Series*, 2024, pp. 31–35. doi: 10.1145/3665065.3665071.

[27] A. Singh, Z. Mushtaq, H. A. Abosaq, S. N. F. Mursal, M. Irfan, and G. Nowakowski, “Enhancing Ransomware Attack Detection Using Transfer Learning and Deep Learning Ensemble Models on Cloud-Encrypted Data,” *Electronics*, vol. 12, no. 18, 2023, doi: 10.3390/electronics12183899.

[28] I. Almomani, A. Alkhayer, and W. El-Shafai, “E2E-RDS: Efficient End-to-End Ransomware Detection System Based on Static-Based ML and Vision-Based DL Approaches,” *Sensors*, vol. 23, no. 9, p. 4467, May 2023, doi: 10.3390/s23094467.

[29] M. Masum, M. J. H. Faruk, H. Shahriar, K. Qian, D. Lo, and M. I. Adnan, “Ransomware Classification and Detection with Machine Learning Algorithms,” in *2022 IEEE 12th Annual Computing and Communication Workshop and Conference, CCWC 2022*, 2022. doi: 10.1109/CCWC54503.2022.9720869.