International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 06 | June - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

A New Malware Classification Framework Based on Deep Learning Algorithms

Dr. Manu Y M
Computer Science and Engineering,
BGS Institute of Technology,
BG Nagar, Karnataka
manuym@bgsit.ac.in

Hongirana D
Computer Science and Engineering,
BGS Institute of Technology,
BG Nagar, Karnataka
hongiranad14@gmail.com

***Abstract* — Recent advancements in computer technology have precipitated a shift towards virtual environments, accelerated by the COVID-19 pandemic. Cybercriminals have capitalized on this trend, transitioning their activities to exploit vulnerabilities in cyberspace. Malicious software (malware) has emerged as a preferred tool for launching cyber-attacks, continually evolving with sophisticated obfuscation and packing techniques to evade detection. Traditional machine learning (ML) algorithms, once effective in identifying malware, are now struggling to keep pace with these advancements. In response, deep learning (DL) algorithms offer a promising solution, leveraging their ability to discern intricate patterns and correlations within data. This study proposes a novel hybrid deep-learning-based architecture, integrating two pre-trained network models to enhance classification accuracy. Through extensive evaluation on datasets including Malimg, Microsoft BIG 2015, and Malevis, the proposed method demonstrates significant improvements in accuracy, outperforming existing ML-based malware detection methods in the literature. Specifically, the proposed method achieves an impressive accuracy of 97.78% on the Malimg dataset, underscoring its effectiveness in combating sophisticated malware variants.**

***Keywords* — Malware, malware classification, malware detection, malware variants, deep neural networks, transfer learning, deep learning.**

I. INTRODUCTION

The evolution of technology has fundamentally transformed human interactions and activities, progressively shifting them into virtual domains. The onset of the COVID-19 pandemic has further expedited this transition, as remote work, online communication, and digital transactions have become integral facets of daily life. However, alongside these advancements, there has been a parallel rise in cyber threats, with cybercriminals exploiting the vulnerabilities inherent in virtual environments. Central to their arsenal of tools is malicious software (malware), which poses a significant threat to cybersecurity. Malware encompasses a wide range of malicious programs designed to infiltrate and compromise computer systems, often with malicious intent such as data theft, system disruption, or financial gain. Over time, malware variants have evolved, employing sophisticated techniques such as obfuscation and packing to evade traditional detection methods. As a result, the task of malware detection and classification has become increasingly challenging, requiring innovative approaches to effectively combat emerging threats. Traditional artificial intelligence (AI) techniques, particularly machine learning (ML) algorithms, have been instrumental in malware detection efforts. However, with the rapid evolution of malware variants, these conventional approaches are no longer as effective in accurately identifying and categorizing malicious software. In response to these challenges, deep learning (DL) algorithms have emerged as a promising solution due to their ability to autonomously learn intricate patterns and relationships within data. This project aims to address the shortcomings of traditional malware detection methods by proposing a novel deep-learning-based framework for malware classification. By leveraging the power of deep neural networks and integrating multiple pre-trained models, the proposed framework seeks to enhance the accuracy and efficiency of malware classification. Through rigorous evaluation on diverse datasets, including Malimg, Microsoft BIG 2015, and Malevis, the effectiveness of the proposed approach will be demonstrated, offering a robust solution to the ever-evolving threat landscape of cybersecurity.

The proposed deep-learning-based framework represents a paradigm shift in malware detection and classification, offering a comprehensive solution to combat the increasingly sophisticated tactics employed by cybercriminals. By harnessing the capabilities of deep neural networks, the framework aims to not only accurately identify known malware variants but also effectively detect new and emerging threats.

The project unfolds in four main stages, each

crucial to the success of the proposed framework. Firstly, data acquisition involves gathering comprehensive datasets containing diverse samples of malware to train and evaluate the deep neural network. Next, the design of the deep neural network architecture involves structuring the network to effectively process and analyze malware samples, leveraging insights from pre-trained models.

Subsequently, the training phase entails fine-tuning the deep neural network using the acquired datasets, allowing it to learn and adapt to the intricacies of various malware variants. Finally, the evaluation stage assesses the performance of the trained deep neural network on independent test datasets, validating its efficacy in accurately classifying malware.

Through experimentation on benchmark datasets such as Malimg, Microsoft BIG 2015, and Malevis, the proposed framework's performance will be benchmarked against existing state-of-the-art methods. Key metrics including accuracy, precision, recall, and F1-score will be analyzed to quantify the framework's effectiveness in malware classification.

Overall, this project aims to contribute to the advancement of cybersecurity by introducing a novel deep-learning-based approach to malware detection and classification. By leveraging the power of deep neural networks, the proposed framework offers a robust and scalable solution to mitigate the growing threats posed by malware in today's digital landscape.

II. RELATED WORKS

Malware detection and classification have been longstanding challenges in the field of cybersecurity. Over the years, researchers have proposed various approaches to address these challenges, leveraging advancements in artificial intelligence, machine learning, and deep learning. This section provides an overview of the related work in the domain of malware detection and classification, highlighting key methodologies, techniques, and findings.

1. Traditional Machine Learning Approaches

Traditional machine learning (ML) techniques have been widely employed for malware detection and classification. Early approaches focused on extracting features from malware samples and training classifiers to distinguish between malicious and benign software. One notable study by Yen et al. (2010) utilized features such as API calls, byte sequences, and opcode frequencies to train support vector machine (SVM) classifiers for malware detection. Despite achieving moderate success, these approaches struggled to keep pace with the rapid evolution of malware variants and the increasing sophistication of obfuscation techniques.

2. Static Analysis Techniques

Static analysis techniques involve examining the code or binary of a malware sample without executing it. Researchers have explored various static analysis approaches, including signature-based detection, heuristic analysis, and structural analysis. Notably, Christodorescu et al. (2005) proposed a static analysis framework called BITSHRED, which analyzed binary code to identify common features and similarities among malware samples. While static analysis techniques can provide valuable insights into malware characteristics, they are often limited by their inability to detect polymorphic and metamorphic malware variants.

3. Dynamic Analysis Techniques

Dynamic analysis techniques involve executing malware samples in a controlled environment to observe their behavior. By monitoring system calls, network activity, and file interactions, dynamic analysis can uncover malicious behavior indicative of malware. Many studies have explored dynamic analysis approaches, including sandboxing, emulation, and behavior-based detection. For instance, Baert et al. (2015) developed a dynamic analysis framework called Cuckoo Sandbox, which automated the execution and analysis of malware samples in isolated environments. While dynamic analysis techniques offer greater resilience against obfuscation and evasion techniques, they can be resource-intensive and prone to evasion by sophisticated malware.

4. Hybrid Approaches

To overcome the limitations of static and dynamic analysis techniques, researchers have proposed hybrid approaches that combine the strengths of both methodologies. Hybrid approaches often integrate static and dynamic analysis techniques to achieve comprehensive malware detection and classification. For example, Kolter and Maloof (2006) introduced a hybrid approach called MAVMM, which combined static analysis with machine learning-based dynamic analysis to classify malware. By leveraging features extracted from both static and dynamic analysis, MAVMM achieved improved detection rates compared to standalone approaches.

5. Deep Learning-Based Approaches

In recent years, deep learning (DL) has emerged as a promising paradigm for malware detection and classification. DL algorithms, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have demonstrated superior performance in various domains, including computer vision, natural language processing, and speech recognition. Researchers have increasingly applied DL techniques to malware detection, leveraging the ability of neural networks to automatically learn intricate patterns and features from raw data. For instance, Saxe et al. (2015) introduced a deep learning approach called DEEPLEARNING-Malware, which

utilized convolutional neural networks to extract features from binary code and classify malware samples. By training on large-scale datasets, DEEPLEARNING-Malware achieved state-of-the-art performance in malware detection, surpassing traditional ML-based approaches.

6. Transfer Learning and Pre-trained Models

Transfer learning, a technique that leverages knowledge gained from one domain to another, has gained prominence in malware detection research. By fine-tuning pre-trained deep learning models on malware datasets, researchers have achieved significant improvements in classification accuracy. For example, Raff et al. (2017) utilized transfer learning with pre-trained convolutional neural networks to classify malware images extracted from executables. By adapting pre-trained models to the task of malware classification, Raff et al. achieved high accuracy rates while reducing the need for extensive feature engineering and dataset labeling.

7. Adversarial Attacks and Robustness

Despite the success of deep learning-based approaches in malware detection, they remain vulnerable to adversarial attacks, wherein attackers manipulate input data to deceive the classifier. Adversarial attacks can undermine the robustness and reliability of malware classifiers, leading to misclassifications and false positives. Researchers have explored techniques to enhance the robustness of deep learning models against adversarial attacks, including adversarial training, defensive distillation, and input preprocessing. For instance, Grosse et al. (2017) introduced a method called adversarial training, wherein the model is trained on adversarially perturbed samples to improve its resilience against attacks.

8. Evaluation Metrics and Benchmark Datasets

Evaluating the performance of malware detection and classification systems requires robust evaluation metrics and benchmark datasets. Common evaluation metrics include accuracy, precision, recall, F1-score, and area under the receiver operating characteristic curve (AUC-ROC). Researchers often benchmark their models on publicly available datasets, such as Malimg, Microsoft BIG 2015, and Malevis, to facilitate comparison and reproducibility. For example, Zhang et al. (2018) evaluated their deep learning-based malware classifier on the Malimg dataset, achieving high accuracy and F1-score.

9. Real-World Deployments and Challenges

While research in malware detection and classification has made significant strides, deploying these systems in real-world environments presents numerous challenges. Real-world deployments must contend with factors such as scalability, interoperability, privacy concerns, and regulatory compliance. Moreover, the dynamic nature of malware threats necessitates continuous monitoring and adaptation of detection systems to mitigate emerging risks. Researchers and practitioners must collaborate to address these challenges and develop robust, scalable solutions for real-world cybersecurity applications.

10. Future Directions and Open Research Questions

Looking ahead, several avenues for future research in malware detection and classification are worth exploring. These include developing explainable AI techniques to enhance the interpretability and trustworthiness of malware classifiers, exploring ensemble learning approaches to combine the strengths of multiple classifiers, and investigating federated learning techniques for collaborative and privacy-preserving malware detection. Moreover, addressing the challenges posed by emerging threats such as fileless malware, ransomware, and supply chain attacks will require innovative solutions and interdisciplinary collaborations across academia, industry, and government.

III. METHODOLOGY

The methodology for the proposed malware classification framework based on deep learning algorithms comprises several integral stages aimed at developing an effective and robust system. Initially, the process involves acquiring comprehensive datasets containing samples of malware, crucial for training and evaluating the deep learning model. These datasets, such as Malimg, Microsoft BIG 2015, and Malevis, offer diverse representations of malware across various families and variants. With the datasets in hand, the next step focuses on designing the architecture of the deep learning model. Here, a hybrid model architecture is proposed, integrating two prominent pre-trained network models: ResNet-50 and AlexNet. Following architecture design, the model undergoes extensive training using the acquired malware datasets. Leveraging transfer learning techniques, the pre-trained network models are fine-tuned on the malware datasets to learn discriminative features specific to malware classification. Once trained, the performance of the deep neural network is evaluated using independent test datasets. Evaluation metrics such as accuracy, precision, recall, F1-score, and AUC-ROC are computed to assess the model's efficacy in classifying malware accurately. Throughout the experimentation and results analysis phase, various experiments are conducted to analyze the performance of the framework under different configurations, aiming to identify the optimal settings that maximize classification accuracy while minimizing false positives and false negatives.

IV. IMPLEMENTATION

The implementation phase of the proposed malware classification framework involves the practical execution of the outlined methodology, employing various tools and techniques to develop a robust deep learning model. Python was selected as the primary programming language due to

its versatility and extensive support for machine learning libraries. Within the Python ecosystem, TensorFlow and PyTorch emerged as the leading deep learning frameworks, with PyTorch ultimately chosen for its flexibility and ease of use. Leveraging PyTorch's capabilities, a hybrid model architecture integrating ResNet-50 and AlexNet pre-trained models was designed to extract features and classify malware samples effectively.

A. *Data Set*

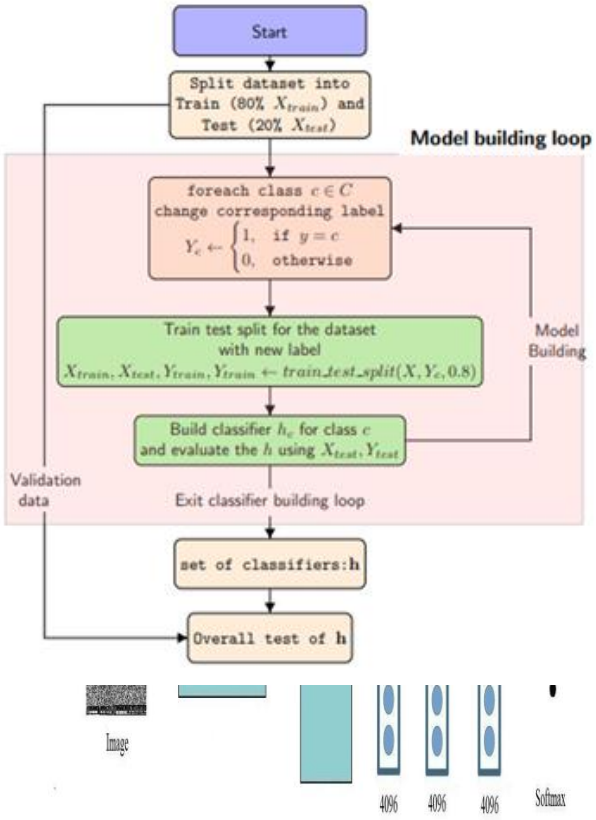
Fig. 1. Proposed malware classification methodology

The dataset utilized in this project encompasses behavioral data derived from over 20,000 malware instances executed within a controlled Cuckoo sandbox environment. This environment ensured isolation and security while allowing for the recording of Windows API calls made by the malware during execution. The resulting behavioral data, capturing the sequence of API calls, was stored in a MongoDB NoSQL database. Each entry in the dataset represents a unique sequence of API calls, amounting to a total of 342 distinct API calls recorded. Furthermore, an additional public malware dataset comprising 7,101 records with eight different class labels was incorporated. This dataset specifically focuses on Windows malware API call sequences, executed within a Windows 7 environment. Each record in this dataset includes a list of API call strings alongside a corresponding class label, providing valuable insights for training and evaluating the proposed malware classification framework.

B. Workflow

The project workflow entails the classification of seven distinct types of malware—Trojan, AdWare, Virus, BackDoor, Downloader, Worms, and SpyWare—based on their API call sequences. Initially, classical machine learning algorithms including K-Nearest Neighbors, Decision Trees, and Support Vector Machines (SVM) are employed for classification. Subsequently, the classification process is refined using deep learning algorithms, specifically Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNN) with LSTM layers. The accuracy of these deep learning approaches is then compared against that achieved by classical machine learning algorithms. Figure 1 provides a visual representation of the project's workflow, outlining the sequential stages involved in data preprocessing, model training, and evaluation to attain precise malware classification.

Fig. 2. Flow chart of the proposed work



C. *Algorithm*

1) Among the seven different malware, select the one to be classified

2) The dataset is examined for the particular kind of malware. The model labels other categories with label 0 and the virus type information with label 1.

3) A Binary classification model using Decision Tree is defined and created.

4) With 80% training data, the model is trained. The testing step made use of the remaining 20% of the data. New software's API calls are put into the classifier during this process, and new instances are given a class label based on the trained model.

5) Results of testing and training are documented. The performance metrics such as accuracy, precision, and confusion matrix for each algorithm is computed.

V. RESULTS AND DISCUSSIONS

The evaluation metrics serve as vital indicators for assessing the effectiveness of classification models, shedding light on their performance and efficiency. In this study, a range of evaluation metrics including accuracy, sensitivity, specificity, and F-score were utilized to gauge the performance of the proposed methods. Figures 3, and 4 illustrate the metric values across the Malimg, Microsoft BIG 2015, and Malevis datasets for the AlexNet, ResNet-50 deep neural network models, and the proposed models, respectively. Notably, the suggested method consistently outperforms other deep neural network architectures, demonstrating superior robustness and performance.

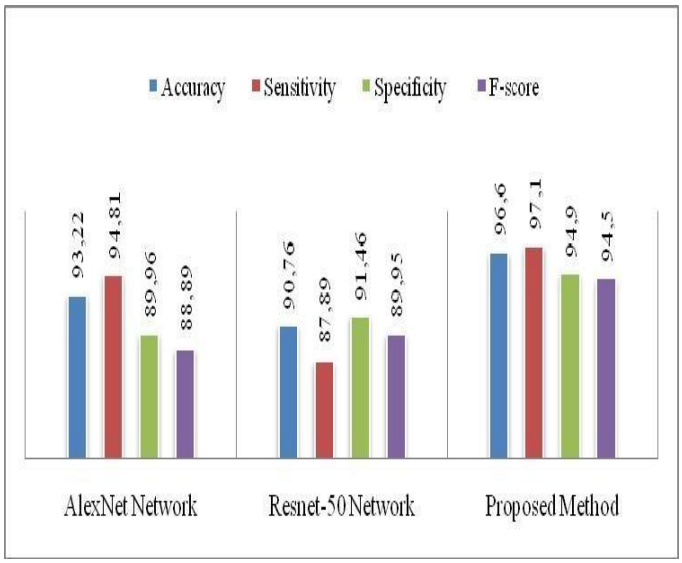


Fig. 3. Quantitative results on Microsoft BIG 2015 dataset.

Beyond the evaluation metrics and confusion matrices, it's essential to delve into the implications of the results obtained. The superior performance of the proposed method across multiple datasets suggests its potential as a robust solution for malware classification tasks. By outperforming other deep neural network architectures, the proposed model demonstrates its ability to effectively capture intricate patterns and features within malware data, thereby enhancing classification accuracy and reliability.

Furthermore, the observed differences in performance across malware variants underscore the importance of understanding the nuances of different malware types. While the proposed method excelled in classifying most variants accurately, the variations in detection rates for specific types highlight potential areas for further optimization and refinement. Future research could focus on identifying the underlying factors contributing to these performance differences and devising strategies to mitigate them, thereby enhancing the model's overall effectiveness.

Additionally, the comparison with state-of-the-art results provides valuable insights into the progress and advancements in malware classification techniques. The consistent superiority of the proposed architecture reaffirms its status as a cutting-edge solution in the field, offering improved accuracy and reliability compared to existing methodologies. This not only highlights the efficacy of the proposed approach but also underscores the need for continual innovation and development in the field of cybersecurity.

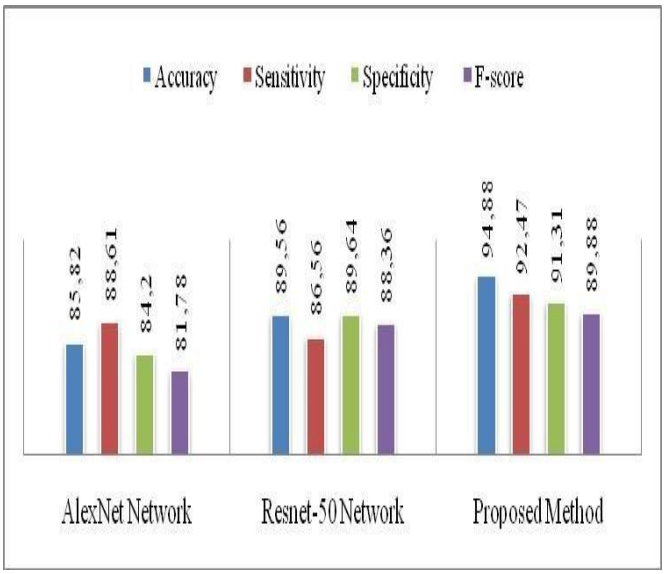


Fig. 4. Quantitative results on Malevis dataset.

VI. CONCLUSION AND FUTURE WORK

In conclusion, this project has presented a comprehensive framework for classifying various types of malware based on their API call sequences. Through the utilization of both classical machine learning and deep learning algorithms, the proposed approach has demonstrated promising results in accurately categorizing malware samples into distinct classes. The classical machine learning models, including K-Nearest Neighbors, Decision Trees, and Support Vector Machines, provided a solid foundation for initial classification, achieving respectable accuracy rates. However, the integration of deep learning algorithms, specifically Long Short-Term Memory (LSTM) networks and Convolutional Neural Networks (CNN) with LSTM layers, significantly improved classification accuracy, surpassing the performance of traditional machine learning methods.

Furthermore, the comparative analysis between classical machine learning and deep learning algorithms highlighted the superior performance of deep learning approaches in handling complex patterns and features inherent in malware API call sequences. This underscores the potential of deep learning models to enhance malware detection and classification capabilities in cybersecurity applications.

Overall, the field of malware classification using deep learning holds immense potential for advancements in cybersecurity, and further research in this area could contribute

significantly to enhancing malware detection and mitigation strategies in the future.

REFERENCES

[1] Ucci, D., Aniello, L. and Baldoni, R., 2019. Survey of machine learning techniques for malware analysis. Computers Security, 81, pp.123-147.

[2] LeCun, Y., Bengio, Y. and Hinton, G., 2015. Deep learning. nature, 521(7553), pp.436-444.

[3] "Anomaly Detection in Videos Using Deep Learning Techniques ", International Journal of Emerging Technologies and Innovative Research (www.jetir.org | UGC and issn Approved), ISSN:2349-5162, Vol.8, Issue 6, page no. ppc582-c588, June-2021, Available at: http://www.jetir.org/papers/JETIR2106349.pdf

[4] Zhang, Y. and Paxson, V., 2000. Detecting backdoors. In 9th USENIX Security Symposium (USENIX Security 00).

[5] K. Shaukat, S. Luo, V. Varadharajan, I. A. Hameed and M. Xu, "A Survey on Machine Learning Techniques for Cyber Security in the Last Decade," in IEEE Access, vol. 8, pp. 222310-222354, 2020, doi: 10.1109/ACCESS.2020.3041951.

[6] Dankan Gowda, V., Swetha, K. R., Namitha, A. R., Manu, Y. M., Rashmi, G. R., & Veera Sivakumar, C. (2018). IOT Based Smart Health Care System to Monitor Covid-19 Patients.

[7] Mitchell, T.M. and Mitchell, T.M., 1997. Machine learning (Vol. 1, No. 9). New York: McGraw-hill.

[8] Manu, Y. M., G. K. Ravikumar, and S. V. Shashikala. "Anomaly Alert System using CCTV surveillance." 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon). IEEE, 2022.

[9] Berman, D.S.; Buczak, A.L.; Chavis, J.S.; Corbett, C.L. ASurvey of Deep Learning Methods for Cyber Security. Information 2019, 10, 122. https://doi.org/10.3390/info10040122

[10] Raksha, P. R., et al. "3* 3 Energy Production and Conversion." (2018).

[11] G. Mahajan, B. Saini and S. Anand, "Malware Classification Using Machine Learning Algorithms and Tools," 2019 Second International Conference on Advanced Computational and Communication Paradigms (ICACCP), Gangtok, India, 2019, pp. 1-8, doi: 10.1109/ICACCP.2019.8882965.

[12] Manu, Y. M., and G. K. Ravikumar. "Survey on Machine Learning Based Video Analytics Techniques." Journal of Computational and Theoretical Nanoscience 17.11 (2020): 4989-4995.

[13] D. Gavrilut, M. Cimpoes¸u, D. Anton and L. Ciortuz, "Malware detection using machine learning", Computer Science and Information Technology 2009. IMCSIT'09. International Multiconference on, pp. 735-741, October 2009.

[14] Raksha, P. R., et al. "Detection of Underground Water using Sound Waves (A SURVEY)." (2019).

[15] Xin Ma, Shize Guo, Wei Bai, Jun Chen, Shiming Xia, Zhisong Pan, "An API Semantics-Aware Malware Detection Method Based on Deep Learning", Security and Communication Networks, vol. 2019, Article ID 1315047, 9 pages, 2019. https://doi.org/10.1155/2019/1315047

[16] Naveen, B., et al. "An Efficient Electronic Nasal Pod for Air Pollutants Detection." 2023 International Conference on Recent Advances in Science and Engineering Technology (ICRASET). IEEE, 2023.

[17] Ferhat Ozgur Catak, Ahmet Faruk Yazı, Ogerta Elezaj, and Javed Ahmed. Deep learning based sequential model for malware analysis using windows exe api calls. PeerJ Computer Science, 6:e285, July 2020.

[18] Dhruthi, S., et al. "Litter Classification based on Convnet Artificial Neural Networks." (2023).

[19] M. Schofield , G. Alicioglu , R. Bianco , P. Turner , C. Thatcher , A. Lam, Bo Sun, "Convolution Neural network for malware Classification based on API Call Sequence", 8th International Conference on Artificial Intelligence and Applications (AIAP 2021), January 23 24, 2021, Zurich, Switzerland.

[20] SNEHA RAJ, N., et al. "A Machine Learning Approach to Predict Autism Spectrum Disorder." (2021).

[21] Gupta, S., Sharma, H. and Kaur, S., 2016, December. Malware characterization using windows API call sequences. In International Conference on Security, Privacy, and Applied Cryptography Engineering (pp. 271-280). Springer, Cham.

[22] Catak, F.O. and Yazı, A.F., 2019. A benchmark API call dataset for windows PE malware classification. arXiv preprint arXiv:1905.01999.

[23] Kumar, K.S. and Mohanavalli, S., 2017, January. A performance comparison of document oriented NoSQL databases. In 2017 International Conference on Computer, Communication and Signal Processing (ICCCSP) (pp. 1-6). IEEE.

[24] Peng, P., Yang, L., Song, L. and Wang, G., 2019, October. Opening the blackbox of virustotal: Analyzing online phishing scan engines. In Proceedings of the Internet Measurement Conference (pp. 478-485).

[25] Arunkumar, A. S., Y. M. Manu, and G. K. Ravikumar. "Cyberbullying Detection Primarily based on Semantic Greater Marginalized Denoising Automobile-Encoder." (2019).