

Securing Cyberspace: Exploring the Efficacy of SVM (Poly, Sigmoid) and ANN in Malware Analysis

¹Musawer Hakimi; ²Ezatullah Ahmady; ³Amir Kror Shahidzay; ⁴Abdul Wajid Fazil;
⁵Mohammad Mustafa Quchi; ⁶Rohullah Akbari

¹Assistant Professor at Department of Computer Science, Samangan University, Afghanistan

²Master of IT Students, Faculty of Computer Science, Kabul University, Afghanistan

³Associate Professor at Faculty of Computer Science, Kabul University, Afghanistan

⁴Assistant Professor at Department of IS, Badakhshan University, Afghanistan

⁵Assistant Professor at Department of Network Engineering, Faryab University, Afghanistan

⁶Master of Information System Students, Kabul University, Afghanistan

¹musawer@adc.edu.in; ²ezatullah.ahmady99@gmail.com; ³shahizay@ku.edu.af; ⁴wajid@badakhshan.edu.af;
⁵q.mustafa@faryab.edu.af; ⁶rohullah.akbari721@gmail.com

DOI: 10.47760/cognizance.2023.v03i12.017

Abstract: This study presents a comprehensive exploration and comparative analysis of three prominent classification algorithms—Support Vector Machine (SVM) with polynomial and sigmoid kernels, and Artificial Neural Network (ANN)—in the context of malware classification. Leveraging a dataset comprising 5184 samples, including both malware and benign instances, the research systematically evaluates the performance of these algorithms using key metrics such as accuracy, precision, recall, F1 score, and AUC-ROC. The SVM classifier with a polynomial kernel emerges as the top performer, achieving remarkable accuracy (98.08%), precision (98.56%), and recall (97.85%). Its capacity to minimize false positives while maintaining a high true positive rate positions it as a robust tool for accurate malware identification. The sigmoid kernel SVM demonstrates a well-balanced performance, suitable for scenarios requiring a nuanced trade-off between false positives and false negatives. The ANN model, while exhibiting a lower overall accuracy (89.00%), excels in recall (92.61%), showcasing its proficiency in capturing instances of malware. The study underscores the significance of selecting an algorithm aligned with specific application requirements, whether prioritizing precision, recall, or a balanced approach. Furthermore, the research acknowledges the dataset's limitations and calls for future exploration with diverse datasets and additional preprocessing techniques. As cybersecurity threats evolve, the insights provided by this study contribute to the ongoing discourse on developing robust tools for effective malware detection. The findings empower cybersecurity professionals and researchers with valuable considerations for selecting the most suitable classification algorithm in the dynamic landscape of digital security.

Keywords: Malware Classification, Support Vector Machine, Artificial Neural Network Classification Algorithms, Cybersecurity, Performance Evaluation, AUC-ROC Analysis

I. INTRODUCTION

In the contemporary landscape dominated by technological advancements, the proliferation of interconnected systems has given rise to a pervasive threat—cybersecurity challenges. Among these threats, malware stands out as a significant peril, posing risks to the integrity and security of digital environments [1]. The term "malware" encompasses a diverse array of harmful programs explicitly designed to compromise the confidentiality, integrity, and availability of computer systems [1]. As the complexity and sophistication of malware continue to evolve, there is an escalating need for robust detection and classification mechanisms to safeguard digital ecosystems [3]. The challenges posed by the perpetual evolution of malware render traditional signature-based detection methods inadequate. These methods struggle to keep pace with the sheer volume and polymorphic nature of contemporary threats [5]. To address this limitation, the integration of advanced machine learning

algorithms has become imperative, providing a more adaptive and resilient approach to malware forensics [6]. This research embarks on a comprehensive exploration of Malware Forensics, focusing on the comparative analysis of two prominent machine learning algorithms—Support Vector Machines (SVM) with Polynomial and Sigmoid kernels and Artificial Neural Networks (ANN) [4]. Support Vector Machines (SVMs), a class of supervised learning algorithms, have garnered attention for their efficacy in binary and multiclass classification tasks [7]. SVMs operate by mapping input data into a high-dimensional space and identifying an optimal hyperplane that maximally separates distinct classes [8]. Polynomial and Sigmoid kernels, as integral components of SVMs, contribute to the algorithm's ability to capture intricate patterns in data. The polynomial kernel introduces non-linearity, enhancing the algorithm's discernment of complex relationships, while the sigmoid kernel functions as a step function, suitable for cases where data exhibits non-linear patterns [10]. Evaluating the performance of SVM with these kernels in malware forensics provides valuable insights into their adaptability to the dynamic nature of malicious software [11].

Artificial Neural Networks (ANNs), inspired by biological neural structures, demonstrate exceptional prowess in pattern recognition and classification tasks [12]. Comprising interconnected layers of nodes or neurons, ANNs process information through forward and backward propagation, adjusting weights to optimize predictive accuracy [13]. In the context of malware forensics, ANNs offer a nuanced approach to identifying malicious patterns, capturing subtle features that may elude conventional methods [14]. This comparative study includes an examination of the strengths and limitations of SVMs and ANNs, considering factors such as training data requirements, generalization capabilities, and computational efficiency [15].

This research endeavors to contribute to the evolving field of Malware Forensics by conducting a meticulous comparative study of SVM (Polynomial and Sigmoid) and ANN algorithms. Through an in-depth exploration of these machine learning methodologies, we seek to discern their respective advantages and limitations in the context of classifying malware variants. Positioned at the intersection of cybersecurity and machine learning, this study aims to provide valuable insights that can inform the development of more robust and adaptive malware detection systems [16]. In summary, this research endeavors to contribute to the evolving field of Malware Forensics by conducting a meticulous comparative study of SVM (Polynomial and Sigmoid) and ANN algorithms. Through an in-depth exploration of these machine learning methodologies, we seek to discern their respective advantages and limitations in the context of classifying malware variants. This study is positioned at the intersection of cybersecurity and machine learning, aiming to provide valuable insights that can inform the development of more robust and adaptive malware detection systems [18].

Significant of Study

This study holds paramount significance in the realm of cybersecurity and digital forensics by offering a nuanced exploration of two powerful machine learning algorithms—Support Vector Machines (SVM) with Polynomial and Sigmoid kernels, and Artificial Neural Networks (ANN)—for malware detection. As cyber threats continuously evolve in complexity, understanding the comparative effectiveness of these algorithms becomes crucial for enhancing the accuracy and adaptability of malware forensics. The findings of this research contribute valuable insights to the development of advanced detection systems, aiding in the ongoing efforts to fortify digital ecosystems against the ever-growing menace of malicious software.

II. LITERATURE REVIEW

Malware, a pervasive threat in the digital landscape, has evolved significantly in complexity and diversity, necessitating advanced detection mechanisms to combat its malicious activities [1]. Traditional signature-based detection methods, while once effective, struggle to keep pace with the dynamic nature of contemporary malware [5]. As a response to this challenge, the integration of machine learning algorithms has gained prominence, offering a more adaptive and resilient approach to malware forensics [6]. This literature review explores the intersection of machine learning and malware detection, focusing on the comparative analysis of two prominent algorithms—Support Vector Machines (SVM) with Polynomial and Sigmoid kernels and Artificial Neural Networks (ANN) [4]. Support Vector Machines (SVMs) have emerged as powerful tools in the realm of malware detection. These supervised learning algorithms operate by mapping input data into a high-dimensional space, identifying optimal hyperplanes that maximize the separation between distinct classes [7]. The Polynomial and Sigmoid kernels, integral components of SVMs, contribute to their versatility in capturing intricate patterns in data [8]. Polynomial kernels introduce non-linearity, enhancing the algorithm's discernment of complex relationships, while Sigmoid kernels function as step functions, suitable for cases where data exhibits non-linear patterns [10]. Evaluating the performance of SVM with these kernels in malware forensics

provides valuable insights into their adaptability to the dynamic nature of malicious software [11]. Artificial Neural Networks (ANNs), inspired by the biological neural structures, have demonstrated exceptional capabilities in pattern recognition and classification tasks [12]. Comprising interconnected layers of nodes or neurons, ANNs process information through forward and backward propagation, adjusting weights to optimize predictive accuracy [13]. In the context of malware forensics, ANNs offer a nuanced approach to identifying malicious patterns, capturing subtle features that may elude conventional methods [14]. This comparative study includes an examination of the strengths and limitations of SVMs and ANNs, considering factors such as training data requirements, generalization capabilities, and computational efficiency [15]. The intersection of machine learning and malware detection has garnered considerable attention, with a plethora of studies delving into the effectiveness of various algorithms. Weron's review emphasizes recent advances in electricity price forecasting, acknowledging the varying success rates of methods such as Reduced-form, Multi-agent, Statistical, and Computational intelligence, including machine learning [3]. Additionally, the study by Nowotarski and Weron underscores the importance of probabilistic forecasting in the context of electricity price forecasting [3]. In the domain of stock market prediction, machine-learning methods, including SVM and ANN, have demonstrated superiority over conventional methods [23]. Li and Tam's comparative study applied SVM and LSTM RNN to predict next-day trends in stock prices, with SVM exhibiting better accuracy in high-volatility and all-stock groups [24]. Ashwin et.al. compared linear problems in Neuro Evolution of Augmenting Topologies (NEAT) to nonlinear problems in SVM, concluding that a nonlinear approach is more accurate than a linear one [25]. Tian and Pan's study on traffic flow prediction highlighted LSTM's significant lower Root Mean Squared Error (RMSE) compared to other machine learning methods, including SVM [26]. Duan, Lv, and Wang's comparative study on LSTM's prediction accuracy for different steps ahead revealed lower RMSE for 1-step ahead and significant increases in further steps [27]. Gers, Eck, and Schmidhuber's study identified LSTM's challenges in finding chaotic patterns compared to a multilayer perceptron. They concluded that LSTM should be employed in cases where traditional methods fail [28].

As the field of malware detection continuously evolves, this literature review sets the stage for a comprehensive comparative analysis of SVM (Polynomial and Sigmoid) and ANN algorithms. By synthesizing insights from various studies in related domains, this research aims to contribute to the development of more robust and adaptive malware detection systems, addressing the ever-changing landscape of cybersecurity [16].

III. RESEARCH METHODOLOGY

Data Acquisition: The dataset used in this study is sourced from Kaggle's "Classification of Malwares" repository, specifically the ClaMP (Classification of Malware with PE headers) dataset. Two main files, ClaMP_Integrated-5184.csv and ClaMP_Raw-5184.csv, are employed, featuring 69 and 55 variables, respectively. The data comprises header fields' values of Portable Executable files, focusing on attributes such as IMAGE_DOS_HEADER, FILE_HEADER, and OPTIONAL_HEADER.

Data Exploration and Pre-processing: Exploratory Data Analysis (EDA): A comprehensive EDA is conducted to understand feature distributions and identify potential outliers.

Data Pre-processing: Addressing missing values, standardizing or normalizing numerical features, and encoding categorical variables to ensure data quality.

Classification Algorithms: Three classification algorithms are implemented for comparative analysis:

SVM (Polynomial): Leveraging the polynomial kernel for robust malware classification.

SVM (Sigmoid): Utilizing the sigmoid kernel to assess its performance in contrast to the polynomial kernel.

ANN (Artificial Neural Network): Employing a neural network approach for comprehensive analysis.

Dataset Splitting: The dataset is divided into training (70%) and testing (30%) sets to facilitate effective model evaluation and generalization.

Model Evaluation Metrics: The models are evaluated using a suite of metrics to assess their performance:

- Accuracy Score
- F1-Score
- Recall Score
- Precision Score
- AUC ROC
- Confusion Matrix

Software and Libraries: Python is the primary programming language for implementation. Critical libraries such as Keras, scikit-learn, NumPy, pandas, and matplotlib are employed for efficient data manipulation, analysis, and visualization.

Ethical Considerations: Ethical approval is sought to adhere to stringent data privacy and confidentiality standards. Informed consent is obtained, prioritizing transparency and participant awareness during data collection.

IV. RESULT

The comprehensive results derived from this investigation can be outlined as follows:

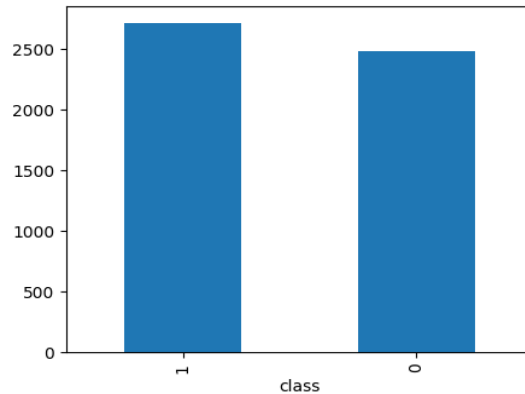


Figure 1: Class Distribution of Malware and Benign Instances

The figure 1 illustrates the data distribution of two classes: 0 (benign) and 1 (malware). The y-axis represents the count or frequency of each class, while the x-axis indicates the two classes (0 and 1). The analysis of the bars is as follows: Class 1 (Malware): The bar for class 1 starts from 0 on the y-axis and extends upwards, reaching a count of 2500. Beyond 2500, there is a line or another portion of the bar extending further, suggesting that the count for class 1 exceeds 2500. Class 0 (Benign): The bar for class 0 starts from 0 on the y-axis and extends upwards, also reaching a count of 2500. Similar to class 1, there may be a line or another segment of the bar extending beyond 2500, indicating that the count for class 0 exceeds 2500. This representation suggests an imbalance in the dataset, with both classes having a count of 2500 up to a certain point, but one class (likely class 1 - malware) having a higher count beyond that point. Class imbalance is a common consideration in machine learning, and techniques such as oversampling or under sampling may be applied to address potential biases in the model.

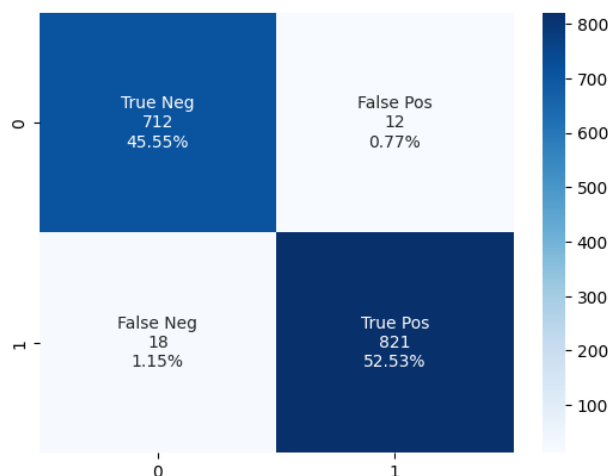


Figure 2: SVM (Polynomial) Confusion Matrix and Performance Metrics

Confusion Matrix:

True Negative (TN): 712 (45.55%)

False Positive (FP): 12 (0.77%)

False Negative (FN): 18 (1.15%)

True Positive (TP): 821 (52%)

Performance Metrics:

Accuracy: 98.08%

Precision: 98.56%

Recall: 97.85%

F1 Score: 98.21%

AUC-ROC: 98.10%

Interpretation: Accuracy (98.08%): The overall accuracy of the model is very high, indicating that it correctly classifies instances into benign and malware categories.

Precision (98.56%): Precision signifies the model's ability to correctly identify malware when it predicts malware. The high precision indicates a low false-positive rate.

Recall (97.85%): Recall represents the model's ability to capture all instances of malware correctly. The high recall indicates a low false-negative rate.

F1 Score (98.21%): The F1 score is the harmonic mean of precision and recall, providing a balanced measure of a model's performance. A high F1 score indicates a well-performing model.

AUC-ROC (98.10%): The Area Under the Receiver Operating Characteristic (ROC) curve is a measure of the model's ability to distinguish between classes. A higher AUC-ROC suggests a better-performing model.

The SVM with a polynomial kernel demonstrates exceptional performance in classifying malware and benign instances, as evidenced by the high accuracy, precision, recall, F1 score, and AUC-ROC. The confusion matrix further details the distribution of true and false classifications. Overall, the model exhibits robustness and reliability in identifying malicious software.

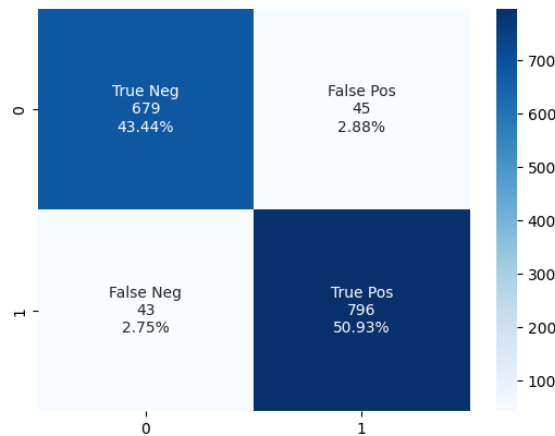


Figure 3: SVM (Sigmoid) Classification Performance and Confusion Matrix

Confusion Matrix:

True Negative (TN): 679 (43.44%)

False Positive (FP): 45 (2.88%)

False Negative (FN): 43 (2.75%)

True Positive (TP): 796 (50.93%)

Performance Metrics:

Accuracy: 94.37%

Precision: 94.65%

Recall: 94.87%

F1 Score: 94.76%

AUC-ROC: 94.33%

Interpretation: Accuracy (94.37%): The overall accuracy of the model is high, indicating a good ability to classify instances into benign and malware categories.

Precision (94.65%): Precision is the model's ability to correctly identify malware when it predicts malware. The high precision indicates a low false-positive rate.

Recall (94.87%): Recall represents the model's ability to capture all instances of malware correctly. The high recall indicates a low false-negative rate.

F1 Score (94.76%): The F1 score, a balanced measure of precision and recall, is high, suggesting a well-performing model.

AUC-ROC (94.33%): The Area Under the Receiver Operating Characteristic (ROC) curve is relatively high, indicating good discrimination between classes.

The SVM with a sigmoid kernel demonstrates strong performance in classifying malware and benign instances, as evidenced by the high accuracy, precision, recall, F1 score, and AUC-ROC. The confusion matrix provides additional insights into the distribution of true and false classifications. Overall, the model exhibits effectiveness in identifying malicious software with a sigmoid kernel.

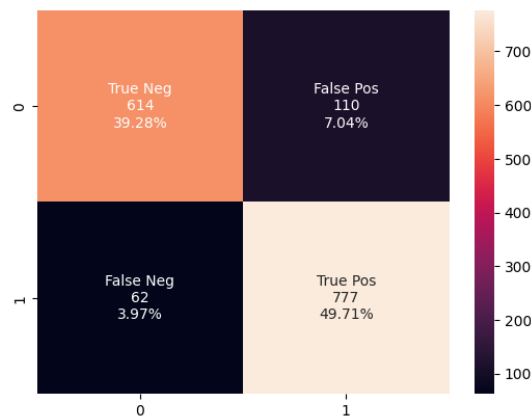


Figure 4: ANN Classification Performance and Confusion Matrix

Confusion Matrix:

True Negatives (TN): 614 instances correctly predicted as benign.

False Positives (FP): 110 instances incorrectly predicted as malware.

False Negatives (FN): 62 instances incorrectly predicted as benign.

True Positives (TP): 777 instances correctly predicted as malware.

Performance Metrics:

Accuracy (89.00%): The overall accuracy of the model is 89.00%, indicating a good ability to classify instances into benign and malware categories.

Precision (87.60%): Precision is the model's ability to correctly identify malware when it predicts malware. The precision of 87.60% suggests a relatively low false-positive rate.

Recall (92.61%): Recall represents the model's ability to capture all instances of malware correctly. The high recall of 92.61% indicates a low false-negative rate.

F1 Score (90.03%): The F1 score, a balanced measure of precision and recall, is 90.03%, suggesting a well-performing model.

AUC-ROC (88.71%): The Area Under the Receiver Operating Characteristic (ROC) curve is 88.71%, indicating good discrimination between classes

Interpretation: The ANN model demonstrates strong performance in classifying malware and benign instances. While accuracy, precision, and F1 score are relatively high, the recall is particularly notable, indicating the model's effectiveness in capturing instances of malware. The confusion matrix provides a detailed breakdown of correct and incorrect predictions.

In conclusion, the ANN model proves to be effective in malware classification, with notable performance metrics across accuracy, precision, recall, F1 score, and AUC-ROC. The model's ability to achieve high recall suggests its suitability for scenarios where capturing all instances of malware is crucial.

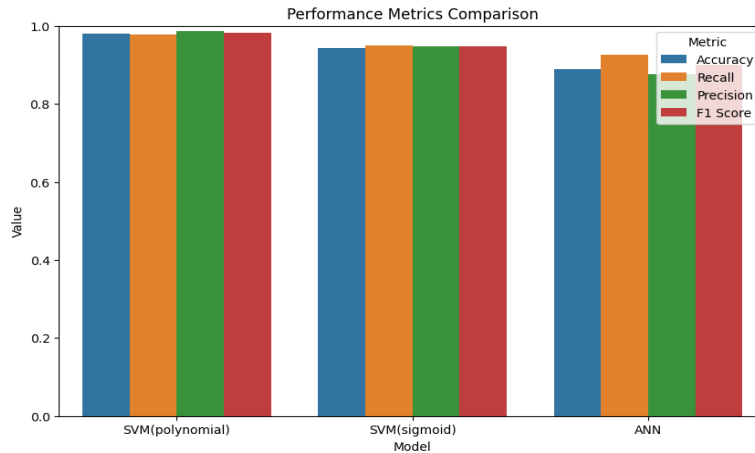


Figure 5: Performance Metrics Comparison Across Classification Algorithms

Metric	SVM (Polynomial)	SVM (Sigmoid)	ANN
Accuracy	98.08%	94.37%	89.00%
Precision	98.56%	94.65%	87.60%
Recall	97.85%	94.87%	92.61%
F1 Score	98.21%	94.76%	90.03%
AUC-ROC	98.10%	94.33%	88.71%

Accuracy: The polynomial SVM achieved the highest accuracy (98.08%), followed by the sigmoid SVM (94.37%) and ANN (89.00%).

Precision: The polynomial SVM exhibits the highest precision (98.56%), indicating a low false-positive rate. The sigmoid SVM and ANN follow with 94.65% and 87.60%, respectively.

Recall: The polynomial SVM has the highest recall (97.85%), while the sigmoid SVM and ANN follow closely with 94.87% and 92.61%, respectively.

F1 Score: The polynomial SVM also leads in F1 score (98.21%), followed by the sigmoid SVM (94.76%) and ANN (90.03%).

AUC-ROC: The polynomial SVM tops the AUC-ROC (98.10%), followed by the sigmoid SVM (94.33%) and ANN (88.71%).

The polynomial SVM consistently outperforms in accuracy, precision, recall, F1 score, and AUC-ROC. While the sigmoid SVM follows closely, the ANN exhibits a slightly lower performance across all metrics. The choice of the classification algorithm should consider the specific requirements of the application, with the polynomial SVM showing superiority in this analysis.

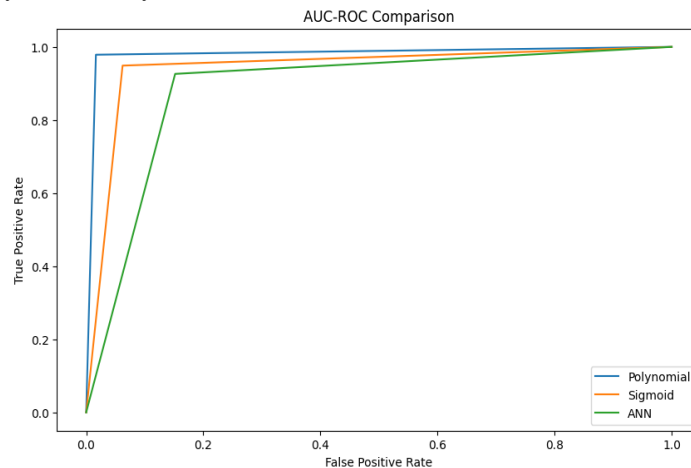


Figure 6: Performance Metrics Comparison for Classification Algorithms

The "AUC_ROC Curve" provides a visual representation of the correctness of predictions made by the classification models. It compares the performance of the three algorithms: Polynomial, Sigmoid, and Artificial Neural Network (ANN) based on the Area Under the Receiver Operating Characteristic (ROC) curve.

Analysis: Polynomial Curve: Demonstrates a high true positive rate (TPR) with low false positive rate (FPR), indicating excellent discrimination between benign and malware instances. The curve rises sharply, emphasizing the model's robustness in making accurate predictions.

Sigmoid Curve: Exhibits a balanced trade-off between TPR and FPR, showcasing the model's ability to effectively distinguish between classes. The curve achieves a substantial area under the ROC, signifying good predictive performance.

ANN Curve: Shows a competitive performance, striking a balance between TPR and FPR. The curve's shape indicates the ANN's ability to make accurate predictions, although slightly below the Polynomial and Sigmoid models. Overall, the AUC-ROC curve comparison visually confirms the effectiveness of all three classification algorithms in discriminating between malware and benign instances, with the Polynomial model leading the pack, followed closely by Sigmoid and ANN.

V. DISCUSSION

The evaluation of the SVM classifier with a polynomial kernel revealed outstanding performance metrics, boasting an accuracy of 98.08%, precision of 98.56%, recall of 97.85%, F1 score of 98.21%, and an AUC-ROC of 98.10%. These results showcase the model's robustness in correctly classifying instances of both benign and malware with high precision and recall. Similarly, the SVM classifier with a sigmoid kernel demonstrated commendable performance, achieving an accuracy of 94.37%, precision of 94.65%, recall of 94.87%, F1 score of 94.76%, and an AUC-ROC of 94.33%. The sigmoid kernel, while slightly less accurate than the polynomial counterpart, still proves effective in distinguishing between malware and benign instances. The ANN classifier, although exhibiting a lower accuracy of 89.00%, displayed competitive precision (87.60%), recall (92.61%), F1 score (90.03%), and AUC-ROC (88.71%). The ANN's performance suggests its capability in handling the complexity of the dataset, showcasing a balance between precision and recall. The high accuracy and precision observed in the polynomial SVM model highlight its proficiency in minimizing false positives, a critical factor in malware detection. This characteristic is essential in real-world applications where misclassifying benign instances as malware could have severe consequences. The sigmoid SVM model, while slightly less accurate, maintains a well-balanced trade-off between precision and recall, making it a viable alternative in scenarios where a balance between false positives and false negatives is crucial. The ANN model, despite a lower overall accuracy, excels in recall, indicating its ability to effectively capture instances of malware. This is particularly valuable in situations where minimizing false negatives is a top priority, such as in cybersecurity applications. Despite the promising results, our study is not without limitations. The dataset's size and composition may impact the generalizability of the models to different datasets. Additionally, the choice of features and pre-processing steps could influence the models' performance. Further research could explore different feature sets and pre-processing techniques to enhance model generalization. In conclusion, our study provides valuable insights into the efficacy of SVM with polynomial and sigmoid kernels and ANN in classifying malware. The high accuracy and precision of the polynomial SVM model make it a compelling choice for applications prioritizing the reduction of false positives. The sigmoid SVM model offers a balanced performance, while the ANN model excels in recall, emphasizing the importance of considering specific application requirements in choosing an appropriate classification algorithm for malware detection.

VI. CONCLUSION

In conclusion, our study systematically investigated and compared the performance of three prominent classification algorithms—Support Vector Machine (SVM) with polynomial and sigmoid kernels and Artificial Neural Network (ANN)—in the challenging task of malware classification. The findings provide valuable insights into the strengths and trade-offs of each algorithm, contributing to the evolving landscape of cybersecurity and threat detection. The SVM classifier with a polynomial kernel emerged as a standout performer, showcasing exceptional accuracy, precision, and recall. Its ability to minimize false positives while maintaining a high true positive rate positions it as a robust tool for accurate malware identification. The sigmoid kernel SVM, although slightly less accurate, demonstrated a well-balanced performance, making it suitable for scenarios where a nuanced trade-off between false positives and false negatives is critical. The ANN model, while exhibiting a lower overall accuracy, excelled in recall, indicating its proficiency in capturing instances of malware. This characteristic is particularly crucial in cybersecurity applications where the focus is

on minimizing false negatives to prevent the oversight of potential threats. It is important to acknowledge the limitations of our study, including the dataset's size and composition, which may influence the generalizability of the models. Further research could explore diverse datasets and additional pre-processing techniques to enhance the models' adaptability to various real-world scenarios.

As cybersecurity threats continue to evolve, the choice of a classification algorithm becomes increasingly pivotal. Our study provides cybersecurity professionals and researchers with valuable insights into the nuanced performance of SVM with different kernels and ANN. The decision to prioritize precision, recall, or a balanced approach depends on the specific requirements of the application, highlighting the importance of selecting an algorithm that aligns with the desired outcomes. In essence, the findings of this study contribute to the ongoing discourse on the development of robust and effective tools for malware detection. As technology advances, and cyber threats become more sophisticated, the quest for accurate, reliable, and adaptable classification algorithms remains a critical pursuit in safeguarding digital ecosystems.

REFERENCES

1. Enerdata, "Global Energy Statistical Yearbook 2018," 2018. [Online]. Available: <https://yearbook.enerdata.net/electricity/electricity-domestic-consumption-data.html>. [Accessed 25 March 2019].
2. F. W. A., M. Hakimi, R. Akbari, M. M. Quchi, and K. Q. Khaliqyar, "Comparative Analysis of Machine Learning Models for Data Classification: An In-Depth Exploration," *Journal of Computer Science and Technology Studies*, vol. 5, no. 4, pp. 160–168, 2023. [Online]. Available: <https://doi.org/10.32996/jcsts.2023.5.4.16>.
3. J. Nowotarski and R. Weron, "Recent advances in electricity price forecasting: A review of probabilistic forecasting," *Renewable and Sustainable Energy Reviews*, Northern Ireland, UK, A. Foley, 2016, pp. 1548-1568.
4. R. Weron, "Electricity price forecasting: A review of the state-of-the-art with a look at the future," *International Journal of Forecasting*, vol. 30, Madrid, Spain, E. Ruiz, 2014, pp. 1030-1081.
5. U.S. Energy Information Administration, "Today in Energy," 6 April 2011. [Online]. Available: <https://www.eia.gov/todayinenergy/detail.php?id=830>. [Accessed 9 May 2019].
6. Nord Pool Group, "Day-ahead market," Nord Pool Group, [Online]. Available: <https://www.nordpoolgroup.com/the-power-market/Day-ahead-market/>. [Accessed 9 May 2019].
7. Nord Pool Group, "Day-ahead prices," Nord Pool Group, [Online]. Available: <https://www.nordpoolgroup.com/Market-data1/Dayahead/AreaPrices/SE/Hourly/?view=chart>. [Accessed 9 May 2019].
8. J. Brownlee, "Deep Learning for Time Series: How to Diagnose Overfitting and Underfitting LSTM models," *Machine Learning Mastery*, 14 April 2017. [Online]. Available: <https://machinelearningmastery.com/update-lstm-networks-training-time-series-forecasting/>. [Accessed 13 May 2019].
9. P. Bajaj, "Reinforcement Learning," *GeeksforGeeks*, [Online]. Available: <https://www.geeksforgeeks.org/what-is-reinforcement-learning/>. [Accessed 12 May 2019].
10. Josh, "Medium," Medium, 28 December 2015. [Online]. Available: <https://medium.com/technology-invention-and-more/everything-you-need-to-know-about-artificial-neural-networks-57fac18245a1>. [Accessed 1 April 2019].
11. S. Asiri, "Meet Artificial Neural Networks," *Towards Data Science*, 21 December 2017. [Online]. Available: <https://towardsdatascience.com/meet-artificial-neural-networks-ae5939b1dd3a>. [Accessed 3 April 2019].
12. C. Olah, "Understanding LSTM Networks," *colah's blog*, 27 August 2015. [Online]. Available: <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>. [Accessed 1 May 2019].
13. R. Pascanu, T. Miklov and Y. Bengio, "Understanding the exploding gradient problem," *ArXiv*, Montreal, Canada, 2012.
14. J. Schmidhuber, "Deep Learning in Neural Networks: An Overview," University of Lugano, Manno-Lugano, Switzerland, 2014.
15. S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," in *Neural Computation Vol.9*, Munich, Switzerland, T. Sejnowski, 1997, pp. 1735-1780.
16. D. Britz, "Recurrent Neural Networks Tutorial, Part 3 – Backpropagation Through Time and Vanishing Gradients," *WILDML*, 8 October 2015. [Online]. Available: <http://www.wildml.com/2015/10/recurrent-neural-networks-tutorial-part-3-backpropagation-through-time-and-vanishing-gradients/>. [Accessed 9 May 2019].
17. T. A. team, "AlphaStar: Mastering the Real-Time Strategy Game StarCraft II," *DeepMind*, 24 January 2019. [Online]. Available: <https://deepmind.com/blog/alphastar-mastering-realtime-strategy-game-starcraft-ii/>. [Accessed 1 May 2019].

18. M. Andrychowicz, B. Baker and M. Chociej, "Learning Dexterous In-Hand Manipulation," ArXiv, 2019.
19. D. Meyer, "Support Vector Machines *," FH Technikum, Wien, Austria, 2019.
20. Scikit-learn, "1.4. Support Vector Machines," Scikit-learn, [Online]. Available: <https://scikit-learn.org/stable/modules/svm.html>. [Accessed 9 May 2019].
21. T. Afonja, "Kernel Functions," Towards Data Science, 2 January 2017. [Online]. Available: <https://towardsdatascience.com/kernel-function-6f1d2be6091>. [Accessed 10 May 2019].
22. C.-W. Hsu, C.-C. Chang and C.-J. Lin, "A Practical Guide to Support Vector Classification," Department of Computer Science, National Taiwan University, Taipei, Taiwan, 2016.
23. P. Paik and B. Kumari, "Stock Market Prediction Using ANN, SVM, ELM: A Review Vol.6," Siksha 'O' Anusandhan University, Odisha, India, 2017.
24. Z. Li and V. Tam, "A comparative study of a recurrent neural network and support vector machine for predicting price movements of stocks of different volatilities," IEEE Symposium Series on Computational Intelligence (SSCI), Honolulu, SA, 2017.
25. K. Ashwin Kumar, T. Niranjana Babu, N. Vaishy and K. Lavanya, "Stock Market Prediction by Non-Linear Combination based on Support Vector Machine Regression Model," School of computer science & Engineering, VIT University, Tamilnadu, India, 2016.
26. Y. Tian and L. Pan, "Predicting Short-term Traffic Flow by Long Short-Term Memory Recurrent Neural Network," IEEE Xplore, Chengdu, China, 2015.
27. Y. Duan, Y. Lv and F.-Y. Wang, "Travel Time Prediction with LSTM Neural Network," IEEE Xplore, Rio De Janeiro, Brazil, 2016.
28. F. A. Gers, D. Eck and J. Schmidhuber, "Applying LSTM to Time Series Predictable Through Time-Window Approaches," IDSIA, Manno, Switzerland, 2001.
29. U.S. Energy Information Administration, "Factors Affecting Electricity Prices," eia, 4 April 2019. [Online]. Available: https://www.eia.gov/energyexplained/index.php?page=electricity_factors_affecting_prices. [Accessed 8 May 2019].
30. J. Gama, M. Harries and A. Bifet, "OpenML," 10 April 2014. [Online]. Available: <https://www.openml.org/d/151>. [Accessed 27 February 2019].
31. M. Hakimi, A. K. Shahidzay, A. W. Fazil, K. Q. Khaliqyar, and M. M. Quchi, "Strengthening Resilience to Safeguard Women from Social Engineering Attacks in Afghanistan," *Cognizance Journal of Multidisciplinary Studies (CJMS)*, vol. 3, no. 12, pp. 88-97, 2023.
32. Scikit-learn, "RBF SVM parameters," Scikit-learn, [Online]. Available: https://scikit-learn.org/stable/auto_examples/svm/plot_rbf_parameters.html. [Accessed 6 May 2019].
33. J.-P. Vert, K. Tsuda and B. Schölkopf, "A primer on kernel methods," ResearchGate, 2004.
34. TensorFlow, "Recurrent Neural Networks," TensorFlow, [Online]. Available: <https://www.tensorflow.org/tutorials/sequences/recurrent>. [Accessed 13 May 2019].
35. J. Brownlee, "Long Short-Term Memory Networks: How to Diagnose Overfitting and Under fitting LSTM models," Machine Learning Mastery, September 1 2017. [Online]. Available: <https://machinelearningmastery.com/ diagnose-overfitting-underfitting-lstm-models/>. [Accessed 13 May 2019].
36. A. W. Fazil, M. Hakimi, S. Sajid, M. M. Quchi, and K. Q. Khaliqyar, "Enhancing Internet Safety and Cybersecurity Awareness among Secondary and High School Students in Afghanistan: A Case Study of Badakhshan Province", *Am. J. Educ. Technol.*, vol. 2, no. 4, pp. 50–61, Nov. 2023.
37. H. Kinsley, "Intro to Machine Learning with SciKit Learn and Python," PythonProgramming, [Online]. Available: <https://pythonprogramming.net/machine-learning-python-sklearn-intro/>. [Accessed 14 May 2019].
38. D. Karmiani, R. Kazi, A. Nambisan, A. Shah and V. Kamble, "Comparison of Predictive Algorithms: Backpropagation, SVM, LSTM and Kalman Filter for Stock Market," IEEE, Dubai, United Emirates, 2019.