NATIONAL COLLEGE OF IRELAND

MSc RESEARCH PROJECT

ENHANCING MALWARE DETECTION AND CLASSIFICATION USING DEEP LEARNING: A HIGH ACCURACY AND LOW LATENCY APPROACH

STUDENT ID: X23356022

SUPERVISOR: MARK MONAGHAN

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Ahaoma Emmanuel Mmesirionye

Student ID: X23356022

Programme: MSc Cyber Security **Year:** 2024/2025

Module: Research Project

Lecturer: Mark Monaghan

Submission Due Date: 11/08/2025

Project Title: Enhancing Malware Detection and Classification Using Deep Learning: A High Accuracy and Low Latency Approach

Word Count: 5324 **Page Count:** 33

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: A.E.M

Date: 11/08/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

TABLE OF CONTENTS

ABSTRACT

The study of malware and its detection and classification has become a very important aspect of cybersecurity, as it enables the identification and classification of several variants of malware in network operations. In this study, the proposition of a comparative analysis between four (4) deep learning techniques for the enhancement of this malware detection was carried out on a dataset obtained from Kaggle, the Windows malware that comprises the SOMLAP dataset. The four (4) deep learning algorithms proposed in this research include multilayer perceptron (MLP), convolutional neural network (CNN), long short-term memory (LSTM), and recurrent neural network (RNN). After the training and evaluation of the models, the long short-term memory (LSTM) was observed as the most efficient model for the detection of malware after achieving an accuracy of 97%. One of the few limitations encountered in this research is due to the constant evolution of malware, and while these deep learning neural network algorithms can detect patterns in the malware structure, they may require frequent retraining to adapt to newly emerging malware threats. However, the results obtained from the research underscore the potential of deep learning sequential models in the enhancement of malware detection. This research contributes to the growing field of cybersecurity, as it offers insights into the advantages of employing deep learning technologies in the quest to mitigate malicious software vulnerabilities in our industry.

KEYWORDS: malware, multilayer perceptron, convolutional neural network, long short-term memory, recurrent neural network, accuracy, cybersecurity.

INTRODUCTION

In the constantly changing field of cybersecurity, the ongoing battle against malicious software, or malware, has become a recurring and enduring challenge (Panda, 2024). Conventional antivirus software has long depended on detection techniques based on signatures. In other words, malware is identified by comparing it to signatures or patterns that have already been documented in big databases (Moubarak, 2020). Even though this specific approach has proven effective in identifying known threats, it requires assistance in keeping up with the rapid proliferation of new malware and its variants. The incidence related to the development of sophisticated and state-of-the-art malware calls for the creation of more robust and adaptable solutions in cybersecurity (Martirosyan, 2023). The limitations of the conventional antivirus programs that rely on the early methods of detection are made worse by the lack of consistency among different antivirus vendors. As a result, their signature databases' scope and currency fluctuate (Corbet, 2018). The limitations previously noted highlight concerns about the effectiveness of antivirus software in detecting and eliminating malware. To address these challenges, the cybersecurity field is actively exploring innovative methods that focus on analyzing malware behavior instead of relying solely on known signatures (Soni, 2021). The rapid advancement of technology in the digital age of Industry 4.0 has significantly influenced both personal and professional aspects of daily life (Soldatos, 2019). The rise of the information society, driven by the Internet of Things (IoT) and its applications, has brought numerous benefits, but these are hindered by security challenges. Cybercriminals exploit networks and personal computers to steal sensitive data for profit or disrupt systems through denial-of-service attacks (Sharma, Gupta and Shabaz, 2022). These attackers use malicious software, known as malware, designed to harm or compromise computer network operating systems (Barnes-Proby, 2018). Confidential information is growing dramatically, and network-related activities are increasing daily. Because unauthorized users, or invaders, contain malicious and undesired behaviors, the network is vulnerable to security problems in this scenario (Reza, 2019). The hacker's primary goal is to try to get the private information. According to the different security-based network techniques, they are using cyberattacks to defeat attacks and illicit activity on physical systems. Therefore, based on the system connected to the network infiltration technique, an efficient intrusion system ought to be required to stop those attacks. There are a few intrusion systems in use today, which are meant to counteract persistent attacks and are generally

categorized as Denial of Service (DoS), Cross-Site Scripting (CSS), etc. As the digital age has brought forth unprecedented levels of interconnectivity and simplicity, malware is at the forefront of the cybersecurity issues that have arisen. Malware is a broad type of software designed to negatively impact or exploit any programmable device, service, or network (Martinelli, 2024). Within the continually changing arena of cyber dangers, malware continues to be a key challenge encountered by professionals in the field of cybersecurity. Because malware varieties are growing exponentially, robust detection systems are necessary for the security of digital infrastructure (Soni, 2024). Traditional signature-based techniques, while effective in detecting known threats, are often insufficient in detecting new, unknown, or polymorphic malware (Schrynemeeckers, 2015). Criminals in the software sector are in constant development of state-of-the-art malware attack types, and they do this by exploiting the loopholes in different software and systems at an exponential rate (Gupta, 2021). This dynamic threat environment, which is constantly evolving, calls for a strong, flexible, and advanced protective system (Sullivant, 2016). The early methods employed for the detection of this malware have largely depended on heuristic approaches (Niu, 2024). Since these heuristic techniques depend on a pre-existing compilation of known malware types, they tend to have a great deal of difficulty in identifying new malware types, even though they are effective in the identification of already existing threats. Despite offering a certain level of protection against unknown threats, these heuristic techniques are prone to high rates of false-positive outcomes. This limitation gives rise to the need for more advanced detection techniques that can successfully maneuver the complexities of dynamic malware types. Machine learning provides the capacity for the identification of pre-existing malware through the systematic process of pointing out patterns within collected datasets. A growing interest in utilizing machine learning and deep learning techniques to increase the detection accuracy of malware detection has resulted from the development of a sense of urgency in the field (Faisal, 2023). Under the general canopy of artificial intelligence, machine learning and deep learning offer powerful features for the analysis of complex data (Rane, 2024). Handwritten pattern recognition, cyber security intrusion, disease, depression, and object detection, alongside many other applications, have all utilized machine or deep learning techniques (Dhiraj, 2019). However, the preprocessing steps that are performed on the data have a big impact on the effectiveness and efficiency of these machine learning models. One of the main obstacles in malware detection is the issue of high dimensionality in the feature space of most datasets. Most of the malware datasets extracted in this era often

contain several features, many of which may be useless, having little to no information in the correlations between important features and targets, leading to increased computational demands or poor model performance. Traditional machine learning models usually process such data with difficulty without sufficient data preparation techniques (Klein, 2024). A fundamental preprocessing step called feature selection aims to minimize the problem arising from dimensionality by keeping just the most important features in the entire data collection with just the right information. However, dimensionality reduction and the preservation of important features for machine learning modeling are frequently not balanced by some of the feature selection techniques. Therefore, by identifying the most important features, feature selection aims to lower the dimensionality of the dataset, consequently increasing model performance and decreasing computing expenses. To ensure that every feature contributes equally to the model, a method known as "feature scaling" attempts to standardize the data into a specific range or distribution. To do this, special statistical techniques like normalization and min-max scaling are commonly utilized. However, in practice, malware detection methods must struggle with highly imbalanced datasets, where cases of malware are often significantly outnumbered by normal ones (Oak, 2019). This inconsistency makes misclassification more likely, especially for groups with fewer features, and it emphasizes the need for a better data preprocessing method to increase the performance of the models. The main aim of this research is to enhance the effectiveness of malware classification and detection using deep learning techniques, focusing solely on accuracy improvements (2017). The determination of malware attacks from normalcy is an especially complex binary classification task due to the presence of high dimensionality in the features of the data and a high degree of redundancy. Therefore, through the implementation of data preprocessing methods and deep learning models, the objective is to determine the most efficient approach that can augment high accuracy in the detection of malware in software (Jo, 2023). In this assessment some statistical scaling and selection methods will be utilized: min-max scaling, standard scaling, and feature selection using chi-square select k best, where k is the predefined number of features that would be needed. To conduct a thorough assessment, experiments utilizing 4 distinct deep learning models were implemented (Shin, 2016). Artificial neural network (ANN), convolutional neural network (CNN), long short-term memory (LSTM), and recurrent neural network (RNN) were the models implemented. Each model possesses distinct advantages and disadvantages. Through a comprehensive assessment of the performance of these models, the main

objective is to offer a refined approach for improved accuracy in malware detection (Indrianti, 2024).

1.1. SIGNIFICANCE OF STUDY

With the increase in complexity and sophistication of cyber threats, traditional malware detection techniques are less effective, especially against the dynamic threats in recent times (LYSENKO, 2020). This study becomes significant as it delves deeper into the application of four (4) deep learning techniques, which involve the following neural networks: Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM), recurrent neural networks (RNN), and the multilayer perceptron for the enhancement of malware detection (Chan, 2022). By utilizing these deep learning techniques, the aim of the research is to achieve higher detection accuracy and faster response times, which are crucial factors for real-time cybersecurity systems. By optimizing deep learning architectures for both high performance and efficiency, this research contributes to the development of intelligent malware defense systems that can operate effectively even in the presence of limited computational resources (Kalamkar, 2020). The results of this study would advance the academic understanding of cybersecurity in artificial intelligence and provide practical frameworks that can be implemented in industry (Bordel, 2022).

1.2. AIM AND OBJECTIVES

The aim of this study is to develop and evaluate a deep learning-based malware detection system using four (4) different deep learning algorithms, assessing their effectiveness in identifying and classifying malicious software with the following objectives:

- To implement four supervised deep learning algorithms for malware detection.
- To optimize the deep learning models for enhanced malware detection.
- To evaluate and compare the performance of the models based on the accuracy of the literature review.
- Justify the use of feature engineering in machine learning when working with network data analysis.
- Deploy the model using FastAPI.

1.3. RESEARCH QUESTIONS

- What optimization techniques can be applied to enhance the performance of deep learning models for malware detection?
- How do the selected models compare in terms of accuracy and effectiveness based on the results of the models in the review section?
- What is the significance of feature engineering in machine learning for network data analysis, and how does it affect the performance of malware detection?

1.4. LIMITATIONS

- Data Quality and Availability: Deep learning models require large volumes of high-quality, labeled datasets to perform effectively. However, acquiring comprehensive and up-to-date malware datasets can be challenging due to legal, privacy, and security concerns. Limited access to diverse malware samples may affect the model's generalizability across new or rare threats.
- Computational Resources: Training and deploying deep learning models, especially in real-time environments, often demand significant computational power (e.g., GPUs or TPUs). This could be a limitation for organizations with limited infrastructure, potentially impacting the low-latency performance the study aims to achieve.
- Adaptability to Evolving Threats: While deep learning can detect complex patterns, malware authors continuously evolve their techniques. The model may require frequent retraining to adapt to newly emerging threats, which can be resource-intensive and time-consuming.

LITERATURE REVIEW

The evolution of cyber threat complexity has led to the need for a robust malware detection and classification system. The conventional approach fails often in the detection of evolved malware due to its dependence on previously available patterns (Gary, 2025). As a result of this, deep learning techniques were introduced as a powerful alternative that offers high accuracy in identifying threats that have not been previously reported by learning patterns in data. Recent studies have explored various deep learning architectures such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) for the enhancement of performance in the detection of malicious attacks. These approaches have produced great results in terms of accuracy and other evaluation metrics. This literature review critically examines recent advancements in both classical and deep learning-based malware detection; it also compares their effectiveness in solving evolving threats. By highlighting gaps in the methodologies, this review establishes the foundation for the development of a more accurate malware detection framework using deep learning algorithms.

2.1. MALWARE DETECTION USING CLASSICAL MACHINE LEARNING ALGORITHMS

Classical machine learning algorithms have also played a crucial role in the detection of malware. These classical algorithms make use of various statistical formulas to manipulate the features in given data to classify malicious behavior (Alaba, 2025). Models such as decision trees, Support Vector Machines (SVM), random forests, and k-Nearest Neighbors (k-NN) have been adopted widely (Mogal, 2024). These methods are dependent heavily on features. engineering and statistics, offering high accuracy, but sometimes they struggle with dynamic and evolving malware threats.

2.2. MALWARE DETECTION USING DEEP LEARNING ALGORITHMS

Deep learning has changed the way of malware detection by allowing automatic feature extraction from raw data, significantly improving the model accuracy (Talbi, 2022). Deep learning models such as Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) networks and hybrid architectures can learn intricate patterns in malware (Jain, 2022). These techniques

offer a more rigid solution against evolving threats, reducing dependence on manual feature design and improving generalization across different types of malware (Kuravatti, 2025).

2.3. SUMMARY OF RELATED STUDIES ON MALWARE DETECTION USING BOTH SUPERVISED CLASSICAL AND DEEP LEARNING TECHNIQUES

Table 2.1: Literature review table

S/N	PAPER TITLE AND REFERENCE	MODELS	DATASET	RESULTS	LIMITATION	IMPROVEMENTS
1	Alomari, E.S., Nuiaa, R.R., Alyasseri, Z.A.A., Mohammed, H.J., Sani, N.S., Esa, M.I. and Musawi, B.A., 2023. Malware detection using deep	LSTM	Android malware datasets from kaggle	99.75% and 94.75% on both datasets respectively	Did not outline types of malware	Involvement of different malware types

	learning and correlation-based feature selection. Symmetry, 15(1), p.123.					
2	Akhtar, M.S. and Feng, T., 2022. Malware analysis and detection using machine learning algorithms. Symmetry, 14(11), p.2304.	KNN, CNN, Naive bayes, random forest, SVM and decision trees.	Collected from the canadian institute for Cybersecurity	Decision trees performed well with a 99% accuracy	Did not incorporate more deep learning models	Implementation of more deep learning models
3	Elayan, O.N. and Mustafa, A.M.,	SVM, KNN, DT, RF, GRU	CICAndMal 2017 dataset.	GRU had an accuracy of 98.2%	Did not incorporate more deep	Implementation of more deep

	2021. Android malware detection using deep learning. Procedia Computer Science, 184, pp.847-852.				learning models	learning models
4	Wu, Y., Li, M., Zeng, Q., Yang, T., Wang, J., Fang, Z. and Cheng, L., 2023. DroidRL: Feature selection for android malware detection with reinforcement learning.	RL droid agent	DroidRL dataset	Achieved an accuracy of 95.6%	Limited machine/deep learning models	Implement more machine learning models

	Computers & Security, 128, p.103126.					
5	Masum, M., Faruk, M.J.H., Shahriar, H., Qian, K., Lo, D. and Adnan, M.I., 2022, January. Ransomware classification and detection with machine learning algorithms. In 2022 IEEE 12th annual computing and communication	DT, RF, NB, LR, Neural network	Ransomware dataset on kaggle	Random forest achieved an accuracy of 99%	Limited deep learning models	Implement more deep learning models

	workshop and conference (CCWC) (pp. 0316-0322). IEEE.					
6	Shatnawi, A.S., Yassen, Q. and Yateem, A., 2022. An android malware detection approach based on static feature analysis using machine learning algorithms. Procedia Computer Science, 201,	SVM, KNN, NB	CICInvesAndMal2019 dataset	SVM achieved an accuracy of 94.36%	Limited scope of machine learning models	Implement more models

	pp.653-658.					
7	Damaševičius, R., Venčkauskas, A., Toldinas, J. and Grigaliūnas, Š., 2021. Ensemble-based classification using neural networks and machine learning models for windows pe malware detection. Electronics, 10(4), p.485.	KNN, SVM, DT, RF, NN, LDA, LR, SGDC, Boosting techniques and ensemble method	ClaMP dataset	Ensemble had an accuracy of 99%	Lack of explainable AI (XAI)	Implementation of the explanation AI
8	Herrera-Silva, J.A.	NB, NN, RF, GBT	Cuckoo Foundation	Gradient boosting	Does not incorporate	Implement more deep

	and Hernández-Álvarez, M., 2023. Dynamic feature dataset for ransomware detection using machine learning algorithms. Sensors, 23(3), p.1053.			trees achieved an accuracy of 99%	more deep learning models	learning models
9	Xing, X., Jin, X., Elahi, H., Jiang, H. and Wang, G., 2022. A malware detection approach using autoencoder in deep learning.	CNN, SVM, DT, NB, autoencoder	Collected from VirusShare	Autoencoder achieved an accuracy of 96%	Lack of adequate preprocessing and low performance of the CNN model	Improve the accuracy of the CNN and preprocess dataset efficiently

	Ieee Access, 10, pp.25696-25706.					
10	Chua, T.H. and Salam, I., 2022. Evaluation of machine learning algorithms in network-based intrusion detection system. arXiv preprint arXiv:2203.05232.	DT, RF, SVM, NB, ANN	CIC-IDS2017, CSE-CIC-IDS2018 and LUFlow dataset	Decision trees achieved an accuracy of 99.70%	Overfitting in model due to poor feature selection	Improvement on the feature selection method utilized.

2.4. JUSTIFICATION FOR PROPOSED MODELS FOR THE STUDY

The proposed deep learning models for this research, multilayer perceptron, convolutional neural network, recurrent neural network, and long short-term memory, for this detection project were chosen because of their proven effectiveness in the literature review and because of their individual strengths in processing dynamic datasets. As seen in the literature review, neural networks outperformed classical machine learning algorithms in identifying unseen malware classes due to their ability to extract and learn hidden patterns in any data. The convolutional neural networks, in

particular, have demonstrated a strong performance in the review, which makes them suitable for malware types represented in the form of sequences of bytes. RNNs and their advanced variant, LSTM, are effective for capturing sequence patterns, which are common in dynamic malware behaviors. ANN, as a foundational deep learning architecture, offers a flexible framework for learning from structured data and serves as a baseline model for the other three deep learning models proposed. The literature also reflects a trend of improving accuracy and adaptability through deep learning, with models like LSTM achieving detection rates above 99% in several studies. Given the high dimensionality and imbalance present in most of the malware datasets reviewed, these models were selected as a solution to collectively address challenges in the literature review, such as the provision of more deep learning models to combat malware, which was a problem seen in most of the reviewed studies; accuracy improvement on the deep learning models, as some of the deep learning models in the review underperformed in contrast to the classical models; and lastly, the importance of feature extraction on the accuracy of these models will be noted. Also, looking at the papers individually, Akhtar et al. (2022) utilized a collection of classical machine learning models with a single CNN model, which did not perform optimally in the research. Ideally, the deep learning model should have performed better than the classical models; hence, part of the reason why the proposed research involves the use of CNN as one of the deep learning models. Most of the research in the literature review had a limitation that was related to a deficiency of sufficient deep learning algorithms; therefore, therein lies the reason for the proposition of this research, where the deep learning models would solely be developed and experimented on. In terms of accuracy constraints from the literature review, most of the deep learning models failed to achieve an accuracy greater than 95% even in situations where they happened to be the best-performing model. This led to the conclusion of the optimization of deep learning models with respect to accuracy improvement.

RESEARCH METHODOLOGY

In this section, the procedure utilized in the development of the malware detection system using the deep learning algorithms will be fully discussed. This section explains in detail the models implemented, the materials and design specifications, and the challenges encountered while developing the proposed system.

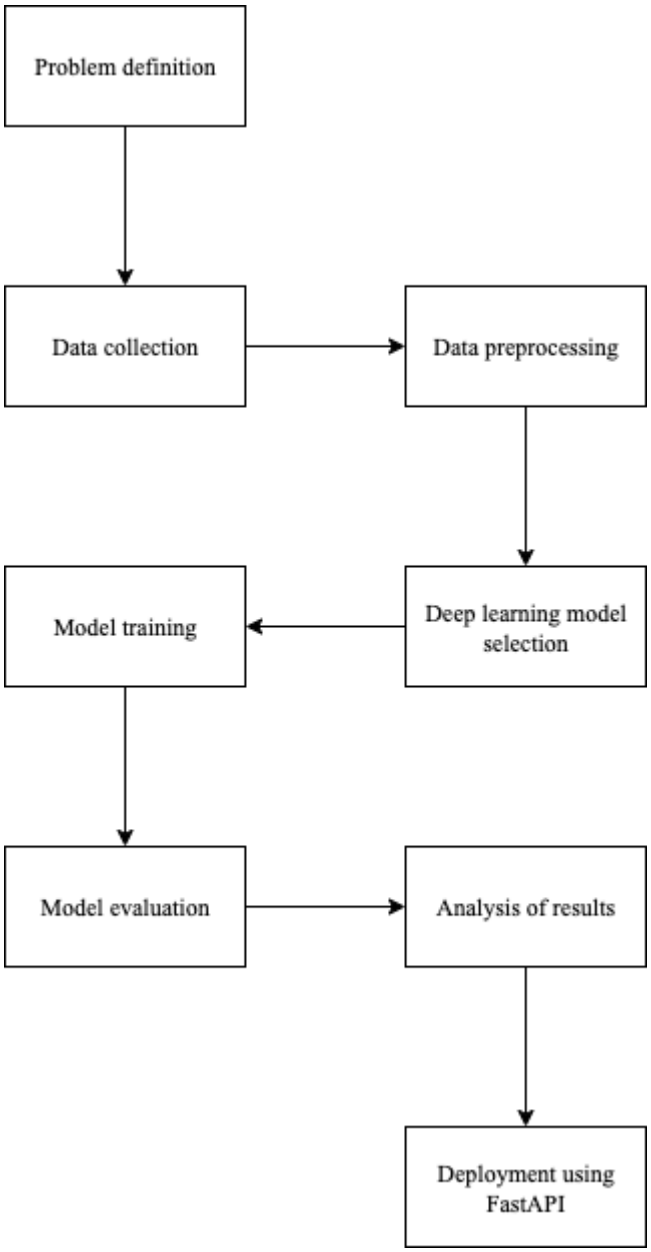

Figure 1: flowchart of design steps

3.1. DESIGN AND IMPLEMENTATION SPECIFICATION

The steps taken in this research to ensure the effectiveness of the result are first the definition of the problem, the data collection process and preprocessing, model training and evaluation, the result analysis, and finally the deployment of the model using FastAPI.

3.1.1. THE DATASET

The Windows malware dataset from Kaggle is a class of malware dataset that comprises the SOMLAP dataset, which is a Windows PE header malware dataset. The idea behind this dataset is that the Windows-based EXE files of malware can be detected through their portable executable file header features. This dataset consists of 51,409 samples that include both benign and malware files with a total of 108 pure portable executable file header attributes. The data contains 19,809 malware file features gathered from VirusShare and 31,600 benign executables gathered from Windows 10 OS. The dataset contains 109 columns.

3.2. MATERIALS AND TOOLS UTILIZED

- Hardware: NVIDIA RTX 4090 for deep learning
- Software and libraries: Python v3.11 as the system programming language, pandas and numpy for data processing, matplotlib and seaborn for visualization, tensorflow for the deep learning implementation, and streamlit for model deployment.

3.2.1. FILE STRUCTURE

- Malware detection ipynb file.
- Saved model weight.
- Saved scaler pickle file.
- Dataset csv.
- FastAPI py file.
- Feature Extractor file.

3.3. DATA PREPROCESSING TECHNIQUES

In this study, data preprocessing was conducted to prepare the dataset for building deep learning models by ensuring it was clean and appropriate. The key steps are outlined below:

- Missing data imputation: The dataset used in this study had no missing values. However, if missing data were present, continuous variables would be filled with mean or median values, while categorical variables would use the mode. If a high percentage of data points were missing, those values would be removed.
- Normalization: The MinMaxScaler was employed during feature selection to normalize the data. Subsequently, continuous variables were scaled using the StandardScaler, which adjusts features to have a consistent mean and standard deviation, typically scaling them to a range of [0, 1].
- Feature selection: Redundant features were eliminated through correlation analysis using the chi-square method to identify and retain the most relevant features.
- Sampling: To address dataset imbalance, where the minority class (e.g., malware samples) was underrepresented, the RandomOverSampler was used. This method duplicates samples from the minority class to match the majority class, ensuring balanced representation for better model learning. SMOTE was avoided as it generates synthetic data, which could negatively impact model training.
- Data splitting: The dataset was divided into training, validation, and testing sets in a 60:20:20 ratio to ensure an unbiased evaluation of the model's performance.

3.4. DEEP LEARNING MODELS

The proposed deep learning models will be properly discussed in this section. The algorithms to be discussed are the multilayer perceptron, convolutional neural network, long short-term network, and recurrent neural network.

3.4.1. MULTILAYER PERCEPTRON (MLP)

The multilayer perceptron (MLP) is an artificial neural network (ANN), which is basically a computational model that was first developed from the inspiration of the biological neural networks as seen in the human brain. This deep learning model operates by automatically adjusting

inbuilt parameters according to relationships in the input and output features; it also permits learning from any data to be possible and is a vital part of any deep learning model, because it can be seen as the basis of the neural network. It is suited for malware detection because of its ability to learn nonlinear relationships in data. In this research, this model was implemented using the TensorFlow framework (Maan, 2025).

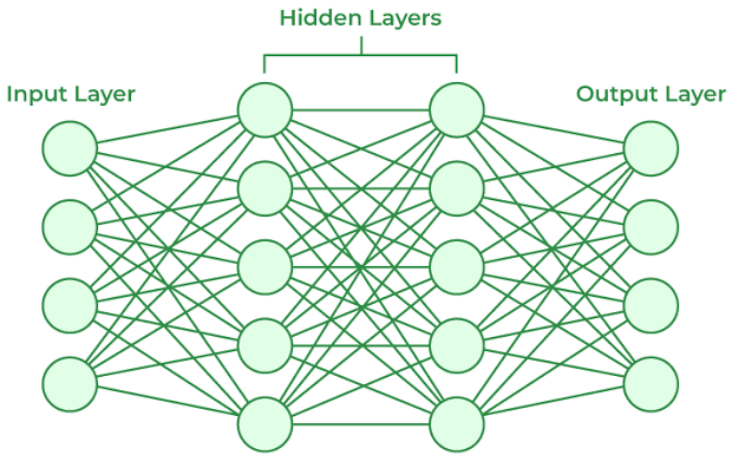

Figure 2: the multilayer perceptron (GeeksforGeeks, 2024)

3.4.2. CONVOLUTIONAL NEURAL NETWORK (CNN)

After the development of the multi-layer perceptron (MLP), the convolutional neural network (CNN), a deep learning model designed to process image and video data in grid format, was developed. The convolution layer, which is an important layer in image processing, is the major difference between a convolutional neural network and a multilayer perceptron. This model is more powerful than a multilayer perceptron because it learns hierarchical patterns in data by using fewer parameters than a regular multilayer perceptron, and it can also detect features regardless of their location in the data. It is usually employed in computer vision tasks but can also be applied to tabular data, such as malware detection.

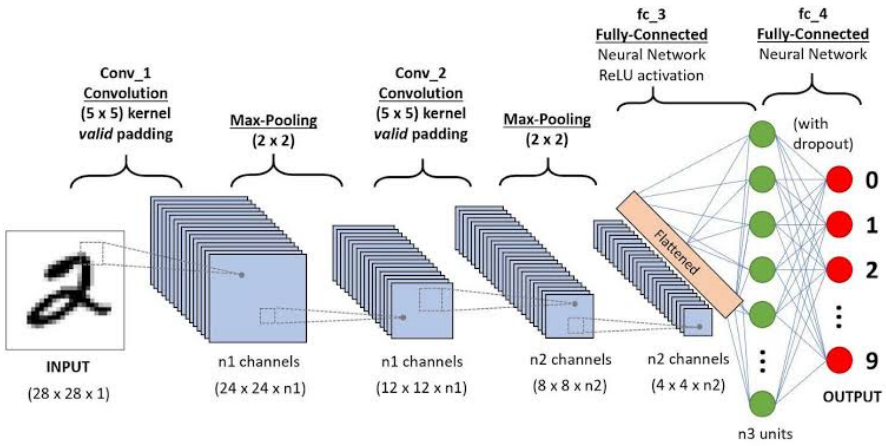

Figure 3: the convolutional neural network (Dandekar, 2023)

3.4.3. RECURRENT NEURAL NETWORK (RNN)

Recurrent neural networks (RNNs) are a type of artificial neural network that make use of latent memory to remember information about previous inputs to interpret sequential data and make predictions (Khan, 2024). This model operates using the principle of feedback connection, where information is enabled to flow in cycles through the network. The recurrent neural network is best suited for tasks such as text processing, speech recognition, and for data that is represented in sequence, such as time series. This model captures temporal dependencies efficiently because each neuron analyzes both the current input and the output from the previous time step. Long-range learning is a challenge for the RNNs' bias, which is a major cause of vanishing or exploding gradients. Therefore, researchers developed variants of the RNN, like the LSTM and GRU, to overcome this limiting factor (Xu, 2022).

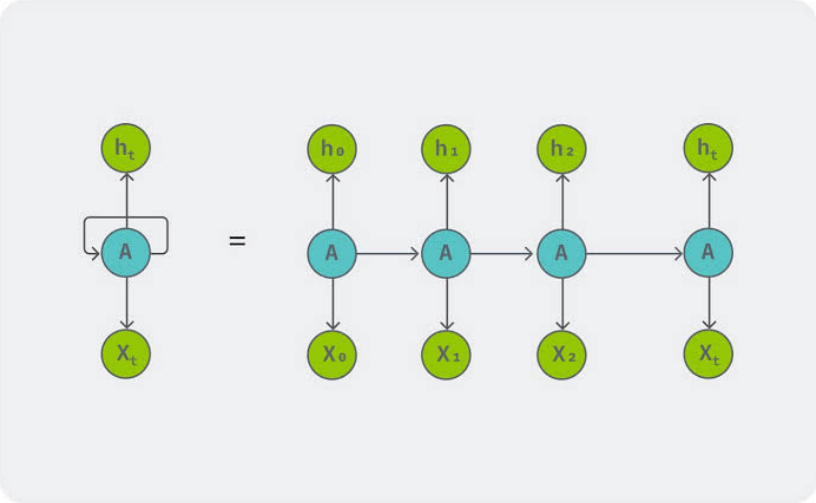

Figure 4: the recurrent neural network (Gadre, 2023)

3.4.4. LONG SHORT-TERM MEMORY (LSTM)

One of the variants of the RNN, the long short-term memory (LSTM), was designed to overcome the limitations of the RNNs, which are vanishing or exploding gradients. These models have a special type of model architecture that uses gating mechanisms and memory cells to control the information flow efficiently. Three gates are present in each LSTM cell, and they are the input gate, which determines how much of the new input should be stored in the cell state; the forget gate, which chooses the information that should be discarded in the cell; and the output gate, which regulates the amount of information transferred from the cell state to the output (Jaiswal, 2022). Together, these gates allow the network to remember crucial information while eliminating any information that is unnecessary. This allows the model to update or maintain information selectively over time. Because of this, the LSTM is highly efficient for long-term dependent learning tasks such as speech recognition, language modeling, and time series.

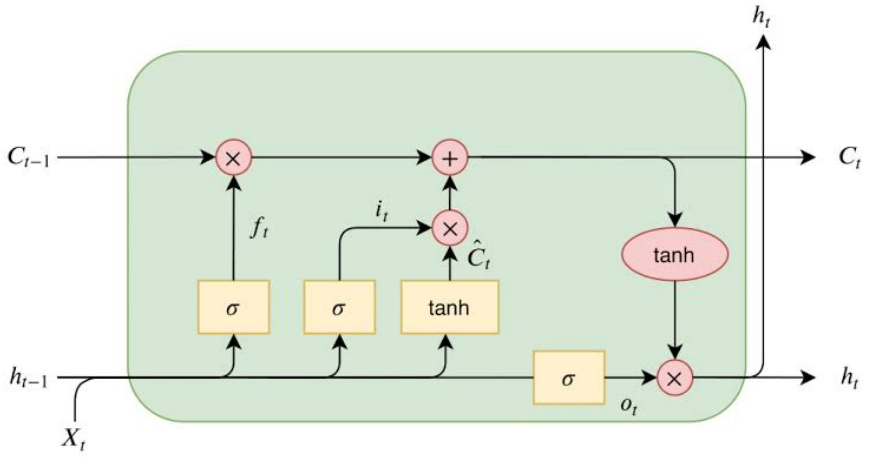

Figure 5: the long short-term memory (Hesaraki, 2023)

3.5. CHALLENGES FACED DURING MODEL IMPLEMENTATION

- In this dissertation, the proposed dataset was from Kaggle, which has a high chance of possessing some statistically standardized figures, which would cause some discrepancies in some vital preprocessing steps such as data standardization as seen in this research.
- As we know, in these kinds of problems, there are usually cases of data imbalance, so utilizing balancing techniques usually introduces some false figures, which may affect the model performance.
- Lack of a robust dataset that shows the real-time malware attacks. Because malware attacks are constantly evolving as time goes on, it is important to use the updated data from firms that we didn't have access to.

3.6. MODEL EVALUATION

In this research, statistical approaches were used to evaluate the performance of the proposed deep learning models after training and testing had been carried out. The performance indicators utilized in this study are accuracy, precision, recall, and F1 score. The confusion matrix is a table that compares the actual labels with the predicted labels produced by the deep learning model, and in this research, it served as the foundation of these assessment metrics. There are four primary parts to the confusion matrix:

- True positive (TP), which represents the cases where the model correctly identifies the positive class.
- True negative (TN) represents the cases where the model correctly identifies the negative class.
- False positive (FP) represents the cases where the model incorrectly identifies the positive class. It is known as a type 1 error.
- False negative (FN) represents the cases where the model incorrectly identifies the negative class. It is known as type 2 error.

The formula for the evaluation metrics derived using these relationships is shown below:

Table 3.1: evaluation metrics

METRICS	FORMULA
Accuracy	$\frac{TP + TN}{TP + FP + TN + FN}$
Precision	$\frac{TP}{TP + FP}$
Recall	$\frac{TP}{TP + FN}$
F1-score	$\frac{2\ x\ Precision\ x\ Recall}{Precision + Recall}$

RESULTS AND DISCUSSION

This chapter highlights and discusses the results obtained after the malware detection model training and classification using the four (4) proposed deep learning neural networks. In this research, the metrics of performance utilized were the accuracy, precision, recall, and F1 score, since they were also the main metrics of performance used in the literature review. The results are shown in table 4.1 below:

Table 4.1: Results for the proposed models

MODEL	ACCURACY	PRECISION	RECALL	F1-SCORE
MULTILAYER PERCEPTRON	96.22%	95.68%	96.52%	96.05%
CONVOLUTIONAL NEURAL NETWORK	96.22%	95.71%	96.47%	96.05%
LSTM	97%	97%	97%	97%
RNN	97%	97%	96%	97%

Table 4.1 shows the result obtained from the comparative analysis of the four (4) deep learning models, focusing on the four important performance metrics: accuracy, precision, recall, and F1-score. From the result table, LSTM and the RNN outperformed the CNN and MLP in terms of all the metrics of performance, with the LSTM being the best-performing model; it beat the RNN only in the recall metrics. In this research, the LSTM was very efficient in detecting and classifying malware in website domains, while the RNN provides a competitive alternative following the LSTM. This highlights the effectiveness of employing sequence models, since malware attacks can be seen in the light of a sequential manner.

4.1. DISCUSSION

In this research, four (4) deep neural networks were proposed for the detection and classification of malware attacks in website domains using a dataset obtained from Kaggle, an online repository for machine learning projects. This discussion section sheds light upon the results obtained from the research conducted, using other existing literature from the review section as a baseline for comparison. The main reason for the selection of the proposed models got its root from the literature review, which showed that most of the research had a limited amount of deep learning neural network implementations. Therefore, this study solves that limitation from the literature review by implementing the proposed algorithms. The research was conducted, and after the training and evaluation, it was observed that the LSTM model outperformed the other deep learning models in the research, also considering the various malware features, which was also a limitation from Alomari et al. (2023).

CONCLUSION AND FUTURE RECOMMENDATIONS

In this comparative research, four (4) deep learning neural networks were proposed for the main objective of malware detection enhancement on website domains using the Windows malware dataset from Kaggle. After the model training and evaluation through a comparative analysis of the four (4) deep learning models, it was observed that the long short-term memory (LSTM) was the most efficient model and can be utilized significantly, as it achieved the best accuracy performance of 97%. This conducted research also made use of this method to show the effectiveness of feature engineering and selection when implementing malware detection, as the selected features contributed to the accuracy achieved. The best-performing model was also deployed using FastAPI, which is a micro framework for model deployment.

In addition, the limitation observed when conducting this research can be linked to the quality of the data utilized, as most of the readily available datasets on the Kaggle platform are synthetically generated; it would require extra tuning to suit a real-world extracted dataset. Therefore, future research should incorporate more robust and quality data with respect to malware detection. Also, more advanced deep learning techniques like the state-of-the-art transformer neural network and the family of large language models can be implemented to achieve better performance.

REFERENCE

- Alaba, F.A. and Rocha, A. (2025) 'Machine learning algorithms on malware detection against smart wearable devices', Studies in Systems, Decision and Control Malware Detection on Smart Wearables Using Machine Learning Algorithms.
- Aljabri, M. and Mirza, S. (2022) 'Phishing attacks detection using machine learning and deep learning models', 2022 7th International Conference on Data Science and Machine Learning Applications (CDMA).
- Barnes-Proby, D. et al. (2018) 'Bridge to opportunities: how one probation agency developed a program designed to connect probationers to high-wage jobs.'
- Bordel, B. et al. (2022) 'Enhancing students' motivation and academic results through international cybersecurity competitions,' ICERI Proceedings ICERI2022 Proceedings.
- Chan, W. (2022) 'Generalizing from small samples: application to the study of deeper learning', AERA 2022.
- Chinnasamy, P. et al. (2022) 'An efficient phishing attack detection using machine learning algorithms', 2022 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC).
- Corbet, S. et al. (2018) 'Cryptocurrencies as a financial asset: a systematic analysis,' International Review of Financial Analysis.
- Dandekar, I. (2023) 'Introduction to Convolutional Neural Networks: Part 1,' *Medium*, 13 February.
- Desfiandi, A. and Soewito, B. (2023) 'Student graduation time prediction using logistic regression, decision tree, support vector machine, and adaboost ensemble learning', IJISCS (International Journal of Information System and Computer Science).
- Dhanavanthini, P. and Chakkravarthy, S.S. (2023) 'Phish-armour: phishing detection using deep recurrent neural networks', Soft Computing - A Fusion of Foundations, Methodologies and Applications.
- Dhiraj and Jain, D.K. (2019) 'An evaluation of deep learning based object detection strategies for threat object detection in baggage security imagery', Pattern Recognition Letters.

- Faisal, M. and Islam, M.S. (2023) 'Improving network security with intrusion detection systems utilizing machine learning and deep learning techniques', 2023 26th International Conference on Computer and Information Technology (ICCIT).
- Ferdaws, R. and Majd, N.E. (2024) 'Phishing url detection using machine learning and deep learning', 2024 IEEE World AI IoT Congress (AIIoT).
- Gadre, V. (2023) 'Recurrent Neural Networks: A Beginner's Guide - Vijay Gadre - Medium,' *Medium*, 4 October.
- Gary, T. (2025) 'Graduate assistantship pay often falls short of a living wage', Physics Today.
- GeeksforGeeks. (2024, August 7). *Artificial Neural Networks and its Applications*. GeeksforGeeks.
- Gregório, J. et al. (2024) 'Deep convolutional neural network and character level embedding for dga detection', Proceedings of the 26th International Conference on Enterprise Information Systems.
- Gupta, V. (2021) 'Requirements engineering challenges for social sector software development: insights from multiple case studies', Requirements Engineering for Social Sector Software Applications.
- Hesaraki, S. (2023) 'Long Short-Term Memory (LSTM) - Saba Hesaraki - Medium,' *Medium*, 27 October.
- Hussein, N. (2023) 'Eye-tracking in association with phishing cyber attacks: a comprehensive literature review', Computer Networks & Communications.
- Indrianti, Y. and Sasmoko (2024) 'Comprehensive self assessment through beebest mobile application to improve performance', 2024 IEEE 9th International Conference for Convergence in Technology (I2CT).
- Jain, M. and Srihari, A. (2022) 'Comparison of machine learning models for stress detection from sensor data using long short-term memory (lstm) networks and convolutional neural networks (cnns)', International Journal of Scientific Research and Management.
- Jaiswal, R. et al. (2022) 'Tsdeeplearning: deep learning model for time series forecasting', CRAN: Contributed Packages.
- Jha, B., Atre, M. and Rao, A. (2022) 'Detecting cloud-based phishing attacks by combining deep learning models', 2022 IEEE 4th International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications (TPS-ISA).

- Jo, Y. and Jung, K. (2023) 'Comparative study of machine learning and deep learning models applied to data preprocessing methods for dam inflow prediction', GEO DATA.
- K, R. and K, M.S. (2022) 'Breast cancer prediction by leveraging machine learning and deep learning techniques with different imaging modalities', 2022 IEEE 7th International conference for Convergence in Technology (I2CT).
- Kalamkar, D. et al. (2020) 'Optimizing deep learning recommender systems training on cpu cluster architectures', SC20: International Conference for High Performance Computing, Networking, Storage and Analysis.
- Khan, A.F. (2024) 'Recurrent neural networks (rnns)'.
- Klein, J. et al. (2024) 'Synthetic data at scale: a development model to efficiently leverage machine learning in agriculture,' Frontiers in Plant Science.
- Kuravatti, D.P. et al. (2025) 'Quantum computing base cybersecurity mathematical model development for geographically underdeveloped areas using multiple zonal approaches using aiml techniques for stoppage of different types of attacks', Journal of Information Systems Engineering & Management.
- Lu, J. (2023) 'The investigation of malware detection model construction based on deep learning algorithms', Proceedings of the 1st International Conference on Data Analysis and Machine Learning.
- Lysenko, S. et al. (2020) 'Design and development of an intellectual agent for detection of cyber threats and malware in corporate networks,' Herald of Khmelnytskyi National University. Technical sciences.
- Maan, P. et al. (2025) 'An empirical framework for automatic identification of video game development problems using multilayer perceptron', Proceedings of the 20th International Conference on Evaluation of Novel Approaches to Software Engineering.
- Martinelli, F., Mercaldo, F. and Santone, A. (2024) 'Evaluating the impact of generative adversarial networks in android malware detection', Proceedings of the 19th International Conference on Evaluation of Novel Approaches to Software Engineering.
- Martirosyan, K.V. and Chernyshev, A.B. (2023) 'The complex control system functionality development for the sophisticated object state', 2023 V International Conference on Control in Technical Systems (CTS).

- Mogal, A.K. and Dubey, V.R. (2024) 'Image classification techniques leveraging support vector machines, decision trees and neural networks', Image Processing Techniques and its Applications in Computer Vision and Artificial Intelligence.
- Moubarak, J. and Feghali, T. (2020) 'Comparing machine learning techniques for malware detection', Proceedings of the 6th International Conference on Information Systems Security and Privacy.
- Murphy, S. et al. (2022) 'Combinatorial evaluation of physical feature engineering, classical machine learning, and deep learning models for synchrophasor data at scale'.
- Niu, W. et al. (2024) 'Ai-based android malware detection methods', Android Malware Detection and Adversarial Methods.
- Oak, R. et al. (2019) 'Malware detection on highly imbalanced data through sequence modeling', Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security.
- Panda, B.K. et al. (2024) 'The battle against diphtheria: nigeria's ongoing struggle', Anti-Infective Agents.
- Rane, N.L., Kaya, Ö. and Rane, J. (2024) 'Artificial intelligence, machine learning, and deep learning for sustainable industry 5.0', Artificial Intelligence, Machine Learning, and Deep Learning for Sustainable Industry 5.0.
- Reza, K., Islam, M. and Estivill-Castro, V. (2019) 'Privacy preservation of social network users against attribute inference attacks via malicious data mining,' Proceedings of the 5th International Conference on Information Systems Security and Privacy.
- Schrynemeeckers, R. (2015) 'Optimizing lateral placement and production while minimizing completion costs using downhole geochemical logging.'
- Sharma, A., Gupta, A.K. and Shabaz, M. (2022) 'Categorizing threat types and cyber-assaults over Internet of Things-equipped gadgets,' *Paladyn Journal of Behavioral Robotics*, 13(1), pp. 84–98.
- Shin, H. et al. (2016) 'Deep convolutional neural networks for computer-aided detection: cnn architectures, dataset characteristics and transfer learning', IEEE Transactions on Medical Imaging.
- Soldatos, J., Lazaro, O. and Cavadini, F. (2019) 'The digital shopfloor: industrial automation in the industry 4.0 era', The Digital Shopfloor: Industrial Automation in the Industry 4.0 Era.
- Soni, A. (2021) 'Has my email been hacked?', The Cybersecurity Self-Help Guide.

- Soni, T. (2024) 'Robust android security: a random forest-based approach to malware detection', 2024 4th International Conference on Sustainable Expert Systems (ICSES).
- Sullivant, J. (2016) 'The evolving threat environment', Building a Corporate Culture of Security.
- Taherian, H. et al. (2022) 'One model to enhance them all: array geometry agnostic multi-channel personalized speech enhancement', ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).
- Talbi, A. et al. (2022) 'Feature importance and deep learning for android malware detection', Proceedings of the 8th International Conference on Information Systems Security and Privacy.
- Thiruvoth, A. and Ogale, P. (2025) 'A phishing detection system for enhanced cybersecurity using machine learning', Proceedings of the 20th International Conference on Software Technologies.
- Xu, T. (2022) 'Analysis on the applicability of rnn, lstm, and gru deep learning algorithms for stock price prediction', Proceedings of the International Conference on Big Data Economy and Digital Management.
- Zhang, Z. and Das, C. (2023) 'Pathogen-defending deubiquitinase possesses distinct specificity towards k6-polyubiquitination', Trends in Microbiology.