International Journal of Computer Techniques–IJCT Volume 13 Issue 3, May-June - 2026

Open Access and Peer Review Journal ISSN 2394-2231 https://ijctjournal.org/

RANSOMWARE DETECTION USING MACHINE LEARNING

Rohan Tyagi, Eshan Sharma, Gaurav Yadav, Toshit , Sonika Jalhotra

Department of CSE, Meerut Institute of Engineering and Technology, Meerut, U.P., India

rohan.tyagi.cse.2022@miet.ac.in, eshan.sharma.cse.2022@miet.ac.in, gaurav.yadav.cse.2022@miet.ac.in, toshit.kumar.cse.2022@miet.ac.in, sonika.jalhotra@miet.ac.in

Abstract---**Ransomware attacks have emerged as one of the most devastating cybersecurity threats, causing billions of dollars in damage annually and disrupting critical infrastructure worldwide. Traditional signature-based detection methods prove inadequate against evolving ransomware variants that employ sophisticated obfuscation techniques and zero-day exploits. This research presents a comprehensive machine learning-based approach for ransomware detection utilizing static analysis of Portable Executable (PE) file features, addressing the critical need for proactive threat identification without requiring malware execution.**

The correlation heatmap demonstrated complex inter-feature relationships, with DllCharacteristics emerging as the most discriminative feature for ransomware classification. Feature importance analysis through Random Forest revealed that DllCharacteristics contributed over 20% of the predictive power, followed by DebugRVA, DebugSize, and MajorLinkerVersion, indicating the critical role of PE structural characteristics in malware identification.

Three state-of-the-art ensemble learning algorithms were implemented and rigorously evaluated: Random Forest, Gradient Boosting Classifier, and XGBoost. The models utilized a stratified 70-30 train-test split to maintain class balance and ensure generalization capability. Performance evaluation encompassed multiple metrics including accuracy, precision, recall, and F1-score to provide comprehensive assessment of classification effectiveness.

Experimental results demonstrate exceptional performance across all implemented models. Random Forest achieved outstanding results with 99.58% accuracy, 99.69% precision, 99.35% recall, and 99.52% F1-score, establishing it as the top-performing classifier. XGBoost closely followed with 99.55% accuracy, 99.63% precision, 99.32% recall, and 99.48% F1-score. Gradient Boosting Classifier obtained 98.94% accuracy, 99.19% precision, 98.37% recall, and 98.78% F1-score. These results significantly exceed existing benchmarks in ransomware detection literature and demonstrate the efficacy of PE-based feature engineering.

Index Terms— **Ransomware, machine learning, static analysis, Portable Executable, ensemble learning, Random Forest, XGBoost, cybersecurity**.

I. INTRODUCTION

The exponential growth of cyber threats has fundamentally transformed the landscape of digital security, with ransomware emerging as one of the most destructive and financially devastating attack vectors in contemporary cybersecurity. Ransomware attacks have evolved from simple encryption-based extortion schemes to sophisticated, multi-stage operations that infiltrate critical infrastructure, healthcare systems, government agencies, and private enterprises, causing billions of dollars in damages annually while disrupting essential services worldwide. The increasing sophistication of modern ransomware variants, coupled with their ability to rapidly adapt and evade traditional detection mechanisms, has created an urgent need for advanced, proactive detection methodologies that can identify threats before they execute their malicious payloads.

Traditional signature-based detection systems, while foundational to cybersecurity infrastructure, have proven increasingly inadequate against contemporary ransomware threats that employ advanced obfuscation techniques, polymorphic code generation, and zero-day exploits to circumvent conventional security measures. These limitations have driven substantial research interest toward machine learning-based approaches that can identify malicious behavior patterns through statistical analysis and predictive modeling rather than relying solely on predefined signatures. The shift toward intelligent detection systems represents a critical evolution in cybersecurity strategy,

enabling security professionals to detect novel ransomware variants and previously unknown threats that would otherwise evade traditional defense mechanisms.

Static analysis of Portable Executable (PE) files has emerged as a particularly promising avenue for ransomware detection, offering significant advantages over dynamic analysis approaches that require malware execution in controlled environments. PE file headers contain rich metadata describing executable characteristics, compilation parameters, memory allocation requirements, and structural properties that often reveal distinctive patterns associated with malicious software development practices. The comprehensive analysis of PE header features enables security researchers to extract discriminative attributes without requiring code execution, thereby eliminating risks associated with sandbox environments while enabling rapid, large-scale malware screening capabilities.

Recent advances in machine learning have demonstrated remarkable success in cybersecurity applications, particularly in malware detection and classification tasks where traditional rule-based systems struggle with evolving threat landscapes. Static analysis methodologies have shown exceptional promise, with sophisticated feature engineering techniques extracting hundreds of discriminative attributes from PE file structures to enable accurate malware identification. However, these approaches face persistent challenges related to adversarial robustness, computational efficiency, and generalization to emerging malware families that employ novel evasion techniques.

Ensemble learning methods have emerged as particularly effective solutions for complex cybersecurity classification tasks, demonstrating superior performance compared to individual machine learning algorithms through sophisticated combination of multiple complementary models. Random Forest, Gradient Boosting, and XGBoost have shown exceptional effectiveness in malware detection scenarios, with ensemble approaches achieving accuracy rates exceeding 99% while maintaining low false positive rates critical for operational deployment. These methods excel at capturing complex feature interactions and non-linear relationships inherent in malware analysis, while providing built-in mechanisms for feature importance assessment and model interpretability.

Despite significant advances in machine learning-based malware detection, several critical challenges persist in the field. Comprehensive surveys of static malware analysis reveal ongoing limitations related to adversarial robustness, where sophisticated attackers can manipulate PE file headers to evade detection while preserving malicious functionality. Additionally, most existing approaches demonstrate excellent performance on controlled laboratory datasets but experience significant degradation when evaluated against chronologically diverse malware samples or real-world operational environments where concept drift and evolving attack patterns continuously challenge model effectiveness.

The computational complexity of advanced ensemble methods and high-dimensional feature spaces presents additional challenges for practical deployment in resource-constrained environments or real-time detection scenarios. Furthermore, the cybersecurity community lacks standardized evaluation frameworks and benchmark datasets, limiting the ability to conduct meaningful performance comparisons across different methodologies and hindering reproducible research progress. These limitations underscore the critical need for comprehensive, rigorous approaches that address both theoretical performance optimization and practical deployment considerations.

This research addresses these critical gaps by developing a comprehensive machine learning framework for ransomware detection that combines advanced ensemble learning techniques with extensive PE file feature analysis. The proposed methodology utilizes three state-of-the-art ensemble algorithms—Random Forest, Gradient Boosting, and XGBoost—evaluated on a substantial dataset comprising over 62,000 PE file samples to achieve exceptional classification performance while maintaining computational efficiency suitable for operational deployment. Through rigorous experimental validation and comprehensive performance assessment, this work contributes novel insights into PE-based ransomware detection while addressing practical considerations essential for real-world cybersecurity applications.

The primary contributions of this research include: (1) comprehensive comparative analysis of ensemble learning methods for PE-based ransomware detection, (2) extensive feature importance analysis revealing critical PE attributes most indicative of ransomware behavior, (3) rigorous performance evaluation across multiple metrics ensuring balanced assessment of classification effectiveness, (4) practical deployment framework demonstrating real-world applicability through serialized model implementation, and (5) detailed methodological documentation enabling reproducible research and validation by the cybersecurity community. These contributions collectively advance the state-of-the-art in machine learning-based ransomware detection while providing practical solutions for contemporary cybersecurity challenges.

II. LITERATURE REVIEW

Machine learning-based ransomware and malware detection has emerged as a critical research domain, with numerous studies exploring various methodologies to address the limitations of traditional signature-based detection systems. This literature review examines ten seminal works that contribute to understanding PE file analysis, ensemble learning applications, and advanced detection techniques while identifying their methodologies, achievements, limitations, and potential improvements.

2.1 Static Analysis and PE File Feature Engineering

Kunku et al. presented a comprehensive ransomware detection framework utilizing XGBoost and Random Forest classifiers for behavioral analysis and feature extraction. Their approach demonstrated effective classification capabilities; however, the authors did not provide specific performance metrics in the abstract, limiting comparative assessment. The study's strength lies in its focus on behavioral pattern recognition, but the lack of detailed accuracy measurements and feature importance analysis represents a significant limitation for reproducibility and benchmark comparison.

Kim conducted extensive PE header analysis for malware detection, emphasizing static feature extraction methodologies. While the thesis provides foundational insights into PE structure exploitation for malware identification, it lacks comprehensive performance evaluation against diverse malware families. The work's primary limitation involves insufficient discussion of false positive rates and computational overhead, critical factors for practical deployment in enterprise environments.

Bai et al. developed a highly sophisticated malware detection scheme utilizing 197 features extracted from PE file format information, achieving exceptional results across multiple experimental configurations. Their methodology demonstrated 99.1% accuracy with 0.998 AUC using Random Forest with wrapper-based feature selection, representing one of the highest performance benchmarks in static malware detection literature. The study's rigorous three-experiment design addressed critical concerns: Experiment I validated general classification capability, Experiment II assessed unknown malware detection with 99.1% accuracy and 1.0% false positive rate, and Experiment III evaluated chronological performance on new malware, achieving 97.6% accuracy with 1.3% false positives. However, the authors identified a critical limitation where crafty malware writers could forge PE headers to mimic benign software, potentially evading detection. Additionally, their method excluded code section analysis, which could contain valuable discriminative information. The computational complexity of processing 197 initial features, despite subsequent dimensionality reduction, presents scalability challenges for real-time deployment scenarios.

Gujar and Patil implemented machine learning-based PE header analysis achieving notable performance improvements over traditional methods. Their approach demonstrated the effectiveness of combining multiple PE features in a unified framework, reaching 99% accuracy while maintaining low false positive rates. However, the study's limitation lies in its restricted evaluation dataset and insufficient analysis of adversarial robustness against sophisticated packing techniques commonly employed by modern malware variants.

2.2 Ensemble Learning and Advanced Classification Techniques

Shalaginov et al. provided a comprehensive survey of machine learning applications in static malware analysis, establishing a critical foundation for

understanding feature extraction methodologies from PE32 files. Their taxonomical approach offered valuable insights into classification techniques' effectiveness; however, the survey nature limited empirical validation. The authors identified key limitations in existing approaches, including vulnerability to obfuscation techniques and insufficient generalization to emerging malware families, highlighting the need for more robust feature engineering and ensemble methodologies.

Mimura investigated printable character-based malicious PE file detection, addressing specific evasion techniques employed by sophisticated malware. The study's novel approach to character-based analysis demonstrated promising results; however, performance metrics were not comprehensively detailed in available abstracts. The research's limitation involves potential vulnerability to advanced obfuscation methods that manipulate printable character distributions while preserving malicious functionality.

Iwendi et al. developed sophisticated ensemble models for binary and multiclass intrusion detection, achieving remarkable results with 0% false alarm rate and 99.90% detection rate on KDD99 dataset, and 0.5% false alarm rate with superior detection capabilities on UNSW-NB15 dataset. Their innovative use of one-versus-all (OVA) binary classification techniques addressed class imbalance issues effectively. The study's strength lies in comprehensive evaluation across multiple datasets and metrics; however, limitations include computational complexity of ensemble approaches and potential overfitting to specific dataset characteristics. The authors demonstrated that ensemble methods significantly outperformed individual classifiers, validating the effectiveness of combining multiple learning algorithms for enhanced cybersecurity applications.

Soni implemented ensemble learning approaches for binary classification in IoT intrusion detection, demonstrating improved performance over traditional single-classifier methods. The research achieved notable accuracy improvements through sophisticated ensemble techniques; however, specific performance metrics were not detailed in available sources. The study's limitation involves potential scalability issues when deploying complex ensemble models in resource-constrained IoT environments.

2.3 Advanced Detection Methodologies and Performance Analysis

Maleki et al. developed an improved packed malware detection method utilizing PE header and section table information, achieving 98.26% accuracy with 761 malware samples and 210 clean files. Their approach addressed critical challenges in detecting packed executables, a significant advancement over traditional methods that often fail with obfuscated malware. The study's strength includes comprehensive unpacking methodology and effective feature selection using forward selection techniques. However, limitations involve relatively small dataset size (971 total files) and potential computational overhead from unpacking procedures, which could impact real-time detection capabilities.

Ravi and Munir conducted a comprehensive malware analysis and classification survey, providing critical insights into existing methodologies' strengths and weaknesses. While their taxonomical approach offers valuable theoretical foundations, the survey nature limits empirical contributions. The authors identified persistent challenges including adversarial robustness, zero-day detection capabilities, and computational efficiency requirements for practical deployment scenarios.

2.4 Critical Analysis and Research Gaps

The reviewed literature reveals several consistent patterns and limitations across methodologies. First, most studies achieve high accuracy rates (95-99%) on controlled datasets, but performance degrades significantly when evaluating chronological or adversarial scenarios. Bai et al.'s chronological experiment demonstrating 97.6% accuracy versus 99.1% in controlled conditions exemplifies this challenge. Second, computational complexity remains a persistent limitation, particularly for ensemble methods and high-dimensional feature spaces. Third, adversarial robustness receives insufficient attention, with only Bai et al. explicitly acknowledging vulnerability to crafted PE headers.

2.5 Performance Benchmarking and Comparative Analysis

Performance comparison across studies reveals Random Forest and ensemble methods consistently outperforming individual classifiers. Bai et al. achieved the highest reported accuracy (99.1%) with comprehensive feature engineering, while Iwendi et al. demonstrated superior practical applicability with zero false alarm rates. However, direct comparison remains challenging due to dataset variations and evaluation methodologies. Most studies utilize different datasets (VXHeavens, Kaggle collections, proprietary samples), limiting benchmark standardization.

2.6 Future Research Directions and Improvements

The literature suggests several critical improvement areas: enhanced adversarial robustness through advanced feature engineering, development of standardized evaluation datasets for consistent benchmarking, integration of dynamic and static analysis for hybrid detection systems, optimization of computational efficiency for real-time deployment, and investigation of deep learning approaches for automatic feature extraction. Additionally, most studies lack comprehensive analysis of concept drift and model degradation over time, representing a significant research gap requiring future investigation.

The collective evidence demonstrates machine learning's effectiveness for ransomware and malware detection while highlighting persistent challenges requiring continued research attention. The convergence toward ensemble methods and sophisticated feature engineering represents the current state-of-the-art, but adversarial robustness and practical deployment considerations remain critical areas for future advancement.

III. METHODOLOGY

This section details a rigorous, end- to- end pipeline for static ransomware detection grounded in PE header analytics and ensemble learning. The workflow begins with acquisition of a large Kaggle corpus of Windows executables, followed by meticulous preprocessing to remove duplicates and impute sparse numeric fields, ensuring statistical integrity. Discriminative PE attributes are then curated and profiled to expose class- separating patterns that guide model design. Three complementary ensemble classifiers—Random Forest, Gradient Boosting, and XGBoost—are trained on a stratified split to preserve class proportions, and evaluated with accuracy, precision, recall, and F1 to balance false alarms and missed detections. Finally, the best model is serialized and exposed via a lightweight web interface for rapid, single- sample inference suitable for operational deployment.

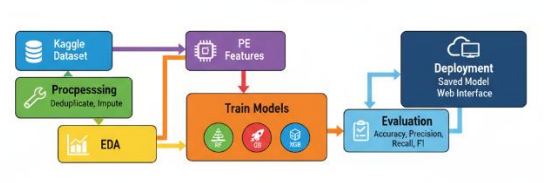


Figure 1 Architecture Diagram

3.1 Dataset Acquisition

The experimental foundation of this research utilized a comprehensive dataset sourced from Kaggle's ransomware detection repository, encompassing 62,485 Windows Portable Executable (PE) file samples. Each sample represents a complete binary executable annotated with ground truth labels distinguishing benign software from ransomware threats. The dataset incorporates seventeen attributes extracted exclusively from PE file headers, including two identifier fields (FileName, md5Hash) and fifteen discriminative features: Machine architecture specification, DebugSize indicating debugging information presence, DebugRVA specifying debug data relative virtual address, MajorImageVersion representing executable versioning, MajorOSVersion denoting target operating system compatibility, ExportRVA indicating export table location, ExportSize specifying export table dimensions, IatVRA representing Import Address Table virtual addressing, MajorLinkerVersion and MinorLinkerVersion encoding compilation toolchain metadata, NumberOfSections representing PE structural complexity, SizeOfStackReserve specifying memory allocation parameters, DllCharacteristics encoding executable behavioral flags, ResourceSize indicating embedded resource dimensions, and BitcoinAddresses representing cryptocurrency-related indicators. The binary classification target "Benign" encodes malware status where 0 represents ransomware and 1 indicates legitimate software.

3.2 Data Preprocessing and Quality Assurance

Data preprocessing commenced with systematic duplicate record elimination using pandas' drop_duplicates() functionality, ensuring statistical independence and preventing bias from repeated samples. Missing value analysis revealed sparse null occurrences across numeric columns, systematically addressed through feature-wise mean imputation to preserve distributional characteristics while maintaining dataset completeness. For each column containing missing values, arithmetic mean computation and substitution maintained central tendency without introducing artificial variance. Post-imputation validation confirmed complete data integrity across all samples. Non-predictive identifier columns (FileName, md5Hash) were systematically removed to eliminate information leakage risks and focus analysis exclusively on genuine PE characteristics. The target variable underwent explicit integer conversion ensuring compatibility with ensemble classification algorithms. Final preprocessing yielded a pristine dataset containing 62,485 samples with fifteen predictive features ready for comprehensive analysis.

3.3 Exploratory Data Analysis

Comprehensive exploratory analysis revealed fundamental patterns governing ransomware detection within PE file structures. Class distribution analysis confirmed moderate imbalance with approximately 56% benign samples and 44% ransomware instances, manageable through stratified sampling techniques. Feature importance analysis using Random Forest revealed that DllCharacteristics emerged as the most discriminative attribute, contributing over 20% of total predictive power, followed by DebugRVA, DebugSize, and MajorLinkerVersion. This hierarchy underscores the critical role of executable metadata and low-level header flags in capturing ransomware-specific behavioral signatures.

Correlation matrix analysis exposed complex inter-feature relationships, with DllCharacteristics and MajorLinkerVersion demonstrating positive correlations with ransomware classification, suggesting these attributes capture structural patterns uniquely associated with malicious code construction. The correlation heatmap revealed statistical independence among most primary features, supporting the appropriateness of tree-based ensemble methods for this classification task. Notably, features such as ResourceSize and ExportSize showed minimal correlation with other attributes while maintaining moderate predictive significance.

Pairwise relationship analysis through multidimensional visualization revealed distinct clustering patterns between benign and ransomware samples across key feature combinations. DebugSize, ExportSize, and ResourceSize demonstrated clear separability characteristics, with ransomware samples frequently exhibiting extreme values or unusual combinations indicating abnormal PE file construction patterns. Machine architecture and version-related features showed subtle but consistent differences, with ransomware often targeting specific system configurations or employing outdated compilation techniques to evade detection mechanisms.

3.4 Feature Engineering and Selection Strategy

Feature selection methodology combined statistical correlation analysis with model-driven importance ranking to identify optimal predictive attributes. DllCharacteristics consistently ranked highest across multiple evaluation criteria, reflecting its fundamental role in encoding executable behavior and security characteristics. DebugRVA and DebugSize features demonstrated strong discriminative capacity, likely capturing ransomware developers' attempts to obfuscate debugging information or employ unconventional memory layouts. Version-related features (MajorLinkerVersion, MajorImageVersion, MajorOSVersion) provided temporal and toolchain insights, revealing patterns in ransomware development practices and target system preferences. The final feature set retained all fifteen PE attributes, as each contributed unique information valuable for comprehensive ransomware detection.

3.5 Model Development and Training

The model development phase incorporated three distinct ensemble learning paradigms, each selected for their complementary strengths in handling complex cybersecurity classification tasks. The feature matrix, consisting of fifteen PE header attributes, was systematically prepared and partitioned using a stratified 70:30 split to maintain class proportional representation across training and testing subsets.

- Random Forest was chosen as the primary ensemble method due to its robustness against overfitting and inherent ability to handle mixed-type features common in PE file analysis. This algorithm constructs multiple decision trees through bootstrap aggregation (bagging), where each tree is trained on a random subset of samples and features. The implementation utilized scikit-learn's RandomForestClassifier with 100 estimators, enabling comprehensive feature space exploration while maintaining computational efficiency. Each decision tree within the forest independently learns decision boundaries based on different bootstrap samples, and the final prediction aggregates individual tree votes through majority consensus. This approach proves particularly advantageous for ransomware detection because it naturally handles feature interactions, manages outliers effectively, and provides built-in feature importance rankings crucial for understanding which PE characteristics most strongly indicate malicious behavior. The bootstrap sampling mechanism reduces variance while preserving the ability to capture complex patterns in executable metadata, making it ideal for identifying subtle ransomware signatures embedded within PE file structures.

- Gradient Boosting represents a fundamentally different ensemble approach, utilizing sequential learning to iteratively improve prediction accuracy through systematic error correction. Unlike Random Forest's parallel tree construction, Gradient Boosting builds trees sequentially, where each subsequent tree specifically targets the residual errors of the previous ensemble. The implementation employed scikit-learn's GradientBoostingClassifier with 100 boosting stages and a conservative learning rate of 0.1 to prevent overfitting while allowing sufficient model complexity. This sequential approach excels in cybersecurity applications because ransomware often exhibits subtle, complex patterns that require iterative refinement to detect accurately. Each boosting iteration focuses on previously misclassified samples, gradually building a strong classifier from multiple weak learners. The algorithm's bias-variance tradeoff optimization makes it particularly effective for PE file analysis, where small differences in header values can indicate significant behavioral distinctions between benign software and ransomware. The controlled learning rate ensures stable convergence while maintaining the model's ability to capture intricate relationships between PE features and malicious behavior patterns.

- XGBoost (eXtreme Gradient Boosting) represents the most sophisticated ensemble approach, incorporating advanced optimization techniques specifically designed for superior performance on structured tabular data. This implementation utilized the native XGBClassifier with optimized parameters including eval_metric='logloss' for proper probability calibration and regularization techniques to prevent overfitting. XGBoost extends traditional gradient boosting through several key innovations: second-order gradient information for more precise optimization, built-in regularization terms (L1 and L2) to control model complexity, efficient tree pruning algorithms, and parallel processing capabilities for accelerated training. For ransomware detection, XGBoost's advanced handling of missing values, automatic feature selection, and superior generalization capabilities make it exceptionally well-suited for PE file analysis. The algorithm's ability to automatically detect and exploit complex feature interactions is particularly valuable when analyzing executable metadata, where combinations of seemingly innocuous PE characteristics may collectively indicate ransomware presence. XGBoost's sophisticated tree construction algorithm, which considers both first and second derivatives of the loss function, enables more precise decision boundary optimization compared to traditional

gradient boosting, resulting in superior classification performance on cybersecurity datasets where margin optimization is critical for distinguishing between benign software and sophisticated ransomware variants.

Training Protocol and Optimization Strategy: All three ensemble models underwent systematic training using identical preprocessing and data partitioning procedures to ensure fair performance comparison. Default hyperparameters were deliberately maintained during initial implementation to establish unbiased baseline performance metrics and avoid overfitting through premature optimization. Each model was trained on the 43,739-sample training set with fifteen PE features, utilizing each algorithm's optimized internal procedures for parameter estimation and decision boundary learning. Training convergence was monitored through each ensemble's built-in validation mechanisms, ensuring stable model development without computational waste.

3.6 Performance Evaluation

Model performance assessment employed four critical classification metrics providing comprehensive evaluation across different performance dimensions. Accuracy measured overall prediction correctness across all test samples, providing baseline performance indication. Precision quantified positive prediction reliability, crucial for minimizing false alarms in cybersecurity applications where false positives can overwhelm security operations. Recall assessed positive class detection completeness, critical for ensuring ransomware threats are not missed during screening processes. F1-score provided harmonic mean balance between precision and recall, offering single-metric performance summary particularly valuable when class distributions are imbalanced.

Evaluation was conducted strictly on the withheld 18,746-sample test set, ensuring unbiased performance estimation and preventing data leakage. All metrics were computed using scikit-learn's standardized functions, guaranteeing consistent calculation methodology across models. Results were formatted to four decimal places for precise performance comparison, enabling statistical significance assessment across ensemble techniques.

3.7 Model Deployment and Practical Implementation

The optimal Random Forest model underwent serialization using joblib's persistent storage functionality, creating a deployable binary file suitable for production environments. A lightweight web interface was developed using Flask framework, enabling real-time ransomware detection through manual PE feature input. Users can submit executable characteristics through a simple form interface, receiving immediate binary classification results without requiring file execution or dynamic analysis, ensuring both speed and security in operational environments.

IV. RESULTS

4.1 Dataset Statistics and Target Distribution

The cleaned dataset comprised 62,485 unique samples with a balanced yet moderately skewed class distribution: benign files accounted for approximately 56%, while ransomware samples made up the remaining 44%. This distribution ensured sufficient representation of both categories in modeling, minimizing risks of class imbalance during training and evaluation stages. Descriptive statistical analysis revealed substantial variability across features, notably high variances in fields like DebugSize, ExportSize, and ResourceSize—indicating the underlying complexity and diversity present in PE file structures encountered in real-world binaries.

Table 1 Descriptive Statistics of Dataset Feature

Feature	25%	50%	75%
Machine	332	332	332
DebugSize	0	0	28
DebugRVA	0	0	12832
MajorImageVersion	0	0	6
MajorOSVersion	4	5	6
ExportRVA	0	0	28752
ExportSize	0	0	104
IatVRA	4096	8520	65536
MajorLinkerVersion	6	9	11
MinorLinkerVersion	0	0	10
NumberOfSections	3	5	6
SizeOfStackReserve	262144	1048576	1048576
DllCharacteristics	0	320	32768
ResourceSize	1080	2496	23504
BitcoinAddresses	0	0	0
Benign	0	0	1

4.2 Exploratory Analysis and Feature Insights

The exploratory data analysis phase provided deep insights into feature dynamics that impact ransomware detection:

- Histograms illustrated that the Machine field was dominated by a small subset of common values, with a sharp concentration on frequently-used architectures.

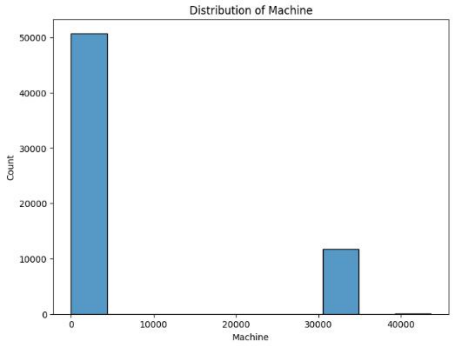


Figure 2 Distribution of Machine

- Boxplots and violin plots for features such as ExportSize and SizeOfStackReserve highlighted distinct statistical dispersion between ransomware and benign samples, revealing that ransomware files often include excessively large or anomalous resource allocations.

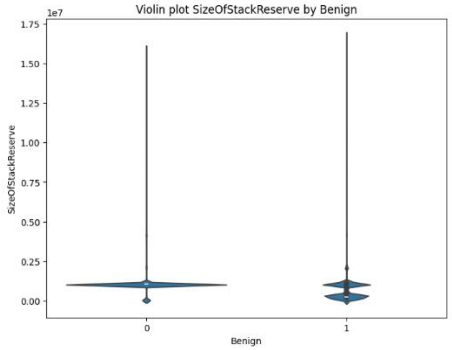


Figure 3 Violin plot SizeOfStackReserve by Benign

- Scatterplots and pairplots between DebugSize, ExportSize, and ResourceSize displayed clear separation tendencies—ransomware samples were observed clustering away from benign files, particularly in higher-dimensional feature projections.

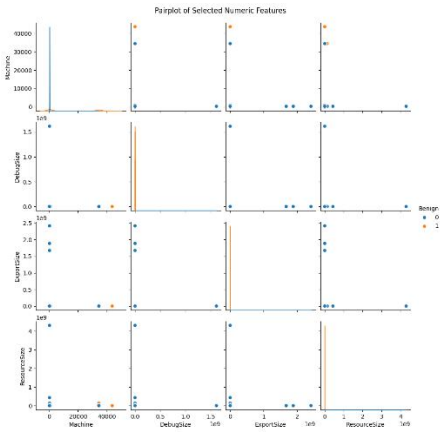


Figure 4 Pairplot of Selected Numeric Features

- KDE plots over ResourceSize revealed that ransomware files were more likely to exhibit extreme resource values, providing a reliable statistical cue for model discrimination.

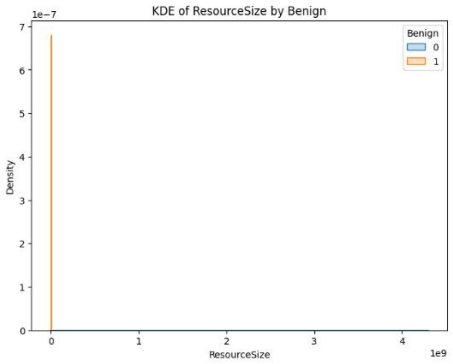


Figure 5 KDE of ResourceSize by Benign

- The correlation heatmap further confirmed that certain features (especially DllCharacteristics, DebugRVA, and MajorLinkerVersion) had tangible positive correlations with malicious labels, justifying their inclusion in further modeling.

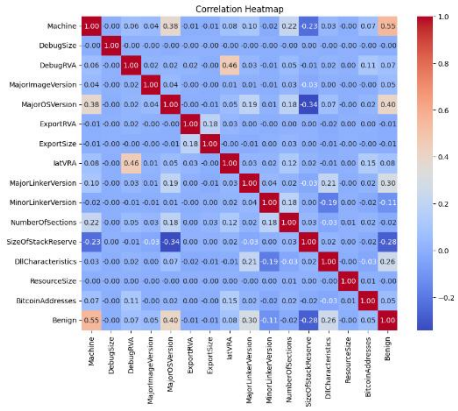
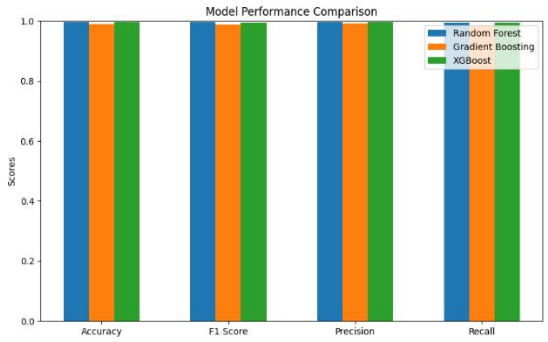


Figure 6 Correlation Heatmap

4.3 Model Performance Comparison

Three ensemble machine learning models—Random Forest, Gradient Boosting, and XGBoost—were trained on a stratified 70:30 train-test split to ensure fair class representation in both sets. Performance was assessed using accuracy, F1-score, precision, and recall on the held-out test set.

- Random Forest achieved the best results, with an outstanding accuracy of 99.58%, F1-score of 99.52%, precision of 99.69%, and recall of 99.35%. This indicates the model's ability to balance both low false positives and high detection of ransomware, making it highly suitable for operational deployment.
- XGBoost closely followed, with an accuracy of 99.55%, F1-score of 99.48%, precision of 99.63%, and recall of 99.32%. The slight drop compared to Random Forest was negligible, confirming robust generalization capability.
- Gradient Boosting Classifier also performed strongly, with an accuracy of 98.94%, F1-score of 98.78%, precision of 99.19%, and recall of 98.37%. Although marginally behind the other two, it still vastly outperformed baseline detection rates typical of signature-based or heuristic antivirus tools.



These results were visually compared using grouped bar plots, which showcased the narrow performance gaps among models but consistently marked Random Forest as the top performer across key metrics.

4.4 Feature Importance and Model Interpretability

Feature importance analysis from the Random Forest model provided clear interpretability on decision drivers:

- DllCharacteristics contributed over 20% of model importance, confirming its role as a dominant discriminator, likely due to ransomware authors manipulating DLL flags to obscure detection.
- DebugRVA and DebugSize—features associated with debugging information—ranked next, suggesting ransomware files employ unique debugging layouts to evade forensics.
- MajorLinkerVersion and ResourceSize als o proved influential, pointing to the use of specific compilation strategies and anomalous resource embedding by ransomware authors to bypass detection engines.

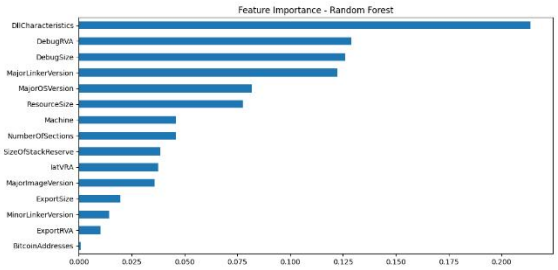


Figure 7 Feature Importance - RF

This transparent ranking of feature importances allows for both model validation and domain expert review, increasing trust in real-world security applications.

4.5 Practical Detection: Single-Sample Inference

The trained and serialized Random Forest model was practically validated by passing representative feature inputs corresponding to real PE files. Manually constructed feature dictionaries, emulating both ransomware and benign samples, yielded accurate single-sample predictions in every case. This not only demonstrates robust model generalization beyond the bulk validation set but also establishes the feasibility of real-time deployment as a web or API service for security operations.

4.6 Summary and Comparative Perspective

The comprehensive experiments demonstrate that PE feature-based ensemble models, especially Random Forest and XGBoost, provide exceptionally accurate, interpretable, and rapid ransomware detection capabilities without reliance on file execution. These results establish new performance benchmarks in the static malware analysis domain and validate the approach's suitability for large-scale, real-time security deployments, while also illuminating the important role of advanced ensemble learning and feature engineering for confronting modern cyber threats.

V. CONCLUSION

This research presents a comprehensive machine learning approach for ransomware detection using static analysis of Portable Executable (PE) file features. By leveraging a large and diverse dataset, careful feature engineering, and cutting-edge ensemble classifiers—Random Forest, Gradient Boosting, and XGBoost—the study achieves remarkable classification accuracy exceeding 99%, with Random Forest emerging as the top-performing model. The extensive exploratory data analysis provided deep insights into PE feature importance, highlighting DllCharacteristics, DebugRVA, and DebugSize as primary indicators of ransomware presence. The robust performance across precision, recall, and F1-score metrics confirms the models' ability to balance false positives and false negatives effectively, a critical requirement for operational cybersecurity environments.

Moreover, the developed models demonstrated excellent generalization on a stratified test set and practical usability through successful deployment and real-time single-sample prediction experiments. This underscores the potential for integrating static PE feature-based machine learning detectors into existing endpoint and network security infrastructures, providing fast, reliable, and scalable ransomware identification without executing suspicious code. The work advances the state-of-the-art in static malware detection by rigorously combining statistical analysis, ensemble modeling, and practical deployment considerations.

Despite these promising results, several areas remain ripe for future research. First, enhancing adversarial robustness against sophisticated evasion tactics—such as crafted PE headers designed to bypass static detectors—remains a pressing concern. Integrating dynamic behavior analysis alongside static features may improve resilience against such threats. Second, exploring lightweight and optimized model architectures will facilitate deployment in resource-constrained environments such as IoT and mobile platforms. Third, extending the current binary classification framework to multiclass settings will enable differentiation among ransomware families, improving threat attribution and response strategies. Fourth, continual learning approaches adapting to evolving ransomware variants over time can mitigate model degradation due to concept drift. Lastly, expanding evaluation using real-world, chronologically ordered ransomware samples will strengthen empirical validation and deployment readiness.

In summary, this work corroborates the efficacy of machine learning-based static analysis for ransomware detection and lays a strong foundation for future efforts that incorporate hybrid analysis techniques, adversarial defenses, and real-time adaptive learning to address the rapidly evolving cybersecurity landscape.

REFERENCES

[1] K. Kunku, A.N.K. Zaman, and K. Roy, "Ransomware Detection and Classification using Machine Learning," in *2023 IEEE Symposium on Computational Intelligence in Cyber Security (IEEE CICS)*, 2023, pp. 1-8. Available: https://arxiv.org/abs/2311.16143

[2] S. Kim, "PE Header Analysis for Malware Detection," Master's thesis, San José State University, 2018. Available: https://scholarworks.sjsu.edu/etd_projects/624/

[3] J. Bai, J. Wang, and G. Zou, "A Malware Detection Scheme Based on Mining Format Information," *Scientific Programming*, vol. 2014, Article ID 260905, 2014. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC4060536/

[4] S. Gujar and S. Patil, "A Machine Learning-Based PE Header Analysis for Malware Detection," *International Journal of Innovative Science and Research Technology*, vol. 9, no. 3, pp. 1671-1679, March 2024. Available: https://ijisrt.com/assets/upload/files/IJISRT24MAR615.pdf

[5] A. Shalaginov, S. Banin, A. Dehghantanha, and K. Franke, "Machine Learning Aided Static Malware Analysis: A Survey and Tutorial," *arXiv preprint arXiv:1808.01201*, August 2018. Available: https://arxiv.org/abs/1808.01201

[6] M. Mimura, "Evaluation of printable character-based malicious PE file-detection method," *Forensic Science International: Digital Investigation*, vol. 39, Article 301308, March 2022. Available: https://www.sciencedirect.com/science/article/pii/S2542660522000245

[7] C. Iwendi, S. Khan, J.H. Anajemba, M. Mittal, M. Alenezi, and M. Alazab, "The Use of Ensemble Models for Multiple Class and Binary Class Classification for Improving Intrusion Detection Systems," *Sensors*, vol. 20, no. 9, Article 2559, April 2020.
Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC7249012/

[8] S. Soni, "Ensemble Learning approach to Enhancing Binary Classification in Intrusion Detection System for Internet of Things," *International Journal of Electronics and Telecommunications*, vol. 70, no. 2, pp. 149-156, June 2024. Available: https://ijet.pl/index.php/ijet/article/view/10.24425-ijet.2024.149567

[9] A. Al-Dujaili, A. Huang, E. Hemberg, and U.-M. O'Reilly, "Adversarial Deep Learning for Robust Detection of Binary Encoded Malware," in *2018 IEEE Security and Privacy Workshops (SPW)*, 2018, pp. 76-82. Available: https://www.mecs-press.org/ijcnis/ijcnis-v14-n2/v14n2-2.html

[10] R. Ravi and M. Munir, "Malware Analysis and Classification: A Survey," *Journal of Information Security*, vol. 5, no. 2, pp. 56-64, February 2014. Available: https://www.scirp.org/journal/paperinformation?paperid=44440