Ransomware Detection Using Machine Learning Techniques

Indra Chaudhary
meindra.ic@gmail.com
College of Applied Business and Technology
Suyash Adhikari
suyashadhikari99@gmail.com
College of Applied Business and Technology

Article Info

Article history:

Received: March 24, 2024

Revised: July14, 2024

Accepted: July 29, 2024

Keywords:

Decision tree, machine learning, multi-layer perceptron, performance evaluation, ransomware detection, support vector machine.

How to Cite

Chaudhary, I., & Adhikari, S. (2024). Ransomware detection using machine learning techniques. *Researcher CAB: A Journal for Research and Development*, *3*(1), 96–114. https://doi.org/10.3126/rcab.v3i1.68424

Abstracts

The proliferation of ransomware attacks is a critical cybersecurity threat that organizations globally face. This situation necessitates effective prevention and mitigation strategies. These malicious programs encrypt data and extort payments, impacting various industries. They highlight the urgent need for robust defense mechanisms. Despite advancements in machine learning for ransomware detection, there is a notable gap in the comparative analysis of individual algorithms such as Decision Tree (DT), Support Vector Machine (SVM), and Multi-Layer Perceptron (MLP). This study aims to fill the gap by providing a comparative analysis of these algorithms. It focuses on using the UG Ransome dataset and key metrics including accuracy, precision, recall, and F-measure. The experiments were conducted using Python. The results demonstrate that the Decision Tree outperforms SVM and MLP across all metrics. It achieves an accuracy of 98.83%, precision of 99.41%, recall of 99.41%, and F-measure of 99.41%. SVM and MLP, on the other hand, achieved lower scores. These results highlight the Decision Tree's superior performance in capturing non-linear data relationships, which is crucial for ransomware detection. The major contribution of this study is the identification of the Decision Tree as a highly effective model for ransomware detection. It significantly outperforms other models. The findings suggest that the Decision Tree's ability to model complexities within the data makes it a robust and reliable tool for safeguarding systems against ransomware attacks. Future research should explore the Decision Tree's performance across diverse ransomware datasets, integrate ensemble learning, investigate adversarial machine learning techniques, and enhance real-time detection methods. These efforts will improve the robustness and applicability of machine learning-based ransomware detection systems.

Corresponding Author:
Suyash Adhikari
suyashadhikari99@gmail.com
College of Applied Business and Technology

Introduction

Ransomware attacks have become a critical cybersecurity concern for organizations worldwide (Celdrán et al., 2022). These malicious programs encrypt victims' data and extort ransom payments, significantly impacting various industries. The scale of the ransomware threat is substantial, with over a third of organizations globally experiencing attempted attacks in 2021, representing a 105% increase from 2020 (Griffiths, 2024). This rise is partly due to challenges in securing networks for remote and hybrid work environments, underscoring the urgent need for robust prevention and mitigation strategies. Despite a reported decrease in attack volume in 2022, cybercriminals continue to refine their tactics, employing "double extortion" schemes and other methods like Denial-of-Service attacks and targeted harassment (Griffiths, 2024). The average ransom payment reached $570,000 in 2021, and underreporting complicates estimating the true number of attacks, with public reports suggesting over 3,640 attacks occurred between May 2021 and June 2022 (Griffiths, 2024).

Recent data from Kaspersky (2024) reveals a concerning trend: a 30% global increase in targeted ransomware groups from 2022 to 2023. These groups focus on high-profile targets and leverage the Ransomware-as-a-Service (RaaS) model, allowing smaller affiliates access to their malware. The number of victims of these targeted attacks has also seen a staggering 71% rise (Kaspersky, 2024).

The problem statement underscores the substantial financial threat posed by ransomware attacks, which not only cause disruption and data loss but also challenge traditional signature-based detection methods with their evolving variants. In this context, machine learning emerges as a promising avenue for early detection, yet the relative effectiveness of different machine learning algorithms remains uncertain. Therefore, this research aims to address these problems by comparing the performance of decision tree (DT), support vector machine (SVM), and multilayer perceptron (MLP) classification models in terms of key evaluation metrics such as accuracy, precision, recall, and F-measure for ransomware detection. The research objectives thus focus on analyzing and comparing the effectiveness of these specific algorithms in detecting ransomware attacks.

Related Works

Nkongolo (2024) introduces a multifaceted analysis of ransomware activity in the cryptocurrency ecosystem, utilizing the UGRansome dataset. Their Ransomware Feature Selection Algorithm (RFSA) achieves notable performance metrics. They find that approximately 68% of ransomware incidents involve bitcoins (BTC) transactions, with average damages of 88.37 USD. TowerWeb commands the highest fee, while CryptoLocker has the lowest. Moreover, a positive correlation between ransomware duration and financial gains underscores the adaptability of ransomware demands, emphasizing the need for continuous cybersecurity adaptation.

Nkongolo et al. (2022) introduce a cloud-based method for zero-day attack classification using the UGRansome1819 dataset. Leveraging Amazon Web Services, they employ Ensemble Learning with a Genetic Algorithm optimizer, integrating Naive Bayes, Random Forest, and Support Vector Machine classifiers. UGRansome1819 outperforms CAIDA and UNSWNB-15 datasets, achieving a 1% classification ratio before and after optimization. Genetic Algorithm feature selection enhances computational efficiency, while additional data samples improve model accuracy. The study advocates for ensemble techniques to address single-classifier instability, achieving 100% specificity and sensitivity for threatening classes. Lastly, optimization boosts SVM model accuracy by 6%.

Machine learning is crucial for proactive cybersecurity, analyzing risks and swiftly addressing breaches (Abushark et al., 2022). With rising cybersecurity incidents, key issues like anomaly detection, vulnerability diagnosis, phishing, denial of service, and malware identification need effective solutions. This research evaluates machine learning-based intrusion detection systems using Multi-Criteria

Decision Making (MCDM) methods like analytical hierarchy process (AHP) and technique for order of preference by similarity to ideal solutions (TOPSIS) under fuzzy conditions to handle decision-making uncertainties, helping design more effective systems (Abushark et al., 2022).

Nkongolo et al. (2021) introduce the UGRansome dataset, derived from modern netflow data, to detect zero-day attacks efficiently. They demonstrate its effectiveness in minimizing false alarms and accurately identifying threats like UDP Scan, Razy, EDA2, and Globe malware through Ensemble Learning algorithms, notably Random Forest. The study highlights the cyclostationary nature of advanced persistent threats, predicting their future use of spamming and phishing techniques. Additionally, they identify the NP-Hard nature of achieving dataset balance due to the non-uniform distribution of threatening classes.

Machine learning (ML) is vital for proactive cybersecurity, quickly addressing threats and intrusions (Alharbi et al., 2021). Despite increased breaches, key issues like anomaly detection and malware require effective solutions (Alharbi et al., 2021). This study evaluates ML-based intrusion detection systems (IDS) using an AHP and TOPSIS under hesitant fuzzy conditions to improve decision-making and system effectiveness (Alharbi et al., 2021).

Network Intrusion Detection Systems (NIDS) are crucial for safeguarding data by identifying malicious activities and unauthorized access (Kayode-Ajala, 2021). This study evaluates various machine learning algorithms using the NSL-KDD dataset, with preprocessing steps including feature scaling and PCA, reducing 122 features to 20 principal components. Seven algorithms are assessed: Logistic Regression, K-Neighbors Classifier, Gaussian Naive Bayes, Linear Support Vector Classifier, Decision Tree Classifier, Random Forest Classifier, and PCA-variant Random Forest. The K-Neighbors Classifier performs best, achieving 98.05% training and 97.94% test accuracy. PCA effectively streamlines computation with minimal accuracy loss, though recall is slightly reduced. Key features identified include login attempts and contact rates with different destination hosts.

Liu & Lang (2019) highlight the significance of intrusion detection systems (IDS) in cybersecurity, noting their ongoing challenges in accuracy and false alarm reduction. They emphasize the effectiveness of machine learning methods in addressing these issues, particularly in distinguishing between normal and abnormal data with high accuracy and detecting unknown attacks. The authors propose a taxonomy for classifying IDS literature based on data objects, providing a valuable framework for cybersecurity research. Additionally, they underscore the growing importance of deep learning techniques in this domain. Finally, Liu & Lang discuss emerging challenges and future directions in IDS research (Liu & Lang, 2019).

Fernando et al. (2020) conducted a comprehensive survey on ransomware detection utilizing machine learning and deep learning algorithms. The authors highlight the urgent need to combat the destructive nature of ransomware and the challenge of reversing its infections. They underscore the escalating threat posed by ransomware, which is fueled by the proliferation of new variants and families in the cyber landscape. The study emphasizes the increasing importance of artificial intelligence in ransomware detection, particularly in identifying zero-day threats. Through a review of prominent research studies, the survey highlights the efficacy of machine learning and deep learning approaches in detecting ransomware malware. The authors also explore the impact of malware evolution on these detection methods and anticipate the future expansion of ransomware into IoT environments.

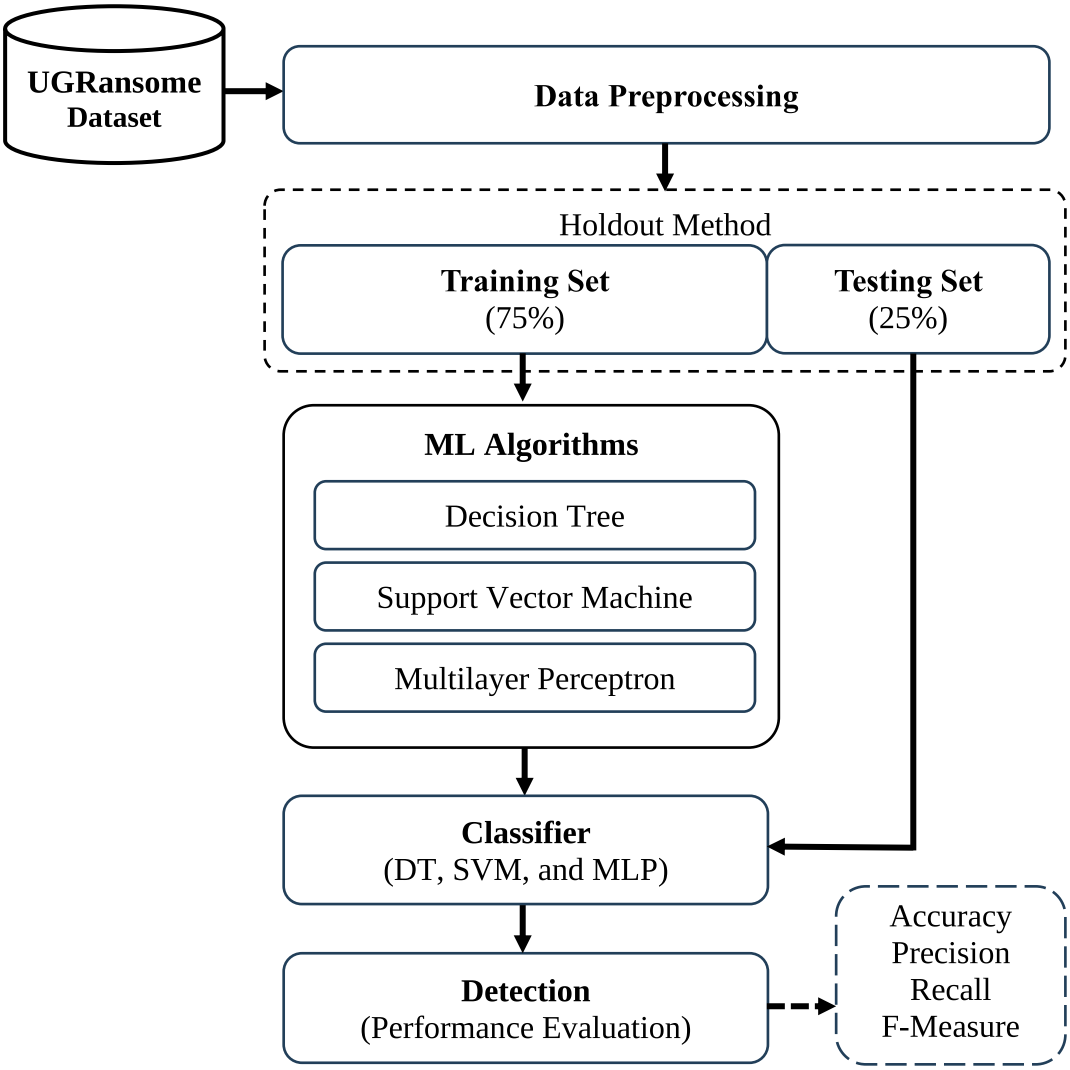
Sweet Bait is an automated system that combats fast-spreading computer worms by using honeypots to detect suspicious traffic and generate worm signatures, which are then distributed for immediate network protection and continuously refined for accuracy (Portokalidis & Bos, 2007). It prioritizes threats by monitoring signature activity, ensuring urgent worms are addressed promptly. Deployed in academic networks worldwide, sweet Bait can respond to zero-day worms within minutes and supports global signature sharing to enhance internet security.

Despite advancements in machine learning for ransomware detection, there are still gaps in the comparative analysis of individual algorithms such as DT, SVM, and MLP. Previous studies, such as those by Nkongolo (2024) and Nkongolo et al. (2022, 2021), have focused on ensemble methods and lack detailed evaluations of these individual models across key metrics like accuracy, precision, recall, and F-measure. The adaptability of DT, SVM, and MLP to evolving threats posed by ransomware has not been thoroughly explored. Furthermore, the research on ransomware-specific datasets and the practical deployment of these algorithms is insufficient. This study aims to address these gaps by comparing the performance of DT, SVM, and MLP models for ransomware detection using the UGRansome dataset and evaluating them based on key metrics.

Methodology

Figure 1 shows the outline of the research process. The research methodology utilizes the UGRansome dataset to enhance the detection and classification of ransomware and zero-day cyber-attacks (UGRansome Dataset, 2023). It involves rigorous data preprocessing to clean, handle missing values, and normalize the data. The dataset is split into training and testing sets, employing diverse machine learning algorithms (DT, SVM, and MLP) to build respective classifiers. The classifier is then applied for classification. Performance evaluation is conducted using metrics like accuracy, precision, recall, and F-measure to ensure reliable and significant findings. This systematic approach lays a solid foundation for advancing cybersecurity research in detecting and classifying emerging threats.

Figure 1. Block diagram of research methodology.



UGRansome Dataset

The UGRansome dataset serves as an extensive cybersecurity resource tailored for the analysis of ransomware and zero-day cyber-attacks, with a particular emphasis on those exhibiting periodic patterns (UGRansome Dataset, 2023). This dataset encompasses a variety of critical elements, including timestamps for precise attack time tracking, categorical flags for different attack types, and protocol data that elucidate the attack vectors employed. Moreover, it provides comprehensive details on network flows to observe data transfer patterns, classifications of ransomware families, and insights into associated malware, thereby enhancing the understanding of attack mechanisms. The dataset facilitates numeric clustering for effective pattern recognition and quantifies financial damage in both USD and BTC, offering a dual perspective on the economic impact of cyber-attacks.

By utilizing machine learning techniques, the UGRansome dataset is able to generate attack signatures, including synthetic signatures, that are extremely valuable for testing and simulating cybersecurity defenses (UGRansome Dataset, 2023). Additionally, the dataset facilitates research on anomaly detection and enhances cybersecurity readiness, making it an essential tool for researchers and practitioners who are committed to detecting and categorizing ransomware and zero-day threats.

This dataset contains 149,043 instances and 13 features with two classes on the target variable. In Table 1, the dataset is summarized, and the features are described in Table 2. The term "Signature or S" represents ransomware, while "Anomaly or A" represents no ransomware. Signatures (S) are patterns of known ransomware activities identified by predefined rules, while anomalies (A) are unusual network behaviors that are not ransomware and are flagged for their abnormal characteristics.

Table 1

Dataset Summary

Description	**Counts**
No. of Instances	149043
No. of Features	13
Positive Samples (S)	106482 (71.44%)
Negative Samples (A)	42561 (28.56%)

Table 2

Feature Description of Dataset

Feature Name	Data Type	Description
Time	Quantitative (Integers)	Timestamp of network attacks
Protocol	Qualitative (Categorical)	Network protocol used (e.g., TCP, UDP)
Flag	Qualitative (Categorical)	Network connection status (e.g., SYN, ACK)
Family	Qualitative (Categorical)	Network intrusion category
Clusters	Quantitative (Integers)	Event clusters or groups
SeddAddress	Qualitative (Categorical)	Formatted ransomware attack links (if applicable)
ExpAddress	Qualitative (Categorical)	Original ransomware attack links (if applicable)
BTC	Numeric	Values related to Bitcoin transactions in attacks (if applicable)
USD	Numeric	Financial damages in USD caused by attacks (if applicable)
Netflow_Bytes	Quantitative (Integers)	Bytes transferred in network flow
IPAddress	Qualitative	IP addresses associated with network events
Threats	Qualitative	Nature of threats or intrusions
Port	Quantitative	Network port number in events
Prediction	Qualitative (Categorical)	Predictive model outcome: Signature (S) represents Ransomware, and Anomaly (A) represents No Ransomware

Source: (UGRansome Dataset, 2023)

Data Pre-Processing

Initially, duplicate rows were removed from the dataset to ensure data integrity. Subsequently, negative values in the time column were rectified by shifting them. To mitigate skewness in the Netflow Bytes column, a logarithmic transformation was applied. Similarly, the right-skewed USD column was transformed using the square root function. Normalization was performed on the BTC column. Finally, categorical values were encoded into numerical values using label encoding methodology.

Holdout Method

To assess the classifier's performance, the holdout method is employed. This method involves dividing the original dataset into two subsets: a training set and a test set. The training set is utilized for constructing and training the data mining model, while the test set evaluates the model's ability to generalize to new data. In this study, classifier models were trained and tested using a split ratio of **75:25**.

Machine Learning Techniques

This study implemented three classification algorithms to train and test the classification models: Decision Tree (DT), Support Vector Machine (SVM), and Multilayer Perceptron (MLP).

Decision Tree

The **decision tree** is a prominent classification algorithm within data mining, leveraging a hierarchical tree-like structure to facilitate decision-making processes (Li et al., 2021; Quinlan, 1990). Decision trees are frequently utilized in operations research and intrusion detection (Li et al., 2021). Among the well-established methodologies for constructing decision trees are the ID3 and C4.5 algorithms, both of which utilize the concept of information entropy to generate the tree from a set of training data (Quinlan, 1986; Quinlan, 1992). The primary distinction between these algorithms lies in their feature selection criteria: ID3 predominantly uses information gain (Quinlan, 1986), whereas C4.5 employs the information gain ratio (Quinlan, 1992). This study used the ID3 algorithm (Li et al., 2021).

Support Vector Machine

The **Support Vector Machine** is a margin-based classification method that identifies an optimal hyperplane to maximize the separation between different classes, adhering to the principle of structural risk minimization. This principle endows SVM with robust generalization capabilities and resilience to overfitting issues (Gu & Lu, 2021). Furthermore, SVM effectively addresses non-linear classification problems by employing kernel functions, which map the original feature space to higher-dimensional spaces where the instances become linearly separable (Alam et al., 2020). Additionally, SVM can be utilized for novelty detection (Gu & Lu, 2021).

Multilayer Perceptron

According to Nosratabadi et al. (2021), the MLP is a type of neural network employing supervised learning with the back-propagation method. MLP has a three-layer structure comprising the input layer, hidden layer(s), and output layer(s), where each neuron connects to all neurons in the subsequent layer. MLP is commonly recognized for its effectiveness in addressing non-linear problems.

Performance Evaluation

According to Han and Kamber (2012), a confusion matrix is described as "a table used for analyzing the results of classifiers, highlighting how classifiers recognize tuples of different classes."

Figure 2.

Standard form of confusion matrix.

	Actual Positive (S)	**Actual Negative (A)**
Predicted Positive (S)	TP	FP
Predicted Negative (A)	FN	TN

When assessing the performance of the classification models, this study relies on established metrics as outlined by Han and Kamber (2012). Accuracy, as described in Equation 1, serves as a fundamental measure of our model's overall correctness in predictions. Precision, detailed in Equation 2, allows us to evaluate the accuracy specifically concerning positive predictions. Equally important is recall, defined in Equation 3, which gauges our model's capability to correctly identify true positives. Lastly, we utilize the F-measure, or F1-score, which harmoniously combines precision and recall to provide a comprehensive understanding of our model's performance, as expressed in Equation 4. These metrics, validated by Han and Kamber's framework, form the basis for the evaluation and comparison of classifiers in this study.

$$\text{Accuracy} = \frac{\boldsymbol{TP+TN}}{\boldsymbol{TP+TN+FP+FN}}\text{---}(1)$$

$$\text{Precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}}\text{---}(2)$$

$$\text{Recall} = \frac{TP}{TP+FN}\text{---}(3)$$

$$F - \text{Measure} = \frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}\text{---}\ (4)$$

Tools used

The tools utilized in this research encompass a robust arsenal for data analysis and model development. Scikit-learn stands out as a pivotal resource for classification tasks and model selection (Pedregosa et al., 2011), providing a comprehensive suite of algorithms and evaluation metrics. TensorFlow emerges as the cornerstone for training intricate deep learning models (Abadi et al., 2016), leveraging its flexibility and scalability to handle complex data. To manipulate and process data efficiently, Pandas (McKinney, 2010) and NumPy (Harris et al., 2020) serve as indispensable allies, offering powerful tools for data manipulation and numerical computation. Meanwhile, Matplotlib (Hunter, 2007) and Seaborn (Waskom et al., 2021) take the reins in visualizing data, enabling clear and insightful representations essential for interpreting and communicating research findings effectively. Together, these tools form a cohesive ecosystem empowering researchers to navigate the intricacies of data analysis and model development with confidence and precision.

Result Discussion

This study focuses on the effective implementation and performance analysis of SVM, DT, and MLP algorithms for ransomware detection. The evaluation metrics include accuracy, precision, recall, and F-measure. The experiments were conducted using Python version 3.9.18 on a system equipped with an Intel® Core™ i7-8550U

CPU and 16 GB RAM, running Windows 11 64-bit operating system. The performance of each model is presented in Table 3.

Table 3 shows the accuracy, precision, recall, and F-measure of three machine learning classification models for ransomware detection: SVM, DT, and MLP. DT has the highest overall performance according to all four metrics. It has an accuracy of 98.83%, precision of 99.41%, recall of 99.41%, and F-measure of 99.41%. SVM comes in second with an accuracy of 61.89%, precision of 72.27%, recall of 61.18%, and F-measure of 66.27%. MLP has an accuracy of 62.16%, precision of 61.84%, recall of 45.91%, and F-measure of 52.69%. Overall, the Decision Tree outperforms the other three models in ransomware detection.

Table 3

Classification Performance of Algorithms

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F-Measure (%)
SVM	61.89	72.27	61.18	66.27
DT	98.83	99.41	99.41	99.41
MLP	62.16	61.84	45.91	52.69

Source: Calculation based on Experiment

Table 4

Advantages of DT over SVM, and MLP in Ransomware Detection

Metric	DT (%)	Advantage over SVM (%)	Advantage over MLP (%)
Accuracy	98.83	36.94	36.67
Precision	99.41	27.14	37.57
Recall	99.41	38.23	53.50
F-measure	99.41	33.14	46.72

Source: Calculation based on Experiment

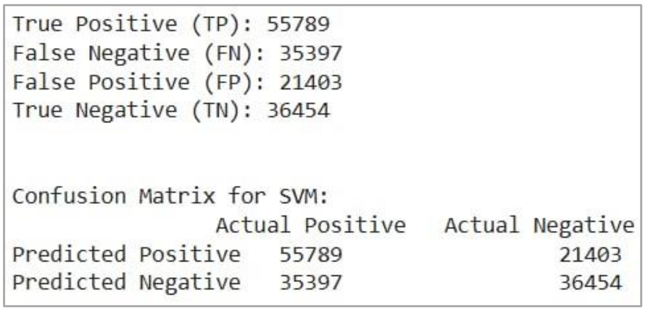
Table 4 illustrates the superiority of the SVM, DT, and MLP models across all evaluation metrics. To calculate the advantage, we subtract the corresponding scores of SVM and MLP from DT's scores (e.g., Accuracy Advantage over SVM = 98.83% - 61.89% = 36.94%). The table highlights the dominant performance of DT in ransomware detection, showing significant improvements over both models.

The advantage ranges from a minimum of 27.14% (precision over SVM) to a maximum of 53.50% (recall over MLP).

This advantage of DT likely stems from its ability to capture non-linear relationships within the data, which is a common characteristic of ransomware. Unlike linear models like SVM, DT's structure allows it to model these complexities, potentially explaining the significant performance difference observed in our results.

The confusion matrix for each experiment is provided (Figure 3, Figure 4, and Figure 5), allowing readers to calculate alternative performance metrics as needed

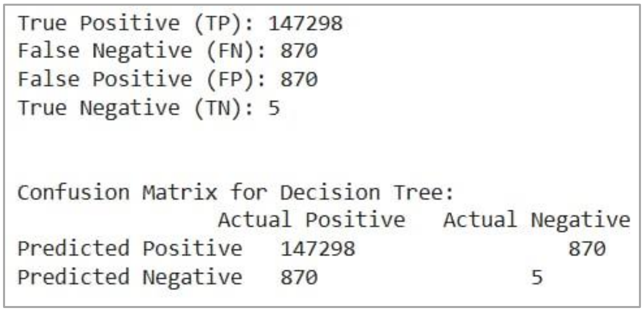
Figure 3. Confusion matrix of SVM



```
True Positive (TP): 55789
False Negative (FN): 35397
False Positive (FP): 21403
True Negative (TN): 36454


Confusion Matrix for SVM:
                Actual Positive   Actual Negative
Predicted Positive   55789                21403
Predicted Negative   35397                36454
```

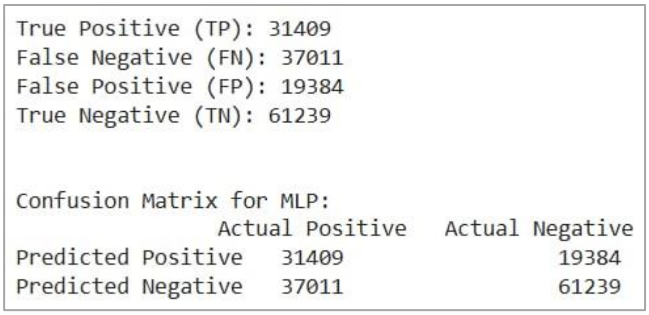
Figure 4. Confusion matrix of DT



```
True Positive (TP): 147298
False Negative (FN): 870
False Positive (FP): 870
True Negative (TN): 5


Confusion Matrix for Decision Tree:
                Actual Positive   Actual Negative
Predicted Positive   147298                870
Predicted Negative   870                5
```

Figure 5. Confusion matrix of MLP



```
True Positive (TP): 31409
False Negative (FN): 37011
False Positive (FP): 19384
True Negative (TN): 61239


Confusion Matrix for MLP:
                Actual Positive   Actual Negative
Predicted Positive   31409                 19384
Predicted Negative   37011                 61239
```

Conclusion

In conclusion, this research investigated the performance of three machine learning models for ransomware detection: SVM, DT, and multilayer perceptron (MLP). The study evaluated their effectiveness in distinguishing between ransomware and benign files using four key metrics: accuracy, precision, recall, and F-measure. The results overwhelmingly favored the DT model, which achieved exceptional scores across all metrics (accuracy = 98.83%, precision = 99.41%, recall = 99.41%, F-measure = 99.41%). This indicates DT's remarkable ability to accurately classify files and minimize false positives or negatives.

Compared to the other models, DT demonstrated a significant advantage. It outperformed SVM by margins ranging from 27.14% in precision to 38.23% in recall, highlighting DT's superior capability in correctly identifying ransomware threats. Similarly, DT maintained a substantial lead over MLP, with advantages exceeding 36% in accuracy and exceeding 53% in recall. These findings suggest that DT offers a more robust and reliable solution for ransomware detection compared to SVM and MLP.

This research contributes significantly to the development of effective ransomware detection methods. While previous studies explored the potential of ensemble models, which combine multiple models, this research demonstrates the outstanding performance achievable with a single DT model. This finding suggests a potentially simpler and more efficient solution for practical implementation. Based on the employed evaluation metrics, DT emerged as the clear winner for ransomware detection. Its exceptional performance across all metrics warrants further investigation into its potential as a robust and reliable tool for safeguarding systems against ransomware attacks.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D. G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., . . . Zheng, X. (2016). Tensor flow: A system for large-scale machine learning. *12th USENIX Symposium on Operating Systems Design and Implementation*, *16*, 265–283. https://doi.org/10.5555/3026877.3026899

Abushark, Y. B., Khan, A. I., Alsolami, F., Almalawi, A., Alam, M. M., Agrawal, A., Kumar, R., & Khan, R. A. (2022). Cyber security analysis and evaluation for intrusion detection systems. *Computers, Materials & Continua/Computers, Materials & Continua (Print)*, *72*(1), 1765–1783. https://doi.org/10.32604/cmc.2022.025604

Alam, S., Sonbhadra, S. K., Agarwal, S., & Nagabhushan, P. (2020). One-class support vector classifiers: A survey. *Knowledge-Based Systems*, *196*, 105754. https://doi.org/10.1016/j.knosys.2020.105754

Alharbi, A., Seh, A. H., Alosaimi, W., Alyami, H., Agrawal, A., Kumar, R., & Khan, R. A. (2021). Analyzing the impact of cyber security related attributes for intrusion detection systems. *Sustainability*, *13*(22), 12337. https://doi.org/10.3390/su132212337

Alshammari, A., Darem Laith A, Sheatah, H., & Effghi, R. (2024). Ransomware Early Detection Techniques. *Engineering, Technology & Applied Science Research*, *14*(3), 14497–14503.

https://etasr.com/index.php/ETASR/article/download/6915/3693

Azugo, P., Venter, H., & Nkongolo, M. W. (2024). Ransomware Detection and Classification Using Random Forest: A Case Study with the UGRansome2024 Dataset. *arXiv preprint arXiv:2404.12855.*

Blockeel, H., Devos, L., Frénay, B., Nanfack, G., & Nijssen, S. (2023). Decision trees: from efficient prediction to responsible AI. *Frontiers in Artificial Intelligence*, *6*. https://doi.org/10.3389/frai.2023.1124553

Celdrán, A. H., Sánchez, P. M. S., Castillo, M. A., Bovet, G., Pérez, G. M., & Stiller, B. (2022). Intelligent and behavioral-based detection of malware in IoT spectrum sensors. *International Journal of Information Security*, *22*(3), 541–561. https://doi.org/10.1007/s10207-022-00602-w

Fernando, D. W., Komninos, N., & Chen, T. (2020). A study on the evolution of ransomware detection using machine learning and deep learning techniques. *IoT*, *1*(2), 551–604. https://doi.org/10.3390/iot1020030

Griffiths, C. (2024, June 26). The Latest Ransomware Statistics (updated June 2024) AAG IT Support. *AAG IT Services*. Retrieved June 27, 2024, from https://aag-it.com/the-latest-ransomware-statistics/#:~:text=During%202021%2C%20at%20least%2015.45,unique%20user%20computers%20in%202021.

Gu, J., & Lu, S. (2021). An effective intrusion detection approach using SVM with naïve Bayes feature embedding. *Computers & Security*, *103*, 102158. https://doi.org/10.1016/j.cose.2020.102158

Han, J., Kamber, M., & Pei, J. (2012). *Data Mining. Concepts and Techniques* (3rd ed.). Morgan Kaufmann Publishers.

Harris, C. R., Millman, K. J., Van Der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E. S., Taylor, J., Berg, S., Smith, N. J., Kern, R., Picus, M., Hoyer, S., Van Kerkwijk, M. H., Brett, M., Haldane, A., Del Río, J. F., Wiebe, M. W., Peterson, P., . . . Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, *585*(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

Hunter, J. D. (2007). MatPlotLib: a 2D Graphics environment. *Computing in Science & Engineering*, *9*(3), 90–95. https://doi.org/10.1109/mcse.2007.55

Kaspersky. (2024, May 23). State of ransomware in 2024. *Securelist*. https://securelist.com/state-of-ransomware-2023/112590/

Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. *Cybersecurity*, *2*(1). https://doi.org/10.1186/s42400-019-0038-7

Kayode-Ajala, O. (2021). Anomaly Detection in Network Intrusion Detection Systems Using Machine Learning and Dimensionality Reduction. *Sage Science Review of Applied Machine Learning, 4*(1), 12-26.

Kok, S., Abdullah, A., Jhanjhi, N., & Supramaniam, M. (2019). Ransomware, threat and detection techniques: A review. *Int. J. Comput. Sci. Netw. Secur, 19*(2), 136.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Li, Y., Author_Id, N., Xu, W., Li, W., Li, A., & Liu, Z. (2021). Research on hybrid intrusion detection method based on the ADASYN and ID3 algorithms. *Mathematical Biosciences and Engineering*, *19*(2), 2030–2042. https://doi.org/10.3934/mbe.2022095

Liao, H., Lin, C. R., Lin, Y., & Tung, K. (2013). Intrusion detection system: A comprehensive review. *Journal of Network and Computer Applications*, *36*(1), 16–24. https://doi.org/10.1016/j.jnca.2012.09.004

Liu, H., & Lang, B. (2019). Machine Learning and Deep Learning Methods for Intrusion Detection Systems: a survey. *Applied Sciences*, *9*(20), 4396. https://doi.org/10.3390/app9204396

McKinney, W. (2010). Data structures for statistical computing in Python. *Proceedings of the Python in Science Conferences*. https://doi.org/10.25080/majora-92bf1922-00a

Nkongolo, M. (2024). RFSA: A Ransomware feature selection algorithm for multivariate analysis of malware behavior in cryptocurrency. *International Journal of Computing and Digital System/International Journal of Computing and Digital Systems*, *15*(1), 901–935. https://doi.org/10.12785/ijcds/150165

Nkongolo, M., Van Deventer, J. P., & Kasongo, S. M. (2021). UGRAnSome1819: A Novel Dataset for Anomaly Detection and Zero-Day Threats. *Information*, *12*(10), 405. https://doi.org/10.3390/info12100405

Nkongolo, M., Van Deventer, J. P., Kasongo, S. M., Zahra, S. R., & Kipongo, J. (2022). A cloud based optimization method for Zero-Day threats detection using genetic algorithm and ensemble learning. *Electronics*, *11*(11), 1749. https://doi.org/10.3390/electronics11111749

Nosratabadi, S., Ardabili, S., Lakner, Z., Mako, C., & Mosavi, A. (2021). Prediction of food production using machine learning algorithms of multilayer perceptron and ANFIS. *Agriculture*, *11*(5), 408. https://doi.org/10.3390/agriculture11050408

Oz, H., Aris, A., Levi, A., & Uluagac, A. S. (2022). A survey on Ransomware: Evolution, taxonomy, and defense solutions. *ACM Computing Surveys*, *54*(11s), 1–37. https://doi.org/10.1145/3514229

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine Learn-ing in Python, Journal of Machine Learning Re-search, 12.

Portokalidis, G., & Bos, H. (2007). SweetBait: Zero-hour worm detection and containment using low- and high-interaction honeypots. *Computer Networks*, *51*(5), 1256–1274.https://doi.org/10.1016/j.comnet.2006.09.005

Quinlan, J. (1990). Decision trees and decision-making. *IEEE Transactions on Systems, Man, and Cybernetics*, *20*(2), 339–346. https://doi.org/10.1109/21.52545

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, *1*(1), 81–106. https://doi.org/10.1007/bf00116251

Quinlan, J. R. (1992). *C4.5: Programs for Machine learning*. https://cds.cern.ch/record/2031749

UGRansome Dataset. (2023, December 11). Kaggle.