IJIRSET
International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY

e-ISSN: 2319-8753 | p-ISSN: 2347-6710

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 11, Issue 11, November 2022

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.118

9940 572 462 | 6381 907 438 | ijirset@gmail.com | www.ijirset.com

International Journal of Innovative Research in Science, Engineering and Technology (IJIRSET)

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.118| A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 11, November 2022 ||

| DOI:10.15680/IJIRSET.2022.1111168 |

An Advanced Malware Detection Technique Integrating Feature Selection and Deep Learning

Chamaraju Y S, Ashwini K S, Pavithra M J

Lecturer, Department of ECE, Government C.P.C Polytechnic, Mysuru, Karnataka, India

Lecturer, Department of ECE, Government Polytechnic, Chamarajanagar, Karnataka, India

Lecturer, Department of ECE, Government Polytechnic, Ramanagar, Karnataka, India

ABSTRACT: Malware continues to be a significant cybersecurity threat, spreading rapidly across networks. As malicious traffic deviates from normal communication, it exhibits asymmetry, making detection more complex and necessitating advanced techniques. Artificial intelligence has shown great potential in this area, yet there remains a lack of focus on efficiently handling large, multidimensional datasets. This study proposes a high-performance malware detection framework that integrates feature selection and deep learning to enhance accuracy and efficiency. To achieve this, two distinct malware datasets are pre-processed and subjected to correlation-based feature selection, generating multiple refined datasets. These selected features are then used to train deep learning models, including dense networks and Long Short-Term Memory (LSTM) models, which are well-suited for sequential data processing. The trained models are evaluated using key performance metrics such as accuracy, precision, recall, and F1-score to ensure a comprehensive assessment. The results reveal that in certain cases, feature selection maintains similar performance levels compared to using the full dataset. However, the level of performance variation depends on the diversity of the datasets used. This highlights the importance of selecting relevant features to optimize malware detection models while reducing computational complexity. This research addresses a crucial challenge in malware detection: managing extensive, intricate data effectively. By demonstrating the efficiency of the proposed methodology, this study contributes to improving malware classification accuracy and distinguishing harmful activity from benign behavior, ultimately enhancing cybersecurity defenses.

KEYWORDS: Malware detection, cybersecurity, feature selection, deep learning, LSTM, artificial intelligence

I. INTRODUCTION

Anything that is purposefully created with the intent to damage computers, servers, or entire networks is considered malware [1]. In its early phases, malware detection and categorization were relatively straightforward due to the absence of sophisticated cipher algorithms, allowing code cross-matching to be an effective technique. Yet, with the increasing prevalence of polymorphism and metamorphism, the threat landscape has evolved significantly, introducing new challenges, particularly with the extensive use of obfuscation techniques. As a result, traditional detection methods that rely on known signatures and patterns are becoming less effective in identifying and neutralizing advanced threats.
One of the primary reasons for the vast number of malware samples generated is the widespread adoption of obfuscation by malware developers, leading to the continuous modification and obfuscation of malicious files that share similar code and origins. This constant evolution allows malware to bypass conventional security solutions and remain undetected for longer periods. Consequently, a generalized machine learning-based malware analysis approach is considered a viable solution, as it can efficiently detect and classify previously unseen malware samples by learning from patterns rather than relying solely on predefined signatures. To enhance malware identification and classification, both static [4] and dynamic [5] analysis techniques are employed during the training phase, ensuring a more comprehensive understanding of malicious behavior.

Polymorphic malware employs an encryption mechanism that dynamically alters the encryption key while encrypting its code in each iteration. This characteristic, while adding a layer of complexity, also makes it more observable [2]. Conversely, metamorphic malware takes sophistication a step further by not only changing its encryption key but also modifying its entire code structure with each iteration. This advanced obfuscation technique makes detection highly challenging for traditional security mechanisms [3], as it prevents malware from maintaining consistent signatures that antivirus programs can detect.

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.118| A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 11, November 2022 ||

| DOI:10.15680/IJIRSET.2022.1111168 |

With metamorphic malware, the difficulty increases even further because, in addition to encrypting its code at each iteration, it also modifies the encryption key, making identification an even more complex task. The increasing volume and sophistication of malware have made manual analysis and human intervention impractical, necessitating the development of automated and intelligent detection frameworks. As the number of malware cases continues to rise daily, relying solely on human inspection and manual analysis has become insufficient, necessitating the development of more advanced detection systems. Traditional signature-based approaches struggle to keep up with these evolving threats, highlighting the need for more robust and intelligent cybersecurity solutions. The integration of artificial intelligence and deep learning techniques has the potential to bridge this gap by enabling proactive and adaptive detection methods.

To gain deeper insights into polymorphic and metamorphic malware, this research will analyze their encryption mechanisms and the unique challenges they pose to conventional detection techniques. By understanding these complexities, we aim to contribute to the development of resilient and adaptable malware detection frameworks capable of effectively identifying and mitigating sophisticated malicious software. This study will explore novel methodologies that can enhance threat intelligence and strengthen cybersecurity defences. Ultimately, this research seeks to improve malware detection accuracy, address the limitations of existing security measures, and propose innovative strategies to combat modern malware threats.

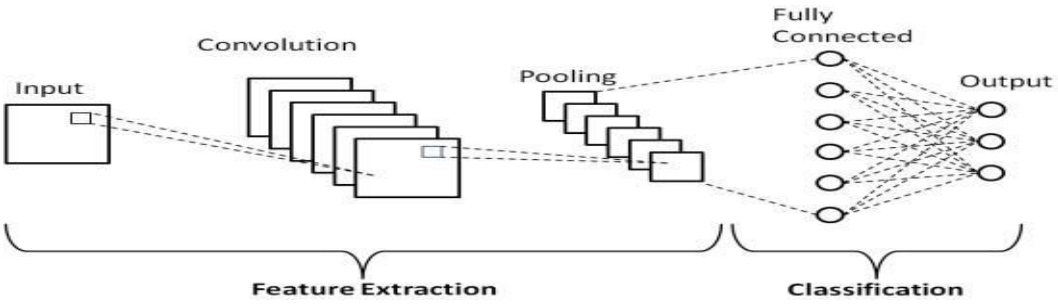

Fig1: Convolutional neural networks

II. LITERATURE SURVEY

In Rathore et al.'s work, the researchers delve into the intersection of machine learning and deep learning methodologies, emphasizing the increasing challenges in malware detection. This study explores innovative techniques for identifying and classifying malicious software by leveraging advanced algorithms and analytical procedures [1]. The authors highlight the significance of their research in addressing the complexities of contemporary cybersecurity threats by presenting their findings at a prominent big data analytics conference. The integration of machine learning and deep learning reflects a forward-looking approach, underscoring the need for intelligent and adaptable systems to counteract the constantly evolving tactics employed by malware developers. This research provides valuable insights into the ongoing efforts to safeguard digital ecosystems from sophisticated cyber threats.

Nasif, Othman, and Sani's research tackles the critical issue of data overload on IoT nodes in smart cities, where the sheer volume of generated data can be overwhelming. To mitigate this burden, the researchers propose an innovative approach that combines lossless compression techniques with deep learning solutions [2]. The application of deep learning in this context presents a modern and efficient strategy for optimizing data management in IoT environments. The study aims to minimize data size without compromising information fidelity by incorporating lossless compression techniques, thereby enhancing resource efficiency on IoT nodes and contributing to the overall sustainability of smart city infrastructures.

Yen, Moh, and Moh's study makes a notable contribution to social network analysis and cybersecurity [6]. By utilizing deep learning for behavior and text analysis, the research proposes a comprehensive strategy to counteract the rising concern of hijacked social network accounts. The authors examine user behavior and textual content to detect patterns indicative of compromised accounts, leveraging deep learning to navigate the intricate and ever-changing dynamics of social network activities. This research presents an advanced methodology by integrating behavioral analysis with text-based insights, offering a robust framework for identifying potential security breaches within social media platforms.

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.118| A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 11, November 2022 ||

| DOI:10.15680/IJIRSET.2022.1111168 |

Liu, Zhang, and Wang's paper significantly advances the field of cybersecurity, specifically in the domain of Android malware detection [8]. The authors introduce a novel approach that employs Bayesian inference to identify malicious activities within the Android ecosystem. Bayesian inference, a statistical and analytical technique, provides a structured method for assessing the likelihood of malware presence based on observed patterns and attributes. The study, presented at the prestigious CIT conference, underscores the academic relevance and practical applicability of the proposed detection framework in enhancing Android security.

Sihwail, Omar, and Ariffin's research makes an important contribution to cybersecurity by focusing on memory-based analysis for malware detection and classification [11]. The study emphasizes the need to analyze system memory, which is often the target of sophisticated attacks, to develop more reliable detection and classification techniques. Recognizing the critical role of memory-centric approaches, the authors situate their work within the broader landscape of evolving cybersecurity threats. By proposing an effective memory analysis framework, the study provides valuable insights into vulnerabilities and key indicators of malicious activity within system memory, contributing to the advancement of malware detection strategies.

III. METHODOLOGY

The proposed application integrates layers of Long Short-Term Memory (LSTM) and Convolutional Neural Networks (CNN) to develop a behavioural malware detection model based on API requests. The model is trained using a pre-processed dataset, which is split into training and testing sets. Its architecture includes an LSTM layer to capture sequential dependencies, a dense layer for binary classification, an embedding layer to represent API calls, a batch normalization layer for stabilization, a 1D convolutional layer for feature extraction, and a max-pooling layer for downsampling.

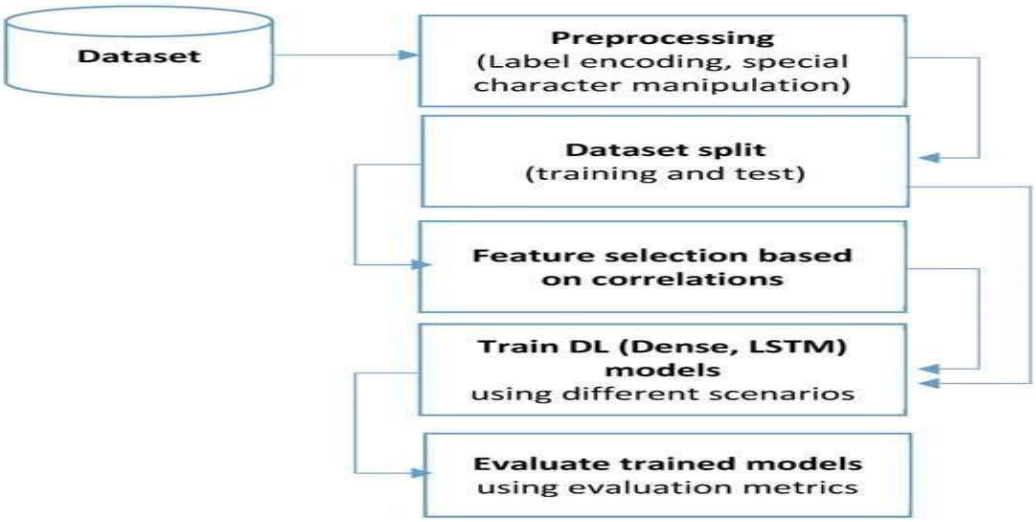

Fig 2. Proposal Model

The training process employs the Adam optimizer with binary cross-entropy loss to enhance performance. Once trained, the model is stored for future use, and its evaluation metrics, such as accuracy and loss, are visualized along with the training history. During the evaluation phase, classification metrics—including accuracy, precision, recall, and F1-score—are used to assess the model's effectiveness. Additional visualizations, such as confusion matrices, precision-recall curves, and Receiver Operating Characteristic (ROC) curves, provide deeper insights into its behavior. The final implementation includes an example demonstrating how the trained model classifies a given API call sequence as either malware or benign. By combining CNN and LSTM layers, this approach leverages the strengths of both architectures in feature extraction from sequential data. While the LSTM layer excels at capturing long-term dependencies, the CNN layer is effective in detecting local patterns within API call sequences. This hybrid technique enables the model to identify subtle patterns indicative of malware behavior. Additionally, feature importance analysis could enhance interpretability by identifying key API calls that significantly influence classification decisions. Future

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.118| A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 11, November 2022 ||

| DOI:10.15680/IJIRSET.2022.1111168 |

work may explore alternative deep learning architectures and hyperparameter optimization to further refine the model's performance.

Dataset

The dataset that has been made available is an essential part of the research that uses Deep Learning methods for malware identification and categorization. 42,797 sequences of malevolent behaviour and 1,079 sequences of benign behaviour altogether, taken from both malware and goodware cases. An MD5 hash, which is a 32- byte string, is used to uniquely identify each example in the dataset. Each 't_i' column in the dataset corresponds to a distinct API call, and the dataset contains sequences of API calls labelled 't_0' through 't_99'. The various kinds of API calls seen in the sequences are represented by the integer values in these columns, which range from 0 to 306. An intricate picture of the software's dynamic behaviour can be seen in the sequences of API calls The class of each example is indicated by the'malware' column, which also acts as the target variable. It is expressed as an integer, where 0 denotes instances of goodware (non-malicious software) and 1 denotes instances of malware.

Preprocessing:

In malware detection, preprocessing enhances dataset quality before feeding it into machine learning models. Techniques like data normalization, feature scaling, and dimensionality reduction ensure consistency and minimize biases. Categorical data is converted into a numerical format using one-hot encoding, while feature engineering extracts meaningful patterns for model training. This stage also handles missing or redundant data and addresses class imbalance through oversampling or under sampling. Exploratory data analysis helps refine preprocessing strategies by revealing feature distributions and relationships. Ultimately, preprocessing optimizes dataset quality, providing a strong foundation for training accurate and reliable malware detection models.

IV. EXPERIMENTAL RESULTS

A classifier's standard evaluation is based on a number of predefined performance indicators. Our models are assessed using the following metrics: Accuracy, Specificity, F-score, Precision, and Recall.

First, we used a confusion matrix and made inferences from it for the classification algorithms. A confusion matrix is a table that, given a set of test data for which the real values are known, provides details on the quality or performance of a model for two or more types of classes. The simplest confusion matrix for the classifier 2 class, as seen in Figure 6, is a two-dimensional confusion matrix.

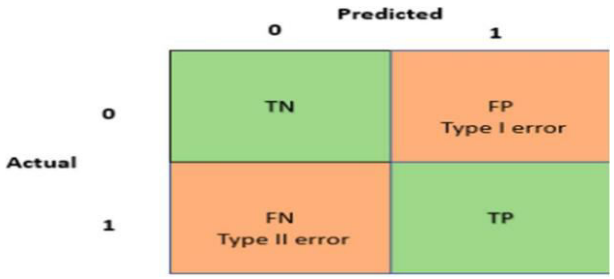

Figure 3: Confusion Matrix

It is possible to determine whether increasing the quantity of training instances could improve the confirmation score by examining these curves, which are often depicted as precision and reduction curves. In particular, when handling unknown data, the curves aid in evaluating how adding additional training cases affects model performance. Increasing the number of training instances for overfit models might result in better performance, however underfit models might not continuously gain from more instances. A loss curve that shows the system's performance throughout several epochs is shown in Figure 8. In this particular experiment, which employed five epochs, the accuracy curve and the loss curve were essential for assessing the efficacy of the model and comprehending its behaviour during training.

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.118| A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 11, November 2022 ||

| DOI:10.15680/IJIRSET.2022.1111168 |

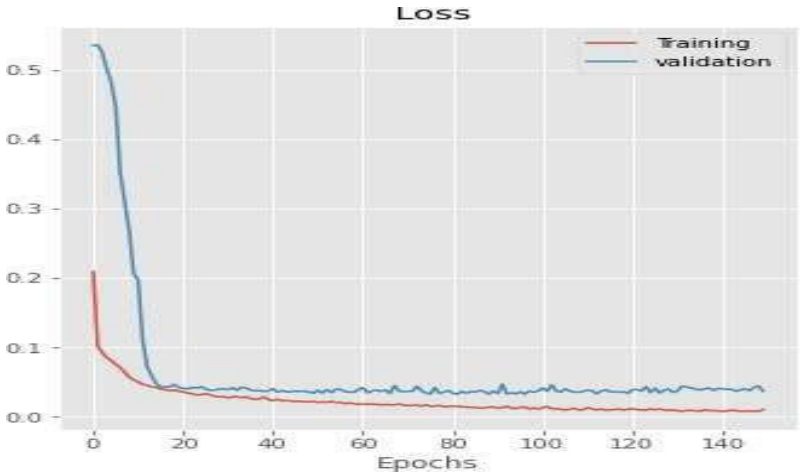

Figure 4: Loss curves

When more instances are introduced throughout the training process, acquisition curves play a crucial role in illustrating the effectiveness and dependability of a sample of learning examples. These curves are essential for determining whether adding more training examples results in an improvement in the verification value, which is the model's performance on unidentified data. Figure 9 shows graphically how adding more training instances can improve the accuracy of an overfit model with unknown data. The curve illustrates the effect of expanding the dataset size on the model's overall accuracy and gives a clear picture of the model's capacity to generalize and perform well on data that has never been seen before. Making judgements on the ideal training dataset size is made easier with the help of this visual understanding.

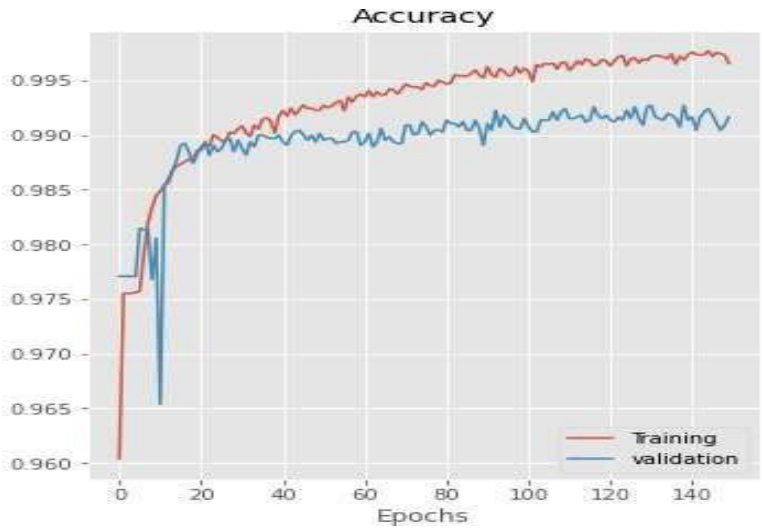

Figure 5: Model Accuracy

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.118| A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 11, November 2022 ||

| DOI:10.15680/IJIRSET.2022.1111168 |

Fig 6: Entry the Input

Based on the model's analysis, the input API call sequence is categorised as possibly harmful. Patterns and traits linked to recognised malware behaviours can be observed in the sequence. Notably, the machine learning model's training on a dataset containing a variety of benign and malicious API call sequences determines the categorization result. In addition to supporting existing malware detection efforts, this classification acts as a preventative step by spotting possible dangers within API call sequences.

Fig 7: Result

Based on the model analysis, the given API call sequence is categorized as Goodware. The sequence shows traits and patterns connected to benign or non-malicious behaviour. This classification is consistent with the training of the model on a variety of datasets containing both malicious and benign API request sequences. By helping the model differentiate between various software behaviour, the identification of Goodware supports precise and efficient malware detection.

The confusion matrix is drawn for the given model shown in figure 12The model's performance in categorizing test cases into "Benign" and "Malware" categories is seen in the confusion matrix visualization, which was created in the research The y-axis shows the actual classes, and the x-axis shows the expected classes. Axes labelled 'Benign' and 'Malware' are allocated to corresponding labels. By showing the proportion of cases that are accurately identified as "Benign" and "Malware," as well as any misclassifications, the confusion matrix sheds light on the model's performance. Off-diagonal elements denote incorrect classifications, whereas diagonal elements show the accurate predictions. The result contributes to the evaluation of the model's classification performance by helping to accurately

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.118| A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 11, November 2022 ||

| DOI:10.15680/IJIRSET.2022.1111168 |

identify both benign and harmful instances.

Figure 8: Confusion Matrix

V. CONCLUSION

In summary, this malware detection project represents a significant advancement in cybersecurity by leveraging deep learning and robust feature selection. The integration of a pre-trained model combining Long Short-Term Memory and Convolutional Neural Network architectures effectively identifies complex patterns in API call sequences. Comprehensive data preparation, including assembling and preprocessing a diverse dataset, ensures strong model training and evaluation. By applying correlation-based feature selection and analyzing instructional curves, this research enhances malware detection accuracy and efficiency.

Performance metrics such as precision-recall curves and confusion matrices provide valuable insights into the model's effectiveness. The classification of API call sequences demonstrates its ability to differentiate between malicious and benign software behaviours. Emphasizing acquisition analyses and instructional curves further clarifies the model's learning process. This work highlights the continuous evolution of malware detection strategies and contributes to strengthening cybersecurity defenses. The innovative techniques proposed pave the way for adaptable and powerful solutions to address the ever-evolving cybersecurity landscape.

REFERENCES

[1] Rathore, H.; Agarwal, S.; Sahay, S.; Sewak, M. Malware detection using machine learning and deep learning. In Proceedings of the International Conference on Big Data Analytics, Seattle, WA, USA, 10–13 December 2018; pp. 402–411.
[2] Nasif, A.; Othman, Z.; Sani, N.S. The deep learning solutions on lossless compression methods for alleviating data load on IoT nodes in smart cities. Sensors 2021, 21, 4223.
[3] Vinayakumar, R.; Alazab, M.; Soman, K.; Poornachandran, P.; Venkatraman, S. Robust intelligent malware detection using deep learning. IEEE Access 2019, 7, 46717–46738.
[4] Singh, A.; Kumar, R. A two-phase load balancing algorithm for cloud environment. Int. J. Softw. Sci. Comput. Intell. 2021, 13, 38–55.
[5] Siyuan C. Proceedings of the United Nations, Department of Economic and Social Affairs. Volume 1. United Nations; New York, NY, USA: 2014. World Urbanization Prospects; pp. 1–32.
[6] Yen, S.; Moh, M.; Moh, T.-S. Detecting compromised social network accounts using deep learning for behavior and text analyses. Int. J. Cloud Appl. Comput. 2021, 11, 97–109.
[7] Shabudin, S.; Sani, N.; Ariffin, K.; Aliff, M. Feature selection for phishing website classification. Int. J. Adv. Comput. Sci. Appl. 2020, 11, 587–595.
[8] Liu, C.-H.; Zhang, Z.-J.; Wang, S.-D. An android malware detection approach using Bayesian inference. In Proceedings of the 2016 IEEE International Conference on Computer and Information Technology (CIT), Nadi, Fiji, 8–10 December 2016; pp. 476–483.

|e-ISSN: 2319-8753, p-ISSN: 2347-6710| www.ijirset.com | Impact Factor: 8.118| A Monthly Peer Reviewed & Referred Journal |

|| Volume 11, Issue 11, November 2022 ||

| DOI:10.15680/IJIRSET.2022.1111168 |

[9] Hoornweg D., Pope K. Socioeconomic Pathways and Regional Distribution of the World's 101 Largest Cities. Volume 1 Global Cities Institute; Oshawa, ON, Canada: 2014.
[10] Qiu, J.; Zhang, J.; Luo, W.; Pan, L.; Nepal, S.; Xiang,Y. A survey of android malware detection with deep neural models. ACM Comput. Surv. 2020, 53, 1–36.
[11] Sihwail, R.; Omar, K.; Ariffin, K.A.Z. An effective memory analysis for malware detection and classification. Comput. Mater. Contin. 2021, 67, 2301–2320.
[12] Mat, S.R.T.; Razak, M.A.; Kahar, M.; Arif, J.; Mohamad, S.; Firdaus, A. Towards a systematic description of the field using bibliometric analysis: Malware evolution. Scientometrics 2021, 126, 2013–2055.
[13] Bajer M. Building an IoT data hub with elasticsearch, Logstash and Kibana; Proceedings of the 2017 5th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW); Prague, Czech Republic. 21–23 August 2017; pp. 63–68.
[14] J. Kinable and O. Kostakis, "Malware classification based on call graph clustering," J. Comput. Virol., vol. 7, no. 4, pp. 233–245, 2011.
[15] E. Gandotra, D. Bansal, and S. Sofat, "Malware Analysis and Classification," no. April, pp. 56–64, 2014.
[16] M. Z. Shafiq, S. M. Tabish, F. Mirza, and M. Farooq, "PE-Miner : Mining Structural Information to Detect Malicious Executables in Realtime Agenda Introduc1on to Domain Problem Defini1on Proposed Solu1on," 2009.
[17] B. Cakir and E. Dogdu, "Malware classification using deep learning methods," pp. 1–5, 2018.
[18] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," pp. 1–12, 2013.
[19] G. E. Dahl, J. W. Stokes, L. Deng, and D. Yu, "Large- scale malware classification using random projections and neural networks," in ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing Proceedings, 2013.
[20] S. Cesare and Y. Xiang, "Classification of malware using structured control flow," Conf. Res. Pract. Inf. Technol. Ser., 2010.
[21] M. G. Schultz, E. Eskin, F. Zadok, and S. J. Stolfo, "Data mining methods for detection of new malicious executables," 2002.
[22] B. Amos, H. Turner, and J. White, "Applying machine learning classifiers to dynamic android malware detection at scale," 2013 9th Int. Wirel. Commun. Mob. Comput. Conf. IWCMC 2013, pp. 1666–1671, 2013.
[23] B. Anderson, D. Quist, J. Neil, C. Storlie, and T. Lane, "Graph-based malware detection using dynamic analysis," J. Comput. Virol., vol. 7, no. 4, pp. 247–258, 2011.

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 6381 907 438 ijirset@gmail.com

www.ijirset.com

Scan to save the contact details