Iraqi Journal of Information and Communications Technology(IJICT)
Vol. 8, Issue 2, August 2025
www.ijict.edu.iq
ISSN:2222-758X
e-ISSN:2789-7362

IMAGE-BASED IOT MALWARE DETECTION USING CHI-SQUARE AND CNN

Mariam H. Al-Musawi [1], **Ban Mohammed Khammas** [2], **Stephen Bassi Joseph** [3]
[1] Department of Computer Networks Engineering, College of Information Engineering, Al-Nahrain University, Jadriya, Baghdad, Iraq
[2] Department of Cyber Security Engineering, College of Information Engineering, Al-Nahrain University, Jadriya, Baghdad, Iraq
[3] Department of Computer Engineering, University of Maiduguri, Nigeria
mariam.hazim@coie-nahrain.edu.iq[1], banm1979@nahrainuniv.edu.iq[2], sjbassi74@gmail.com[3]
Corresponding Author: **Ban Mohammed Khammas**
Received:06/11/2024; Revised:04/03/2025; Accepted:21/04/2025
DOI:10.31987/ijict.8.2.307

Abstract*- The Internet of Things (IoT) constitutes an expanding network of interconnected gadgets that enable intelligent systems to gather, analyze, and disseminate data. However, this rapid growth raises cyber-attack risks due to poor configurations and outdated systems. Malware, which exploits system vulnerabilities, represents a significant threat to the information security of IoT systems. Thus, malware detection in IoT systems is a critical concern. Therefore, this research paper presents an IoT malware detection method based on an image dataset and the Chi-square method as well as applying the Convolutional Neural Network (CNN) deep learning model to detect the IoT malware. This study attempts to investigate the impact of the chi-square Feature Selection (FS) method on the effectiveness of CNNs for identifying IoT malware, by directly applying feature selection to the images to discern the most informative ones from the dataset before passing them to the CNN deep learning model, demonstrating robust outcomes and validating the efficacy and robustness of the suggested approach for identifying IoT malware. An experimental comparison was carried out between the suggested method that Involved training the CNN on the feature-selected dataset (FS+CNN Model) and the (CNN Model) that was trained on the full dataset and was also evaluated using the presented state-of-the-art to add to the method's reliability. The accuracy of the (Fs + CNN Model) reached 98.19% while its precision, recall, and F1-score were 99.52%, 95.90 %, and 97.68 %, respectively, outperforming the CNN Model's accuracy with 94.75 %, precision with 93.00 %, recall with 91.43 % and f1-score with 90.43 %. It also outperformed the state-of-the-art evaluation with an accuracy value of 97.93 %, a precision value of 98.64 %, a recall value of 88.73 %, and an f1-score value of 93.94 %.

keywords: **Internet of Things (IoT), Convolutional Neural Networks (CNN), Deep Learning, Chi-square, Feature Selection.**

I. INTRODUCTION

The Internet of Things (IoT) is a continually expanding network of interconnected gadgets that allow intelligent systems and services to identify, gather, disseminate, and analyze data. The proliferation of linked devices in IoT systems is accelerating, becoming an essential technology for diverse applications [1]. This increasing prevalence of IoT devices across diverse environments could raise cyber-attack risks owing to their poor configurations, such as weak passwords and outdated systems. Vulnerabilities in the TCP/IP stack, such as Ripple-20 and Urgent-11, have harmed millions of IoT devices, enabling attackers to access their critical capabilities. Ensuring the security of these systems is essential because of their inherent characteristics and extensive uses. Countermeasures must account for processing power and data storage capacity, rendering host-based security applications and intricate cryptographic solutions impractical because of their computational cost [2]. Meanwhile, malware represents a crucial risk to the information security of IoT systems,

ISSN:2222-758X
e-ISSN:2789-7362

since it is deliberately created to disrupt, damage, or acquire illicit access to computer systems, IoT networks, and devices. IoT malware encompasses several categories including botnets, ransomware, and spyware. The unique characteristics of IoT systems, such as resource constraints, component variability, and the lack of defined security procedures, make them vulnerable to malware attacks. Thus, malware detection methods are a critical concern [3].

Static and Dynamic analysis methods are the most common methods used for malware detection [4]. In the static method, malware features are derived from the malware's physical composition, including the binary format of the malware file, while the dynamic analysis performed during the execution of malware emphasizes the extraction of behavioral features. Both techniques utilize the extracted features to provide reliable solutions for identifying malicious software [5, 6]. Currently, the adoption of CNN-based deep learning techniques is made practicable due to its ability to attain elevated degrees of accuracy and precision in identifying malware using the binary or multi-channel images produced by transforming malware-infected machine code files (.byte) using the vision approaches [7]. Therefore, this study investigates the effectiveness of the Chi-Square Feature Selection (FS) method [8] on the model performance as well as the implementation of the Convolutional Neural Network (CNN) deep learning model in IoT malware detection utilizing the malware's visual representation, using the benchmark Kaggle IOT-Malware images dataset by applying it on the images directly to select the most informative images from the dataset, even though it is relatively uncommon strategy. The contributions that are mainly presented by this study are listed as follows:

- The CNN model utilization performs efficiently for identifying malware in the IoT environment.
- Implementing the chi-square test to identify the best images and reduce the dataset size, hence improving the deep CNN model's classification performance as well as minimizing the detection time of the model.
- Conducting a comparative analysis of the performance of the CNN deep learning model before and after the application of the Chi-Square FS method, to evaluate its impact, while also contrasting its performance with the latest relevant publication.

The rest of the paper includes a brief review of the most relevant studies in IoT Malware detection, which is presented in section II, while section III presents the proposed methodology; finally, section IV presents the obtained results and the discussion, meanwhile, the conclusion is presented in section V.

II. LITERATURE REVIEW

Numerous research studies have suggested novel methodologies for IoT malware detection that apply the CNN deep learning model to the visual representation of malware files; the following section offers a succinct overview of the most significant research studies.

In [9] introduced an IoT malware detection framework called iMAD, which used a newly developed deep CNN architecture to analyze various texture patterns by examining edges and improving features to distinguish malware from benign in IoT networks. The model was assessed using the Kaggle IOT-Malware images dataset and demonstrated exceptional detection accuracy, surpassing other robust pre-trained CNN architectures when evaluating their results. The study shows the capability of modifying the CNN architecture to intensify the feature learning process, although, the feature selection effect at the

ISSN:2222-758X
e-ISSN:2789-7362

input level continues to be uncertain.

Another research has also introduced a newly updated CNN architecture combined with ensemble classifiers within a framework termed (DSBL), as given by [10]. The DSBL framework was assessed using the IOT-Malware dataset, and its efficacy has been compared with many deep CNN models, including VGG, InceptionV3, GoogleNet, and ResNet, to illustrate its superior performance. While the developed CNN architecture focused on creating more varying feature maps throughout capturing structured and unstructured feature patterns within the CNN architecture by utilizing dilated convolutional layers with multi-paths accompanying boundary and regional processing, it did not explore selecting features before passing the data to the model.

On the other hand, in [11] introduced a more lightweight CNN (LCNN) which incorporates two key operations, depthwise convolution and channel shuffle, for the categorization of IoT malware images. The proposed technique involved transforming malware binaries into multidimensional Markov pictures for further analysis using the developed LCNN to identify the malware. The system demonstrated superior accuracy and a reduction in trainable parameters compared to other deep learning models, including VGG16.

In terms of feature extraction and selection from the image datasets, in [12], Introduced an innovative framework for detecting IoT malware that employs the Discrete Wavelet Transform (DWT) to decompose images into various coefficients, which are subsequently fused by implementing a Generative Adversarial Network (GAN) to reconstruct the images from the extracted coefficients; thereafter, to detect the IoT malware, a lightweight CNN has been utilized. Two benchmark datasets, the IoT malware images dataset, and the Malimg images dataset, are used to evaluate the effectiveness of the suggested framework. Both datasets show better performance than the state-of-the-art algorithms. In [13] also extracted the features from the images utilizing the Double-Density Discrete Wavelet Transform (D3WT) and a hybrid model combining CNN and Long Short-Term Memory (LSTM) to identify and classify IoT malware. The proposed technique is assessed using three datasets: IoT malware, Microsoft BIG-2015, and Malimg, demonstrating very high accuracy for each dataset.

Additionally, regarding the application of the Chi-Square FS approach in malware detection, in [14], suggested a one-dimensional CNN architecture for network anomaly detection in cybersecurity. The suggested approach categorized the network traffic data into several segments and treated each part separately. Before model training, FS is performed using the Chi-square approach to exclude less important features from the dataset, hence improving detection accuracy. Even though the study employed the chi-square feature selection method, it only focused on text-based network traffic data and one-dimensional CNN not considering its relevance to image-based malware detection.

Another study has also employed the chi-square FS to select the most relevant features from textual traffic data with the goal of intrusion detection in IoT networks presented by [15]. The presented methodology involved using a hybrid CNN-LSTM model trained on the selected features after applying the Synthetic Minority Over-sampling Technique (SMOTE) to ensure a balanced dataset. The CNN-LSTM model performed well surpassing other experimental models demonstrating its efficiency.

Conversely, by using the ensemble FS approach and deep learning models, in [16] presented a methodology for identifying IoT network attacks. This approach is built on the benefits of several filter-based FS techniques, including variance threshold,

ISSN:2222-758X
e-ISSN:2789-7362

mutual information, Chi-square, ANOVA, and L1-based methods. To increase the effectiveness of the employed deep learning models (CNN, RNN, GRU, and LSTM), the chosen features were combined to form an ensemble method. All models attained the maximum detection accuracy, ranging from 97.05% to 97.87%.

Meanwhile, in [17] the chi-square method was employed, resulting in the selection of twenty significant features as inputs for the CNN model. To enhance CNN's performance and find the most optimal hyperparameters for the method, random and grid searches were employed. The experiments demonstrate that CNN surpasses other machine learning models in accuracy and precision, particularly due to the enhancements provided by the Chi-Square method, which facilitated the selection of only the significant features in the CNN algorithm's performance.

None of the prior studies have utilized the chi-square FS method for direct image selection, as this approach is relatively uncommon despite all employing CNNs for IoT malware identification. Consequently, this study aims to explore how the chi-square feature selection method affects CNN ability to detect IoT malware through its visual representation, applying FS directly to the images to identify the most informative ones from the dataset. The aforementioned studies were assessed based on accuracy and precision. This research utilizes the publicly available IOT-Malware dataset from Kaggle, with f1-score, recall, accuracy, and precision as the chosen performance evaluation metrics.

III. METHODOLOGY

The underlying structure of the suggested methodology is illustrated in Fig. 1. To assess the impact of the Chi-square FS method, two identical CNN models were trained independently: one on the full dataset (CNN Model) and the other on the feature-selected dataset (FS + CNN Model). The model that performed the best was selected to perform the malware detection on the presented method.

The process included two main steps:

1) Dataset Pre-processing
2) CNNs deep learning model training and evaluation to decide the best for IoT preprocessed dataset to be prepared for use in further steps

Following the dataset preprocessing, the next step is to employ the best method for training the model with the preprocessed dataset. During the model training phase, two distinct methodologies have been examined: the first using the entire dataset within the CNN model and the second employing the selected dataset obtained by Chi-square FS which represents as mentioned before the FS + CNN Model. It should be noted that the CNN model utilized in both methods had the same number of layers and the same set of parameters to perform a fair comparison.

For the application of the FS, a comprehensive investigation was conducted using the Chi-Square FS method to identify the feature images that significantly affect the decisions of the deep learning model. The efficacy of the CNN model is influenced by the dimensions of the training and testing data. The CNN model requires fine-tuning for a variably sized dataset. The FS approach was used to reduce data dimensionality and training time for CNN, enhancing accuracy. The method resulted in less complexity and improved performance. FS in CNN expedites data processing, enabling efficient analysis on low-power devices in the IoT context.

ISSN:2222-758X
e-ISSN:2789-7362

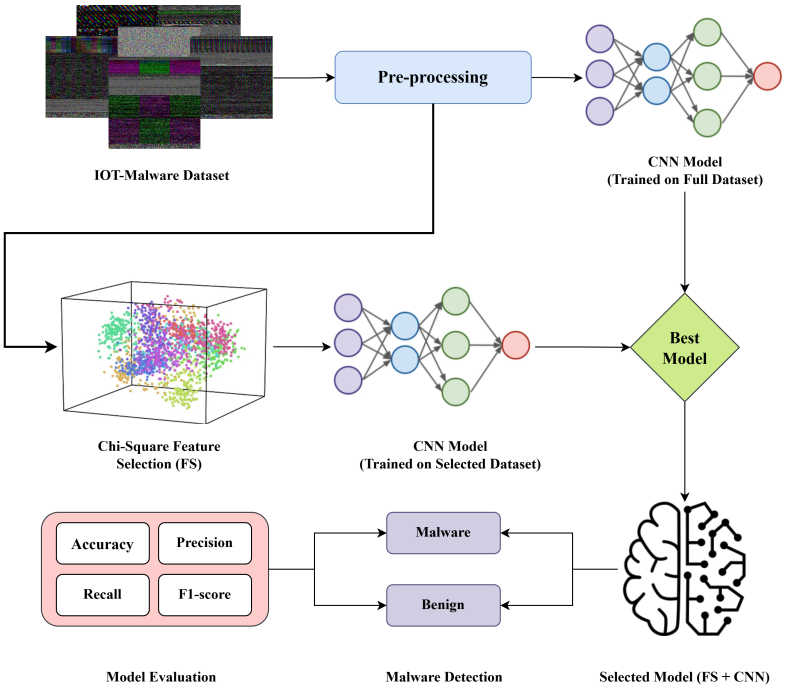


Figure 1: The proposed methodology representation.

A. *Dataset Description*

The utilized dataset is the publicly accessible IOT-Malware dataset from Kaggle [18], which includes two classifications: The benign dataset comprises 2,486 benign images, whereas the malware dataset consists of 14,733 malicious images, with each image measuring 255×255 pixels. Fig. 2 presents a visual depiction of the employed IoT-Malware dataset.

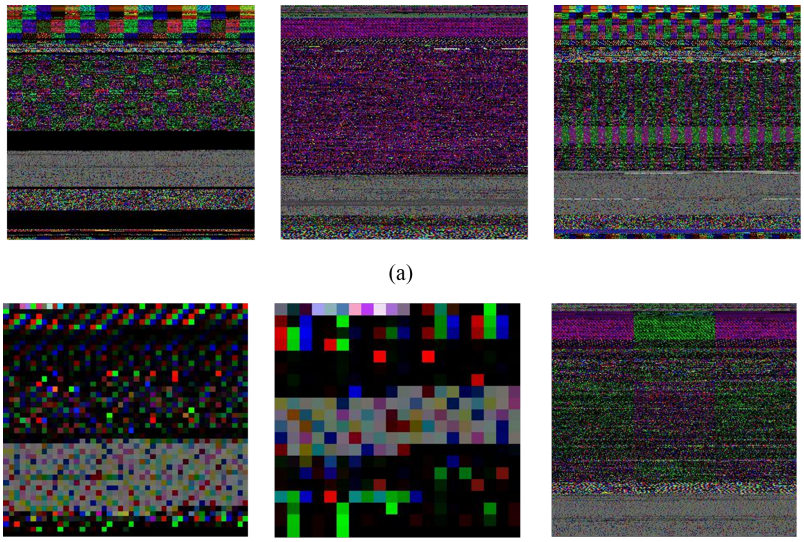
B. *Dataset Pre-processing*

The initial stage in preparing the data for later stages is the data pre-processing. The format of the data is converted so that it can be effectively processed by the deep learning models, which makes it crucial for enhancing the ability to generalize the CNN model. The pre-processing phase involved resizing images to a standard input dimension of 244×244 pixels and normalizing the values of the pixels within a 0 to 1 range to improve the convergence of the model during training. Throughout this phase, data augmentation was employed to address the dataset imbalance issue and enhance its robustness. Image transformation (from grayscale to RGB and back), rotation (0 – 360) degrees, scaling (0.5 – 1), shearing (-0.5, +0.5), and reflection (left and right) were all part of this procedure.

C. *Convolutional Neural Network (CNN) Model*

This research study has employed the CNN algorithm to implement malware detection as it has been proven to be the most effective option for malware image identification and classification [19, 20]. The CNN model consists of three

ISSN:2222-758X
e-ISSN:2789-7362



(a)

(b)

Figure 2: Visual representation of the IOT-Malware Dataset (a) Benignware samples. (b) Malware samples.

essential layers: the convolutional layer, the max-pooling layer, and the fully connected layer. The convolutional layer processes the input sample by analyzing its attributes and performing a dot product between two matrices. Parameters like filter size, stride, and zero padding optimize the results of the convolutional kernels. The ReLU function Eq. (1) is employed to enhance non-linearity in the feature map.

$$\text{ReLU}(f(y)) = \max(0, y) \tag{1}$$

The max-pooling layer selects the maximum values from the region to create a pooling matrix, lowering the feature vector dimensionality and minimizing parameters by down-sampling the input size. The fully connected layer consolidates all characteristics into singular feature vectors, facilitating feature translation from input to output features and functioning as a classifier to evaluate the outputs from the convolutional and pooling layers in sequence.

The CNN deep learning model proposed in this study comprises a sequence of four convolutional layers and max-pooling layers, succeeded by two fully connected layers. The core architecture is illustrated in Fig. 3. The input dataset consists of 3 channel images with dimensions of 244×244 pixels. The model employs a 32 convolution filter, 64 convolutional filters, 128 filters, and a ReLU activation function. The data is flattened into a 1D vector and passed to the next two fully connected layers. The last layer utilizes only one neuron at the output layer, employing a sigmoid activation function in the second layer to perform binary classification and IoT malware detection.

D. Feature Selection Method

Feature selection (FS) is crucial for deep learning models to enhance detection accuracy by extracting prominent features, minimizing dimensions and classification time, and removing features lacking important information, thereby enhancing

ISSN:2222-758X
e-ISSN:2789-7362

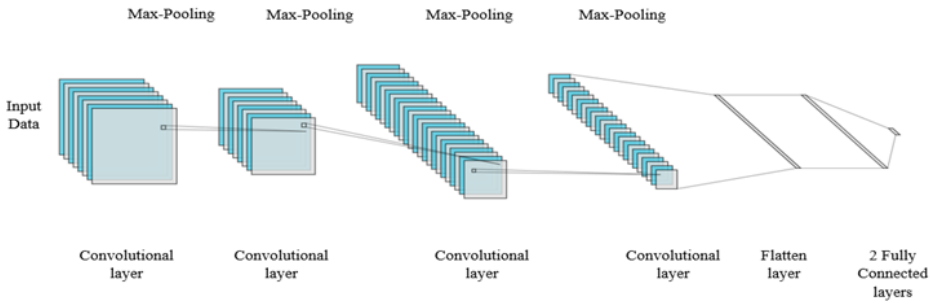


Figure 3: The architecture of the CNN deep learning model.

detection performance [21-22].

The presented study utilized the Chi-square feature selection approach to identify the most informative images characterized by great detail as well as variations in the intensity of pixels (texture, edges, etc.). The Chi-square test evaluates the match between actual data (theoretical expectation) and predicted data distributions. Regarding the selection of image features, a histogram is computed for each image separately, following its conversion to grayscale for simplified analysis. The histogram visually represents the distribution of pixel intensity values (0-255) and highlights discernible patterns, which represent here the predicted data [23]. Meanwhile, the uniform histogram illustrates the actual data, wherein each value in the dataset appears approximately an equal number of times, hence offering a clear representation of data distribution over the value spectrum. The chi-square test quantifies the deviation between the actual and the predicted data via Eq. (2):

$$X^2 = \sum_{i=0}^{k-1} \frac{(O_i - E_i)^2}{E_i} \tag{2}$$

Where, X^2 denotes the chi-square test value, O_i signifies the observed frequency for the intensities of the pixels in each image (as indicated by the image histogram), and E_i indicates the expected frequency for the intensities of the pixels in each image (as represented by the uniform histogram), with k representing the entire number of images in the dataset.

The higher chi-square value signifies a greater dependence of the image on the theoretical expectation, reflecting a smaller discrepancy and showing more distinctive information relative to its uniform distribution. Therefore, in this study, the highest 3959 images ranked by chi-square were chosen from the total images in the dataset.

IV. EXPERIMENT AND RESULTS

A. *Implementation details*

The dataset was separated by a 70:30% ratio for the training and testing to assess the CNN deep learning model. The test set was subsequently divided in an 80:20% ratio to enable the cross-validation implementation, assess the model's resilience during training, and perform the parameters optimization to enhance its performance. The experimental settings included a 64-batch batch size, a 0.001 learning rate, and 10 epochs. The Adam optimizer was used for training. The Kaggle platform

[24] had 16 GB RAM and an Nvidia Tesla P100 GPU. The Keras software package [25] with a TensorFlow [26] backend was used for coding, training, and assessment

B. Performance Evaluation Metrics

The Confusion Matrix has been employed in addition to the detection time of the CNN deep learning model to assess the performance as well as the effectiveness of the suggested IoT Malware detection approach. The Confusion Matrix is an efficient tool for quantifying accuracy, precision, sensitivity or (recall), and f1-score, utilized to evaluate the classifier's performance, and it is a cross table that enumerates the instances which were accurately predicted or classified (cells on the principal diagonal) vs those which were inaccurately predicted or classified (off-diagonal cells) [27, 3].

Table I, together with Eqs. (3-6), defines the methodology employed to compute its parameters. In the context of IoT malware identification and classification, a value of 1 signifies malware, whereas 0 denotes benign; hence, these values are categorized as True Positive (TP): If both the predicted and actual numbers are 1, it indicates that the malicious sample was identified and the prediction was accurate, True Negative (TN): Both the expected and actual numbers are zero, False Positive (FP): A value that is actually 0 but is anticipated to be 1, and False Negative (FN): The actual value represented by 1 while the predicted value represented by 0.

TABLE I
THE CONFUSION MATRIX

Predicted Values	**The Actual Data Values**	
	TF	FP
	FN	TN

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

$$Sensitivity = Recall = \frac{TP}{TP + FN} \tag{5}$$

$$F1\text{-}score = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \tag{6}$$

C. Results and Discussion

This work conducted a full empirical evaluation to assess the effectiveness and the efficacy of the presented work and examine the impact of the chi-square concerning previously published solutions. The proposed approach was trained and assessed using the benchmark Kaggle IOT-Malware dataset. Precision rate and detection rate (recall) are the primitive metrics used to evaluate the malware detection method's efficacy. The first step in securing and controlling the spread is accurately identifying malware that has been infused into a system. If the suggested detection technique's precision is only increased, there could be a rise in false alarms. Lowering the false alarm could result in a lower detection rate. The suggested model took advantage of the disparity by equating the F1-score, the harmonic mean of the two parameters, in

consideration of this intuition, along with the significance of accuracy, which shows how precisely the model can generate predictions.

As previously discussed in section III, the IoT malware detection was carried out using two different methods: the first one relied on the entire dataset (CNN Model), while the other one employed a smaller dataset obtained by the FS technique using the Chi-square (FS+CNN) as only 3959 images were selected from the total dataset, henceforth abbreviated to remember that the CNN model utilized in both methods had the same number of layers and the same set of parameters. Table II and Fig.4 show the study's results in regard to the accuracy, precision, recall, and f1-score metrics, showing that the entire dataset achieves an accuracy of 94.75% on the (CNN Model), whereas the selected dataset's accuracy on the (FS + CNN Model), which employed the same parameters, was 98.19%, drawing the conclusion that the accuracy increased by 3.44% as a result of the FS applying the chi-square approach on the images. Furthermore, Fig. 4 proves that the (FS + CNN Model) outperformed the (CNN Model) that was trained on the entire dataset regarding the precision, recall, and f1-score, increasing by 99.52%, 95.90%, and 97.68%, respectively. The accuracy and loss convergence patterns for both methods,

TABLE II
THE PROPOSED METHOD PERFORMANCE

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
CNN	94.75	93.00	91.43	90.43
FS + CNN	98.19	99.52	95.90	97.68

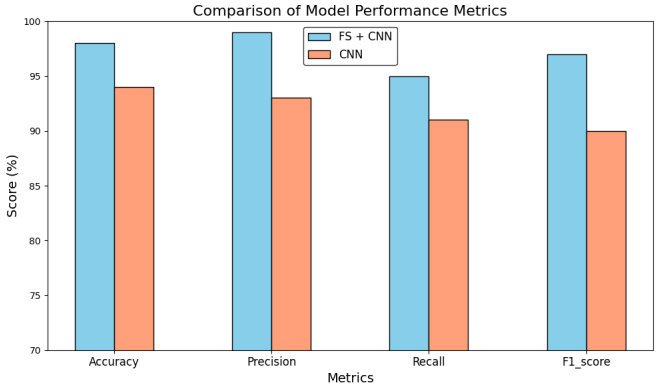
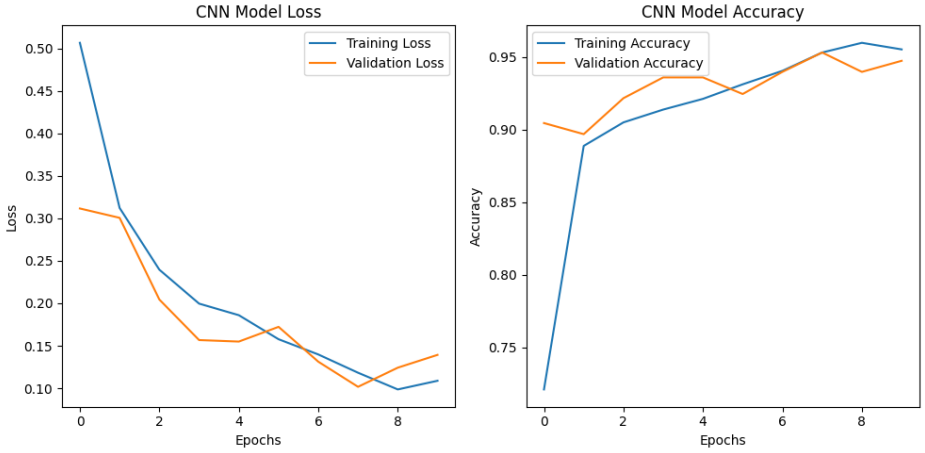


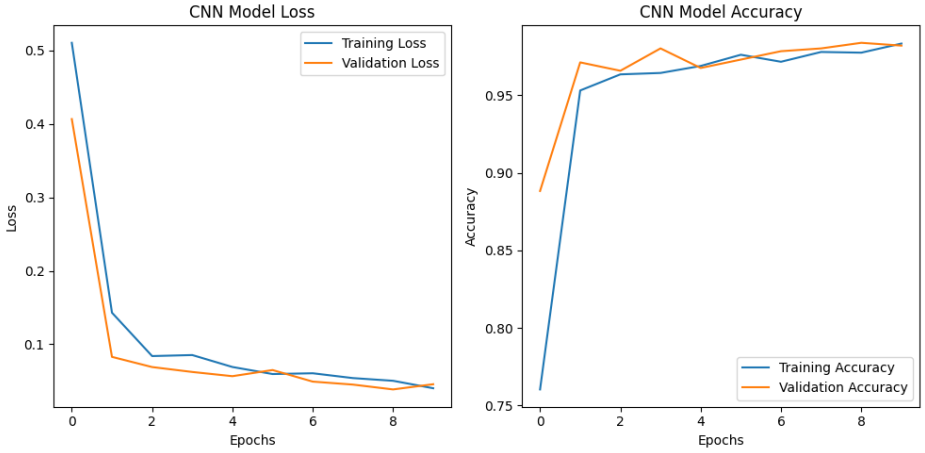
Figure 4: Performance comparison of the suggested method.

pertaining to training and validation data as a function of epochs, are illustrated in Fig. 5. The model loss drops from 0.1392 to 0.0456 and the detection time is reduced by half to 3.5838, after applying the FS method. Table III. indicates that the model was showing overfitting when trained on the complete dataset because it showed better training accuracy compared with testing accuracy, meaning that the model performed poorly by learning noise and certain patterns that generalize badly on new data. In order to minimize the model loss and error values for both approaches, the best number of epochs was determined to be 10. Table III also shows that the CNN model with the selected dataset performed the best. In addition, the FS method assisted in lowering the model's complexity by removing the images with less informative features, which

in turn decreased the size of the dataset and increased the model's efficacy by both enhancing the total performance of the presented model and lowering its computational cost. Meanwhile, it demonstrates competitive performance relative



(a)



(b)

Figure 5: Model loss and accuracy: (a) full dataset (b) after applying FS.

TABLE III
THE TRAINING AND TESTING MEASURES FOR THE PROPOSED METHOD

Model	**CNN**	**FS + CNN**
Epochs	10	10
Training Accuracy (%)	95.37	98.47
Training Loss	0.1103	0.0376
Testing Accuracy (%)	94.75	98.19
Testing Loss	0.1392	0.0456
Detection Time (s)	6.6828	3.5838

to the DSBEL framework, achieving slightly less accuracy yet significantly higher precision, suggesting that the proposed method generates fewer false positives than the DSBL, therefore demonstrating its strength, not to mention that it shows practically the same values in terms of recall and f1-score metrics.

As mentioned before, this study offers a comparison analysis against the current state-of-the-art methodologies utilizing the same dataset, represented by the iMAD [9] and DSBEL [10] frameworks, hence enhancing its reliability and validating its robust conclusions. Table IV presents the comparison of its efficacy with the previous work studies, and shows that our proposed IoT malware detection method exhibits outstanding performance, surpassing the existing iMAD methodology in terms of accuracy, precision, recall, and f1-score, proving its effectiveness.

TABLE IV
COMPARATIVE ANALYSIS OF THE PRESENTED MODEL
WITH THE CURRENT LITERATURE WORKS

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
CNN	94.75	93.00	91.43	90.43
FS + CNN	98.19	99.52	95.90	97.68
iMAD [9]	97.93	98.64	88.73	93.94
DSBEL [10]	98.50	98.42	95.97	97.12

V. CONCLUSIONS

This research paper introduces an IoT malware detection system based on an image dataset and a CNN deep learning model in addition to the use of Chi-square feature selection. This method attains robust performance with considerably high accuracy, illustrating its viability in IoT malware detection. The IoT malware detection was carried out using two different methods: the first one relied on the entire dataset, while the other one employed a smaller dataset obtained by the FS technique to investigate the impact of the chi-square FS technique on the CNN deep learning model performance and comes with solid results showing its effectiveness. Furthermore, a comparison analysis against the most recent state-of-the-art methodologies that utilized the same dataset was conducted to validate its conclusions. This knowledge motivates us to concentrate on the application of feature selection techniques within deep learning models. Additionally, there exists potential to further improve the proposed method by incorporating an additional stage involving a distinct deep learning model to enhance accuracy and sustain better outcomes.

FUNDING

None.

ACKNOWLEDGEMENT

The author would like to thank the reviewers for their valuable contribution in the publication of this paper.

CONFLICTS OF INTEREST

The author declares no conflict of interest.

REFERENCES

[1] I. H. Sarker, A. I. Khan, Y. B. Abushark, and F. Alsolami, "Internet of Things (IoT) Security Intelligence: A Comprehensive Overview, Machine Learning Solutions, and Research Directions," Mar. 07, 2022. doi: 10.20944/preprints202203.0087.v1.
[2] C. Alex, G. Creado, W. Almobaideen, O. A. Alghanam, and M. Saadeh, "A Comprehensive Survey for IoT Security Datasets Taxonomy, Classification and Machine Learning Mechanisms," Computers Security, vol. 132, p. 103283, Sep. 2023, doi: 10.1016/j.cose.2023.103283.
[3] B. M. Khammas, S. Hasan, N. Nateq, J. S. Bassi, I. Ismail, and M. N. Marsono, "First Line Defense Against Spreading New Malware in the Network," in 2018 10th Computer Science and Electronic Engineering (CEEC), Colchester, United Kingdom: IEEE, Sep. 2018, pp. 113–118. doi: 10.1109/CEEC.2018.8674214.
[4] "IoT malware detection using static and dynamic analysis techniques: A systematic literature review - Kumar - 2024 - SECURITY AND PRIVACY - Wiley Online Library." Accessed: Nov. 12, 2024. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/spy2.444
[5] A. A. Darem, F. A. Ghaleb, A. A. Al-Hashmi, J. H. Abawajy, S. M. Alanazi, and A. Y. Al-Rezami, "An Adaptive Behavioral-Based Incremental Batch Learning Malware Variants Detection Model Using Concept Drift Detection and Sequential Deep Learning," IEEE Access, vol. 9, pp. 97180–97196, 2021, doi: 10.1109/ACCESS.2021.3093366.
[6] N. Aljubory and B. M. Khammas, "Hybrid Evolutionary Approach in Feature Vector for Ransomware Detection," in 2021 International Conference on Intelligent Technology, System and Service for Internet of Everything (ITSS-IoE), Sana'a, Yemen: IEEE, Nov. 2021, pp. 1–6. doi: 10.1109/ITSS-IoE53029.2021.9615344.
[7] M. R. Babaei Mosleh and S. Sharifian, "An efficient cloud-integrated distributed deep neural network framework for IoT malware classification," Future Generation Computer Systems, vol. 157, pp. 603–617, Aug. 2024, doi: 10.1016/j.future.2024.03.051.
[8] A. F. Rasheed, M. Zarkoosh, and S. S. Al-Azzawi, "The Impact of Feature Selection on Malware Classification Using Chi-Square and Machine Learning," in 2023 9th International Conference on Computer and Communication Engineering (ICCCE), Kuala Lumpur, Malaysia: IEEE, Aug. 2023, pp. 211–216. doi: 10.1109/ICCCE58854.2023.10246084.
[9] M. Asam et al., "IoT malware detection architecture using a novel channel boosted and squeezed CNN," Sci Rep, vol. 12, no. 1, p. 15498, Sep. 2022, doi: 10.1038/s41598-022-18936-9.
[10] S. H. Khan et al., "A new deep boosted CNN and ensemble learning based IoT malware detection," Computers Security, vol. 133, p. 103385, Oct. 2023, doi: 10.1016/j.cose.2023.103385.
[11] B. Yuan, J. Wang, P. Wu, and X. Qing, "IoT Malware Classification Based on Lightweight Convolutional Neural Networks," IEEE Internet Things J., vol. 9, no. 5, pp. 3770–3783, Mar. 2022, doi: 10.1109/JIOT.2021.3100063.
[12] S. K. Smmarwar, G. P. Gupta, and S. Kumar, "Deep malware detection framework for IoT-based smart agriculture," Computers and Electrical Engineering, vol. 104, p. 108410, Dec. 2022, doi: 10.1016/j.compeleceng.2022.108410.
[13] S. K. Smmarwar, G. P. Gupta, and S. Kumar, "AI-empowered malware detection system for industrial internet of things," Computers and Electrical Engineering, vol. 108, p. 108731, May 2023, doi: 10.1016/j.compeleceng.2023.108731.
[14] M. K. Hooshmand and D. Hosahalli, "Network anomaly detection using deep learning techniques," CAAI Trans on Intel Tech, vol. 7, no. 2, pp. 228–243, Jun. 2022, doi: 10.1049/cit2.12078.
[15] V. Choudhary, S. Tanwar, and T. Choudhury, "A Hybrid Deep Learning Model for Intrusion Detection System in the Internet of Things Environment," in 2023 4th International Conference on Data Analytics for Business and Industry (ICDABI), Bahrain: IEEE, Oct. 2023, pp. 682–689. doi: 10.1109/ICDABI60145.2023.10629562.
[16] S. D. A. Rihan, M. Anbar, and B. A. Alabsi, "Approach for Detecting Attacks on IoT Networks Based on Ensemble Feature Selection and Deep Learning Models," Sensors, vol. 23, no. 17, p. 7342, Aug. 2023, doi: 10.3390/s23177342.
[17] R. Rawi, M. H. N. Hamka, H. S. Husin, and N. Allias, "Investigating the Performance of Optimizing the Convolutional Neural Network in Detecting Malware Attack," in Advances in Technology Transfer Through IoT and IT Solutions, A. Ismail, F. N. Zulkipli, Z. Awang Long, and A. Öchsner, Eds., in SpringerBriefs in Applied Sciences and Technology. , Cham: Springer Nature Switzerland, 2023, pp. 81–89. doi: 10.1007/978-3-031-25178-8_9.
[18] A. Elmasry, "IOT_Malware." Accessed: Mar. 12, 2025. [Online]. Available: https://www.kaggle.com/datasets/anaselmasry/iot-malware
[19] Y. Liu, H. Fan, J. Zhao, J. Zhang, and X. Yin, "Efficient and Generalized Image-Based CNN Algorithm for Multi-Class Malware Detection," IEEE Access, vol. 12, pp. 104317–104332, 2024, doi: 10.1109/ACCESS.2024.3435362.
[20] B. B. Gupta, A. Gaurav, R. W. Attar, V. Arya, A. Alhomoud, and K. T. Chui, "Sustainable IoT Security in Entrepreneurship: Leveraging Univariate Feature Selection and Deep CNN Model for Innovation and Knowledge," Sustainability, vol. 16, no. 14, p. 6219, Jul. 2024, doi: 10.3390/su16146219.
[21] J. P. Verma, "Chi-Square Test and Its Application," in Data Analysis in Management with SPSS Software, J. P. Verma, Ed., India: Springer, 2013, pp. 69–101. doi: 10.1007/978-81-322-0786-3_3.
[22] T. Radhakrishna and N. E. Majd, "Edge Computing Ransomware Detection in IoT Using Machine Learning," in 2024 International Conference on Computing, Networking and Communications (ICNC), Big Island, HI, USA: IEEE, Feb. 2024, pp. 244–248. doi: 10.1109/ICNC59896.2024.10556351.
[23] M. Es-Sabry et al., "Securing Images Using High Dimensional Chaotic Maps and DNA Encoding Techniques," IEEE Access, vol. 11, pp. 100856–100878, 2023, doi: 10.1109/ACCESS.2023.3315658.
[24] "Kaggle: Your Home for Data Science." Accessed: Mar. 12, 2025. [Online]. Available: https://www.kaggle.com/
[25] "Keras: Deep Learning for humans." Accessed: Mar. 12, 2025. [Online]. Available: https://keras.io/
[26] "TensorFlow." Accessed: Mar. 12, 2025. [Online]. Available: https://www.tensorflow.org/
[27] A. Vanacore, M. S. Pellegrino, and A. Ciardiello, "Fair evaluation of classifier predictive performance based on binary confusion matrix," Comput Stat, Nov. 2022, doi: 10.1007/s00180-022-01301-9.

