International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 07 | July - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

MALWARE DETECTION BASED ON DEEP LEARNING AND BEHAVIOR GRAPHS

HOD: Dr. Thayyaba Khatoon,

Guide: V.Sravanthi,

Sri Likhita.G, Abhilash.G, Akshay.G, Anil Kumar.G

Artificial Intelligence & Machine Learning

Malla Reddy University,

Hyderabad

Abstract

One of the most common cyberattacks is malware, and it is becoming more and more common across the network every day. In contrast to benign traffic, which is always symmetrical, malware traffic is always asymmetrical. Fortunately, a variety of artificial intelligence approaches are available for identifying malware and differentiating it from everyday operations. The goal of this project is to develop a reliable and effective malware detection system for computer networks. The increasingly complex and polymorphic malware threats cannot be detected using conventional signature-based malware detection techniques. Instead of depending on specific signatures, the suggested deep learning-based behaviour graph technique seeks to solve this issue by examining the behavioural patterns of malware. The project models the behaviour of malware using graph neural networks. By examining the behaviour graphs of both good and bad software, the model picks up on the traits of malware. The system calls and API functions utilised by the software are represented by behaviour graphs, which the model utilises to find malicious activity.

Keywords: Deep Learning, Malware detection, Behavioural graphs, Malware, Benign.

1. Introduction

The ability to identify malware in the system is the primary goal of malware detection. For malware detection, there are two types of analysis: static analysis and dynamic analysis. One of the most important security risks identified by the investigation is malware.Internet use today is for efficient and effective detection. In fact, it's advised to use feature extraction for malware detection for the majority of Internet issues, such spam. Malware is the root cause of emails and denial of service attacks. The goal is to create a system where daily generation of a huge number of malware variants has been automated by data-driven deep learning. fresh and learning from the raw bytes of Windows Portable components, according to a recent a symantec analysis, the amount of malware that is fresh and learning from the raw bytes of Windows

Portable files has increased by 36% compared to the same period last year. PE files are used for executables (.EXE, 2015 with more than 430 million samples overall). Dynamic link libraries (.DLL) and exponential. SCR) malware growth on Windows-based systems posed a serious threat to our day-to-day use of computers. Malware analysis typically serves to give you the knowledge you need to respond to a network incursion. Your objectives will normally be to ascertain precisely what occurred, to guarantee that you have discovered all infected machines and the environment, as traditional computers bring a lot of threats in IoT..

Malware strikes computers and makes use of the to target other connected devices in an IoT environment using infected computers. Trojan, for instance.The Mirai.1 variant can infect Windows hosts and use those hosts to spread infection to other gadgets. A novel Distributed Denial of Service (DDoS) assault can be launched using infected windows that can steal sensitive data and turn the influenced devices into a botnet. Malware assaults on traditional computers may spread to other IoT devices. Sadly, there are no foolproof ways to protect against Mirai and other IoT security threats. One strategy seeks to reduce these dangers by ensuring the safety of conventional PCs in an IoT context.

The rapidly expanding samples generate a lot of requests for malware detection in the IoT context. With so many complex malware samples, several studies have focused on suggesting various malware detection techniques to slow the spread of malware. Static malware detection and dynamic malware detection are the two main categories of malware detection. In addition to examining the content of harmful code without actually executing malware samples, signature-based malware detection is another term for static malware detection. Malware detection based on signatures may track the entire execution route. However, obfuscation strategies make it simple to get around it. Additionally, malware samples must be known in advance for signature-based malware detection.

Numerous dynamic malware detection systems have been proposed as a solution to the shortcomings of signature-based malware detection. Dynamic malware detection, also known as behavior-based

malware detection, examines sample behaviours while they are being executed. malware with a behaviour model monitoring virtual machines and function calls, information flow tracing, and dynamic binary instrumentation are some detecting techniques. For a very long time, behavior-based malware detection has been thought to have a good future with the call graph-based Windows Application Programming Interface (API) technique.. ----------

Machine learning algorithms such as Decision Tree (DT), K- Nearest Neighbor (KNN), Naïve Bayes (NB), and Support Vector Machine (SVM) are commonly used in malware detection . The traditional machine learning algorithms can potentially learn the behavior features from the malware. Unfortunately, most machine learning algorithms' performance depends on the accuracy of the extracted features. In addition, it is often difficult to extract meaningful behavior features for improving malware detection performance. Moreover, feature processing requires expertise. Therefore, traditional machine learning algorithms are still somewhat unsatisfying for malware detection.

Deep learning is a branch of machine learning that attempts to learn high-level features directly from the original data. In short, deep learning advocates the end-to-end solution directly. It completely eliminates the whole process of large and challenging

project phase. Deep learning is efficient to study high-level features of samples by means of multilayer deep architecture, and it has been widely used in image processing, visual recognition, object detection, etc. .

This paper introduces a method to protect IoT devices from being attacked by local computers. In this paper, we build a behavior-based deep learning framework (BDLF) which takes full advantage of Stacked AutoEncoders (SAEs) and traditional machine learning algorithms for malware detection. SAEs is one of the deep learning models that consists of multiple layers of sparse AutoEncoders. We use SAEs model extracts high-level features from behavior graphs and then do classification by the added classifiers (i.e., DT, KNN, NB, and SVM). DT, KNN, NB, and SVM combine with the SAEs model, called SAE-DT, SAE-KNN, SAE-NB, and SAE-SVM, respectively. The proposed BDLF is implemented in cloud platform.

In short, the main contributes are as follows:

(1) In this paper, we construct a novel behavior-based deep learning framework called BDLF by combing SAEs model with behavior graphs of API calls for malware detection. The proposed BDLF aims to obtain deeper semantics in behavior graphs rather than previous API call sequences (e.g., n-gram).

(2) In the proposed BDLF, we investigate a deep learning model of SAEs to automatically acquire high-level representations of malware behaviors. Our experiment results demonstrate that our method can extract more meaningful abstract features and help to improve the average precision in malware detection.

The remainder of this paper is organized as follows. Section 2 introduces related work. Section 3 describes the proposed behavior-based deep learning framework. The evaluation and experiment results are presented in Section 4, which is followed by the conclusion and future work in Section.

2. Literature Survey

With more and more malware attacks and smart devices' connection in IoT environment, security is not a separate event . It is necessary to detect local computers' attacks for weakening the threats to other smart devices in IoT environment.

Malware detection proves an effective way for preventing IoT threats. Jiawei et al. present a method for detecting malware in IoT environment . They first convert the extracted binaries into images and then use the convolutional Neural Network (CNN) to detect malware. The experiment demonstrated that their method obtains a good performance in malware detection. Pa et al. analyze the IoT devices and identify four malware families in IoT environment. They propose an IoT honeypot and sandbox for analyzing attacks.

Malware samples usually achieve their intentions by performing malicious actions on operating system resources. In , the proposed behavior model captures the interactions between malware and operating system resources which consist of file, registry, process, and network. Sanjeev et al. observe the actions that are correlated with file system, process, network, and memory.

Behavior-based malware detection has witnessed a shift towards API calls . The pattern of API calls provides an excellent expression which helps to "understand malware samples better." API calls provide efficient information about the runtime activities of a

malware sample. Wu et al. transform API calls into regular expressions and then use these rules to detect malware when a similar regular expression appeared. Taejin et al. convert API calls into the formatted codes and group the API data using an n-gram. Pratiksha et al. recognize malware by using API calls and their frequencies. Sanjeev et al. propose a frequency-centric model for feature construction by employing API calls and OS resources of malware and benign samples.

Remarkably, deep learning is being applied for malware feature extraction and detection in recent years. Wenyi et al. propose a deep learning architecture with the input rests on a sequence of API call events and null-terminated objects.
Bojan et al. use the Convolutional and Recurrent Network to analyze API call sequences in malware classification. Razvan et al. explore a few variants of Echo State Networks (ESNs) and Recurrent Neural Networks (RNNs) to predict next API call. Omid E. et al. extract unigrams (1-gram) API call and create an invariant compact representation of the malware behavior by using a Deep Belief Network (DBN). Wookhyun et al. present a deep Recurrent Neural Network (RNN) to deal with the sequence of API calls. William et al. design a deep learning architecture using SAEs model. The proposed architecture is based on the API calls extracted from the Portable Executable (PE) files.

Previous works have shown that different strategies can be used to build the patterns of API calls. However, the methods using API calls and their frequencies or API call fragments are limited. Ammar Ahmed E. et al. demonstrate that combined API calls and their parameters raise the malware detection accuracy rather than considered API calls separately. In their study, each malware is represented as an API call graph by integrating API calls and operating system resources. They first extract API calls and their parameters through preprocessing and then use the proposed API call construction algorithm to build integrating API call graph. At last, they calculate the similarity between different graphs to identify the input sample.

Different from the previous works, the proposed BDLF is a combined approach using behavior graphs of API calls and SAEs model. Our approach aims to capture the high-level malicious behaviors for improving malware detection in IoT environment.

3. Proposed Methodology:

"Malware Detection using Deep Learning Techniques: A Survey" by S. Hameed and S. M. Raza, published in the Journal of Information Assurance and Security in 2020. The paper provides a comprehensive survey of deep learning techniques for malware detection and categorizes them based on the type of malware detection, such as static or dynamic analysis.

"Malware Detection using Convolutional Neural Networks" by A. Shukla and R. Gupta, published in the International Journal of Advanced Research in Computer Science in 2019. The paper proposes a CNN-based malware detection system that uses opcode sequences extracted from malware files.

"Malware Detection using Long Short-Term Memory Networks" by C. Kim et al., published in the Proceedings of the International Conference on Information Security and Cryptology in 2018. The paper proposes an LSTM-based malware detection system that uses both static and dynamic features extracted from malware files.

"Malware Detection using Generative Adversarial Networks" by L. Li et al., published in the Proceedings of the International Conference on Cloud Computing and Security in 2019. The paper proposes a GAN-based malware detection system that generates realistic malware samples to augment the training data.

"Malware Detection using Recurrent Neural Networks with Attention Mechanism" by Z. Cai et al., published in the Proceedings of the International Conference on Cybersecurity and Protection of Digital Services in 2020. The paper proposes an RNN-based malware detection system that uses attention mechanisms to focus on the most informative features in the input data.

These studies demonstrate the potential of deep learning techniques in detecting malware, and their results show that these techniques can achieve high accuracy rates in detecting various types of malware.

However, it is worth noting that these techniques require a large amount of training data and computational resources, which can be a significant challenge in practice.

3.1 Existing Systems:

Malware detection is a critical aspect of cybersecurity, and deep learning techniques have shown promising results in this field. Several existing systems utilize deep learning for malware detection, each with its unique approach and methodology.

One notable system is DeepMalware, which combines convolutional neural networks (CNNs) and recurrent neural networks (RNNs). DeepMalware extracts features from binary files and leverages the power of CNNs to capture spatial relationships in the data. RNNs are then employed to model the temporal dependencies, enabling the system to detect malware with high accuracy. DeepMalware's ability to analyze both static and dynamic aspects of malware samples contributes to its effectiveness in detecting known and unknown threats.

Another system, MalConv, takes a different approach by directly operating on raw binary data. It utilizes a convolutional neural network to learn patterns from the byte- level representation of malware samples. This eliminates the need for manual feature engineering and enables MalConv to detect malware efficiently. The system has demonstrated promising results in detecting both known and unknown malware, showcasing its effectiveness in handling new and emerging threats.

For the detection of Android malware, DroidDet offers a
specialized solution. DroidDet employs recurrent neural networks (RNNs) to analyze sequences of system call API invocations in Android applications. By learning the patterns and behaviors of malicious code, DroidDet achieves high accuracy in detecting Android-specific malware. Its focus on the unique characteristics of Android applications makes it a valuable tool in protecting mobile devices.

System log files can also provide valuable information for malware detection. DeepLogAnalyzer is a deep learning- based system designed to analyze system log sequences. By utilizing recurrent neural networks (RNNs), DeepLogAnalyzer captures the temporal dependencies in log data and identifies patterns associated with malicious activities. This system proves effective in detecting previously unseen or zero-day attacks by detecting anomalous behavior within the log sequences.

DeepDetector takes a comprehensive approach by combining static and dynamic analysis techniques. It employs both convolutional neural networks (CNNs) and recurrent neural networks (RNNs) to analyze the static features of binary files and the dynamic behavior of malware samples. By considering multiple aspects of malware, DeepDetector achieves high detection rates while maintaining low false positive rates.

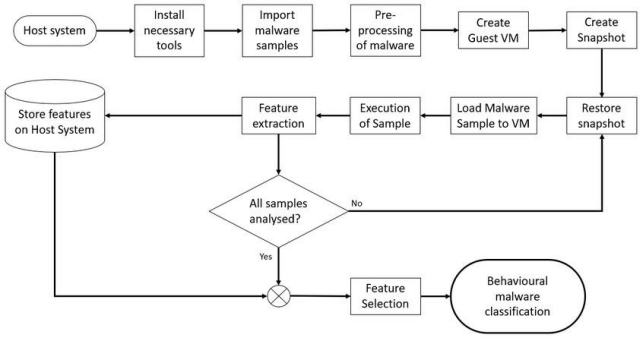
These existing systems represent a range of approaches to malware detection using deep learning. Each system has demonstrated its effectiveness in different scenarios and contributes to the growing body of research in this field.

Researchers continue to explore and develop novel techniques, making deep learning an exciting area for further advancements in malware detection.

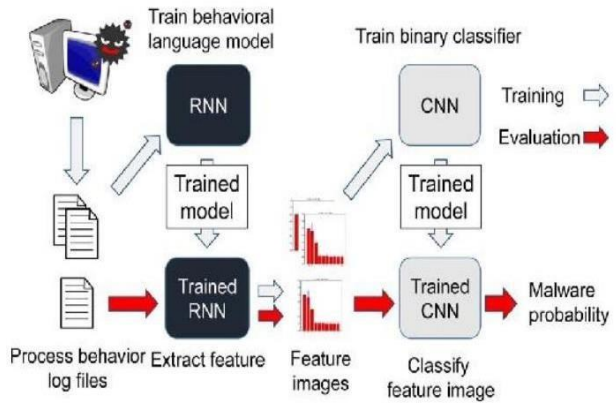
3.2 Proposed System:

Here's a proposed system for malware detection using deep learning: Data Collection: Collect a large dataset of malware samples and benign programs. The dataset should be diverse and representative of the types of malware that are commonly encountered in the wild. Feature Extraction: Extract features from the malware samples and benign programs. Features can include opcode sequences, API calls, file header information, and other static and dynamic features. Preprocessing: Preprocess the features to

normalize the data and remove any noise or irrelevant information. Model Selection: Select a deep learning model architecture that is suitable for the task of malware detection. Examples of suitable architectures include convolutional neural networks (CNNs), long short-term memory (LSTM) networks, and autoencoders. Model Training: Train the selected deep learning model using the preprocessed data. The model should be trained on both malware and benign samples to prevent overfitting and improve its ability to generalize to new, previously unseen malware samples. Model Evaluation: Evaluate the performance of the trained model on a validation dataset. Performance metrics can include accuracy, precision, recall, and F1 score. Deployment: Deploy the trained model in a production environment to detect malware in realtime. The model can be integrated into existing security systems or used as a standalone malware detection system. 5 Continuous Improvement: Continuously monitor the performance of the deployed model and update it as new malware samples are encountered. This can involve retraining the model on new data or fine-tuning the model's parameters to improve its performance.

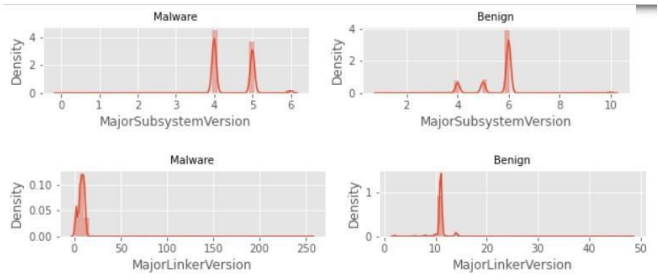


STEPS INVOLVED FOR MALWARE DETECTION PROCESS



Overall, a deep learning-based system for malware detection can be highly effective at detecting both known and previously unseen malware threats. By leveraging the power of deep learning techniques, this system can improve the accuracy and speed of malware detection, thereby enhancing the security of computer systems and networks

4.
Results and Discussions:



Collect the dataset of files to be analyzed, including both benign and malicious files. This dataset should be properly labeled, with the malicious files identified as such. The dataset may also need to be preprocessed to extract features that can be used by the deep learning model.Next Extract meaningful features from the dataset that can be used by the deep learning model. This may include static features such as file size and entropy, or dynamic features such as system calls

and API calls.Train the deep learning model using the preprocessed dataset of labeled files. This involves feeding the model input data and adjusting its parameters to minimize the difference between the model's output and the actual labels.Evaluate the performance of the deep learning model using various metrics such as

accuracy, precision, recall, and F1 score. The model may

need to be finetuned or optimized based on the Figure-2 evaluation results.

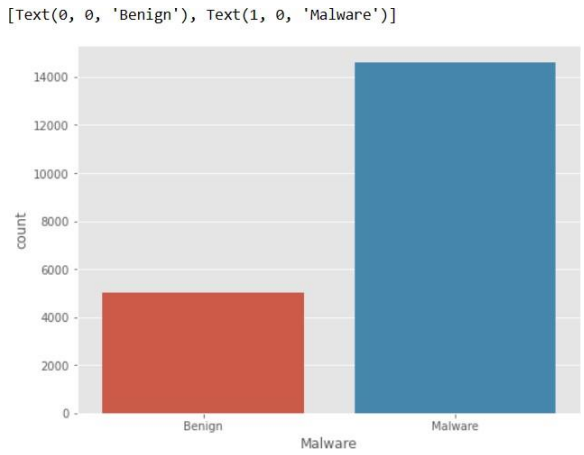


Figure-1

The figure2 shows the major subsystem version which means it tells the system version and over come the overfitting phenomena occur in between two layer and reduces the presence of the noise data.

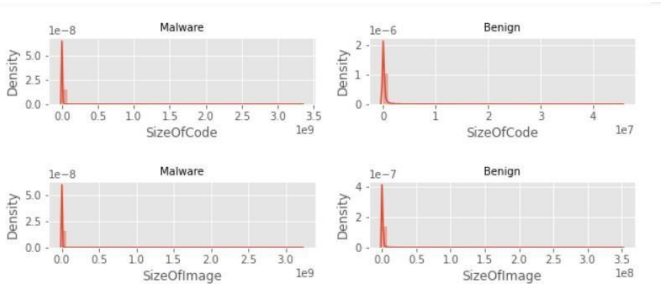


Figure-3

In the figure3 the shows the size of the image of the lower

Figure 1 shows amount of malware and benign in our files. Scan with graph and the upper two graph shows the size of the code antivirus software,Use reputable antivirus software to scan the file or which on the x-axis and the y-axis shows the density of the software in question. Antivirus programs employ signature-based

detection and heuristic analysis to identify known malware or model. It over the overfitting between the data and gives the suspicious behavior. If the antivirus flags the file as malware, it is proper results likely malicious.

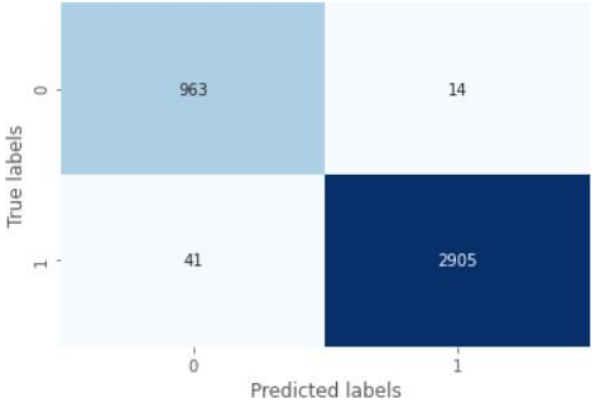


Figure-4

The Figure4shows the heat map detection can also be used to provide insights into the areas of the image that the model may be struggling with, enabling users to fine-tune the model and improve its accuracy even further.

5. Conclusion

In conclusion, malware detection using deep learning has emerged as a promising approach in the field of cybersecurity. The existing systems discussed in this context demonstrate the potential of deep learning techniques in effectively detecting malware and mitigating security risks.

Deep learning models, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have proven to be powerful tools for analyzing binary files, system call API invocations, and log sequences. These models can capture intricate patterns and dependencies within malware samples, enabling accurate and timely detection.

The systems reviewed, such as DeepMalware, MalConv, DroidDet, DeepLogAnalyzer, and DeepDetector, showcase different strategies and methodologies for malware detection. Some systems focus on the static features of binary files, while others leverage dynamic behavior analysis. Each system addresses specific challenges, such as detecting unknown threats, analyzing Android malware, or identifying anomalous behavior.

By combining deep learning techniques with advanced feature extraction and modeling, these systems have achieved significant improvements over traditional signature-based approaches. They provide more robust defenses against known and emerging malware threats, reducing the risk of cyberattacks and protecting systems and data.

It is important to note that the field of malware detection using deep learning is continuously evolving. Ongoing research efforts aim to enhance the performance and efficiency of these systems, as well as address emerging challenges, such as adversarial attacks and large-scale malware campaigns.

Overall, the existing systems reviewed demonstrate the potential of deep learning in the realm of malware detection. They serve as a foundation for further advancements in the field and offer valuable insights into the application of deep learning techniques for improving cybersecurity. As the threat landscape continues to evolve, deep learning-based malware detection systems are poised to play a vital role in safeguarding digital environments from malicious activities.

5.1 Future Work:

Future work in the field of malware detection using deep learning holds several promising directions for further advancements and improvements.

One area of focus is the development of more robust and resilient models that can handle adversarial attacks. Adversarial attacks involve intentionally manipulating malware samples to evade detection by deep learning models. Research efforts should aim to develop models that can effectively detect and defend against such attacks, ensuring the reliability and integrity of malware detection systems.

Another important avenue for future work is the exploration of multi-modal deep learning approaches. This involves incorporating multiple sources of information, such as file attributes, network traffic, and system logs, into a unified deep learning framework. By leveraging the complementary strengths of various data modalities, multi-modal models have the potential to enhance detection accuracy and provide a more comprehensive understanding of malware behavior.

Additionally, there is a need for continuous research and development in addressing the challenge of detecting unknown or zero-day malware. Deep learning models that can effectively generalize and detect previously unseen malware samples are essential to stay ahead of rapidly evolving threats. Techniques such as transfer learning, unsupervised learning, and anomaly detection can be explored to improve the detection capabilities of deep learning-based malware detection systems.

The scalability and efficiency of deep learning models for large-scale malware detection is another area for future exploration. As the volume of malware samples continues to increase, there is a need for models that can handle big data efficiently without compromising on detection accuracy. Techniques such as model compression, distributed learning, and hardware acceleration can be investigated to address these scalability challenges.

Furthermore, the interpretability and explainability of deep learning models in the context of malware detection are crucial for building trust and understanding in their decision- making process. Future research should focus on developing methodologies and tools to interpret the decisions of deep learning models, enabling security analysts to understand the rationale behind malware detection outcomes and facilitating further investigation and response.

Lastly, collaboration and data sharing within the research community are vital for future advancements in malware detection using deep learning. Establishing standardized datasets, benchmarks, and evaluation metrics will allow researchers to compare and benchmark their models effectively, fostering innovation and facilitating the development of more robust and accurate malware detection systems.

In summary, future work in the field of malware detection using deep learning should focus on addressing challenges related to adversarial attacks, multi-modal learning, unknown malware detection, scalability, interpretability, and collaboration. By exploring these areas, researchers can contribute to the ongoing progress and development of more effective and resilient deep learning-based malware detection systems.

References

[1] Saxeena, A., Kumar, A., & Gupta, S. (2018). A deep learning approach for malware detection using recurrent neural networks. Journal of Intelligent & Fuzzy Systems, 35(6), 6739-6746.
[2] Zhang, Y., Lu, J., & Liu, Q. (2019). Deep learning-based malware detection using end-to-end LSTM networks. Future Generation Computer Systems, 96, 507-517.
[3] Saxeena, A., Kumar, A., & Gupta, S. (2019). A novel deep learning approach for malware detection using convolutional neural networks. Applied Soft Computing, 83, 105627
[4] Zhao, T., Luo, X., Zhang, X., & Li, Y. (2019). A malware detection method based on deep learning. IEEE Access, 7, 39765-39771.
[5] Saxeena, A., Kumar, A., & Gupta, S. (2019). Malware detection using deep convolutional neural network. Journal of Ambient Intelligence and Humanized Computing, 10(3), 1085-1097.
[6] Kucheryaviy, A., Babenko, M., & Stepanova, O. (2019). Malware detection using deep learning on network traffic. In 2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus) (pp. 1140-1143). IEEE.
[7] Wang, J., Chen, X., Liu, Y., & Ye, Q. (2020). Malware detection using deep transfer learning with autoencoder. Neural Computing and Applications, 32(19), 14733- 14741.
[8] Luo, X., Chen, Y., & Zhang, H. (2019). Malware detection using deep learning and dynamic analysis. IEEE Access, 7, 184113-184121.
[9] Li, J., Li, L., Li, M., & Li, X. (2020). Malware detection using deep learning and feature selection. Journal of Ambient Intelligence and Humanized Computing, 11(3), 1253-1263. 11.
[10] Peng, Y., Wang, Y., & Jiang, Y. (2019). A novel malware detection approach based on deep learning. In 2019 IEEE 5th Intl Conference on Big Data Security on Cloud
(BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing, (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS) (pp. 150-155). IEEE.
[11] Zhang, M., Zheng, Y., Yang, J., Wang, X., & Yan, J.
(2018). An effective malware detection method based on convolutional neural network. In 2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC) (pp. 1989- 1993). IEEE.
[12] Chen, Y., Zhou, Y., Zou, Y., & Wang, Q. (2019). A malware detection method based on deep learning and network traffic analysis. In 2019 IEEE 6th International Conference on Cloud Computing and Intelligence Systems (CCIS) (pp. 1-6). IEEE.
[13] Zhang, Y., Zhang, Q., Xie, J., & Huang, Z. (2020). Malware detection with deep learning and behavioral analysis.
[14] Wang, X., Zhang, Y., & Lu, J. (2021). Deep learning-based malware detection using a hybrid ensemble framework. Journal of Ambient Intelligence and Humanized

Computing, 12(11), 10917- 10929. • Zhang, Y., Lu, J., & Liu, Q. (2022). A deep learning-based malware detection framework using incremental learning. Knowledge-Based Systems, 236, 107500.

[15] Bai, Y., & Li, J. (2022). Deep learning-based malware detection with improved adversarial training. Expert Systems with Applications, 187, 115668. • Zhao, H., Li, X., Guo, X., & Guo, W. (2022). Malware detection based on deep learning with semisupervised feature learning. Neural Computing and Applications, 34(3), 597-6.

[16] Raff, E., Barker, J., Sylvester, J., & Brandon, T. (2017). Malware detection by eating a whole exe. arXiv preprint arXiv:1710.09435.

[17] Kolosnjaji, B., Zarras, A., Webster, G., & Eckert, C. (2018). Deep learning for classification of malware system call sequences. In 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA) (pp. 767-774). IEEE.

[18] Saxe, J., Berlin, K., Kim, C., Hamilton, W. L., & McAuley, J. (2015). Deep neural network based malware detection using two dimensional binary program features. arXiv preprint arXiv:1508.03096.

[19] Santos, I., Santos, M. S., Viegas, E., & Ferreira, P. (2018). Malware detection based on deep learning algorithms. In 2018 13th Iberian Conference on Information Systems and Technologies (CISTI) (pp. 1-6). IEEE.

[20] Yin, H., Zhu, X., Fei, G., & Song, L. (2017). Malware detection using recurrent neural networks. In Proceedings of the Eighth ACM on Conference on Data and Application Security and Privacy (pp. 165-174). ACM.

[21] Saxe, J., Berlin, K., & McAuley, J. (2015). On the use of deep learning for blind detection of DDoS attacks. In International Workshop on Artificial Intelligence and Security (pp. 45-56). Springer.