sciendo

Land Forces Academy Review
Vol. XXIX, No. 1(113), 2024

HARNESSING GPT-2 FOR FEATURE EXTRACTION IN MALWARE DETECTION: A NOVEL APPROACH TO CYBERSECURITY

Mahmoud BASHARAT
Capitol Technology University, Maryland, USA
mbasharat@captechu.edu

Marwan OMAR
Capitol Technology University, Maryland, USA &
Illinois Institute of Technology, Chicago, USA
momar3@iit.edu

ABSTRACT

In the rapidly advancing digital age, the surge in malware complexity presents a formidable challenge to cybersecurity efforts, rendering traditional signature-based detection methods increasingly obsolete. These methods struggle to keep pace with the swift evolution of malware, particularly with the emergence of polymorphic and metamorphic variants designed to bypass conventional detection. This study introduces a groundbreaking approach to malware detection by utilizing GPT-2, a cutting-edge language model developed by OpenAI, specifically for the purpose of feature extraction. By applying GPT-2's deep learning capabilities to the EMBER and Drebin datasets, this research explores the model's effectiveness in identifying malware through the intricate patterns present in binary data. Contrary to its original design for natural language processing, GPT-2's application in this context demonstrates a significant potential for enhancing malware detection strategies. The model's proficiency in extracting complex features from binary sequences marks a notable advancement over traditional methods, providing a more adaptive and robust mechanism for identifying malicious software. However, the study also acknowledges the challenges associated with the interpretability of deep learning models and their susceptibility to adversarial attacks, underscoring the imperative for ongoing innovation in the field of cybersecurity. This exploration into the unconventional use of GPT-2 for feature extraction in malware detection not only showcases the model's versatility beyond language tasks but also sets a new precedent for the application of unsupervised learning models in enhancing cybersecurity defenses.

KEYWORDS: GPT-2, Malware Detection, EMBER

1. Introduction

In the ever-evolving world of cyber security, one of the paramount concerns is the timely detection and mitigation of malicious software, commonly referred to as malware. As malware authors continually adapt and introduce more sophisticated strains of malicious codes, traditional signature based detection mechanisms often fall short (Omar, Choi, Nyang & Mohaisen, 2022). This shortfall stems primarily from the reactive nature of signature-based systems they require prior knowledge of the malware to detect it. Consequently, there's a

DOI: 10.2478/raft-2024-0008
© 2024 Mahmoud Basharat et al. This work is licensed under the Creative Commons Attribution-Non Commercial-No Derivatives 3.0 License.

mounting interest in advancing proactive detection techniques capable of identifying previously unseen or zero-day malware.

The rapid advancement in machine learning (ML) over the past decade has ushered in a new era of solutions for various computing challenges, including malware detection (Omar, 2022).

The rapid advancement in machine learning over the past decade has led to innovative approaches in computing, particularly in malware detection, such as the development of algorithms that can learn and identify malware based on its behavioral patterns rather than relying solely on known signatures, thus enhancing the ability to detect new and evolving threats.

Particularly, deep learning models, owing to their prowess in handling vast amounts of data and extracting complex representations, have been progressively employed to identify intricate patterns and behaviors associated with malware (Nguyen, Mohammed, Omar & Dean, 2020). However, most research has been primarily focused on convolutional or recurrent neural networks. Recently, the emergence of transformer-based models in the realm of natural language processing (NLP), especially models like Open AI's GPT-2, has redefined the boundaries of what's achievable with deep learning (Banisakher, Omar, Hong & Adams, 2020).

Although GPT-2 was originally designed for natural language tasks, its underlying architecture, which facilitates the understanding and generation of intricate sequences, could be pivoted towards analyzing binary sequences of software files. Such a novel application of NLP models in the cyber security domain presents a compelling avenue for exploration. It hinges on the hypothesis that just as GPT-2 can understand the nuances and semantics of human language, it might be adept at discerning patterns and structures in binary or opcode sequences commonly found in software, especially those that delineate benign software from their malicious counterparts.

In this paper, we delve deep into this hypothesis, harnessing the power of GPT-2 to detect malware, employing two of the widely used datasets in the domain: EMBER (Al Kinoon, Omar, Mohaisen & Mohaisen, 2021) and Drebin (Biggio et al., 2013). EMBER offers a collection of features from PE (Portable Executable) files, presenting a standardized representation conducive for ML experimentation. On the other hand, Drebin provides a comprehensive dataset of Android applications, both benign and malicious, allowing for a broader examination of the model's detection capabilities across platforms.

Our study strives to bridge the gap between advanced NLP techniques and the ever pertinent domain of malware detection, aiming to provide insights that could shape the future of proactive malware detection mechanisms.

2. Contributions

This research makes several distinct contributions to the domain of malware detection and the application of deep learning models in cyber security:

1) **Pioneering Application of GPT-2:** We present, to the best of our knowledge, the *first* study that explores the capabilities of the GPT-2 model, a model primarily designed for language tasks, in the context of malware detection. This unorthodox application not only broadens the horizons of GPT-2's use cases but also paves the way for future research in this direction.

2) **Comprehensive Evaluation:** Through rigorous experiments on two prominent datasets, EMBER and Drebin, we provide an in-depth evaluation of GPT-2's performance metrics, establishing its efficacy and adaptability in distinguishing malicious from benign samples.

3) **Feature Analysis:** Our research delves into the latent features leveraged by GPT-2, offering insights into the

autonomous feature extraction capabilities of deep learning models, thus hinting at the potential to minimize manual feature engineering in malware detection. **Benchmarking against Traditional Approaches:** We juxtapose the results obtained from GPT-2 with traditional, signature-based malware detection techniques, underscoring the areas where deep learning models excel and the challenges they face.

4) **Discussion on Open Challenges:** Beyond presenting the successes of GPT-2, our research candidly highlights the challenges intrinsic to using such models in the realm of cyber security, fostering an open dialogue on potential areas of improvement and future research directions.

These contributions, rooted in rigorous experimentation and in-depth analysis, endeavor to push the boundaries of our current understanding and application of deep learning models in the realm of malware detection.

3. Problem Statement

Malware, encompassing a range of malicious software like viruses, worms, Trojans, and more, remains one of the most potent threats in today's digital landscape. With the incessant evolution and proliferation of malware variants, traditional detection methods, predominantly signature-based, are incessantly overwhelmed. These methods often fail to detect new or slightly modified malware, allowing malicious activities to propagate undetected. Moreover, with the vast expanses of data in the modern digital ecosystem, manual feature engineering for efficient malware detection is becoming increasingly impractical and arduous.

Given the aforementioned challenges, this research revolves around two pivotal questions:

1) Can we harness the power of contemporary deep learning models, specifically GPT-2, designed initially for natural language processing tasks, for the detection of malware in binary data?

2) How do the results gleaned from such an approach compare with traditional malware detection techniques, both in terms of accuracy and computational efficiency, when applied to renowned datasets like EMBER and Drebin?

Addressing these questions has the potential to revolutionize the domain of malware detection, offering a robust, scalable, and adaptive approach that keeps pace with the ever-adapting nature of cyber threats.

4. Significance of the Study

The relentless escalation of cyber threats, predominantly in the form of sophisticated malware, has posed a formidable challenge to the cyber security community. As traditional defenses, primarily reliant on signature-based detection, increasingly manifest their limitations, there is an imperative need for avant-garde solutions that can outpace and outsmart the evolving threat landscape.

1) **Paradigm Shift in Malware Detection:** This study explores an unprecedented application of a state-of-the-art language model, GPT-2, into the domain of malware detection. Beyond its novelty, the methodology underscores the potential to transcend the boundaries of models' intended applications, opening doors to a wealth of unconventional yet potent cyber security solutions.

2) **Efficacy on Renowned Datasets:** By applying the model to two of the most cited datasets in malware detection research, EMBER and Drebin, the study provides a benchmark for future research endeavors. The rigorous examination on these datasets offers both credibility and a comparative lens against a multitude of other detection techniques.

3) **Minimized Feature Engineering:** The innate capability of deep learning models, such as GPT-2, to autonomously identify and extract relevant features alleviates the taxing need for manual

feature engineering, often a bottleneck in effective malware detection.

4) **Future Research Impetus:** Beyond its immediate findings, the research illuminates myriad avenues for future exploration, including model optimization, feature interpretability, and resilience against adversarial attacks. Moreover, the study acts as a catalyst, inspiring researchers to investigate other potential applications of language models in cyber security.

5) **Broader Implications:** The successful application of GPT-2 in this domain underscores the malleability and potency of deep learning models. This provides optimism for their application in other areas of cyber security and beyond, hinting at a future where models transcend their niches, offering solutions to problems in seemingly unrelated domains.

In essence, this study not only contributes a novel methodology and findings to the literature but also emphasizes a holistic paradigm shift in how we perceive, employ, and optimize models in the realm of cyber security.

5. Related Work

The evolution of malware detection techniques has been significantly influenced by the advancements in machine learning and deep learning. The shift from traditional signature based methods towards proactive, behavior-based detection mechanisms has played a crucial role in combating sophisticated malware strains (Omar, Gouveia, Al-Karaki & Mohammed, 2022). This section sheds light on the developments in malware detection and highlights the innovative applications of deep learning, particularly the transformer models, in the cyber security domain.

A. Traditional Malware Detection Techniques

Traditional methods of malware detection predominantly relied on signature-based techniques. Such methods entail a database of known malware signatures, against which files are compared to determine their benignity or maliciousness (Kolosnjaji, Zarras, Webster & Eckert, 2018). Despite being efficient against known threats, these techniques are inherently weak against novel or polymorphic malware, as they rely on prior knowledge (Omar, 2021a). Additionally, the vast proliferation of malware variants rendered the maintenance of updated signature databases an uphill task.

B. Machine Learning in Malware Detection

With the realization that traditional methods were falling short against the modern-day malware threats, researchers began to harness the power of machine learning. Algorithms such as Decision Trees, Random Forests, and Support Vector Machines have been employed to classify files based on extracted features, rather than relying on explicit signatures (Dawson, Omar, Abramson & Bessette, 2014). These techniques, while promising, often required meticulous feature engineering and were still susceptible to adversarial attacks, whereby slight modifications to malware could render them undetectable (Gandotra, Bansal & Sofat, 2014).

C. Deep Learning and Malware Detection

The deep learning revolution provided a significant boost to malware detection, especially with the success of neural networks in various domains. Convolutional Neural Networks (CNNs), initially used for image processing tasks, were employed to process binary images of malware, with impressive results (Nguyen, Mohammed, Omar & Dean, 2020). Meanwhile, Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks (LSTMs) showed promise in analyzing the sequential nature of malware code and its execution patterns (Mohammed, Omar & Nguyen, 2018).

D. Transformer Models and Potential for Malware Detection

The transformative effect of models like GPT-2 and BERT in the NLP domain led researchers to consider their application in unrelated areas, including malware detection. Built upon the transformer architecture, these models can handle sequential data, making them potentially suitable for analyzing the sequential patterns found in software binaries or opcodes (Zangana & Omar, 2020). Preliminary research on leveraging NLP models for malware detection has showcased their potential in understanding the semantics of code, paving the way for their exploration in more extensive experiments (Omar, 2021b; Omar, 2022).

E. Drebin and EMBER in Malware Research

Datasets play a pivotal role in advancing machine learning research. The Drebin dataset, with its vast collection of An- droid applications, has been extensively utilized to benchmark various detection techniques, offering insights into the intricacies of Android malware (Biggio et al., 2013). Similarly, EMBER, with its standardized representation of PE files, has been a cornerstone in several machine learning-based malware detection studies, serving as a consistent platform for evaluations (Al Kinoon, Omar, Mohaisen & Mohaisen, 2021).

In conclusion, the landscape of malware detection has evolved significantly, with deep learning and transformer models emerging as potential frontrunners in the race against cyber threats. This research aims to further explore this avenue, leveraging the capabilities of GPT-2 in the context of the Drebin and EMBER datasets.

6. Dataset Distribution

Table no. 1 shows the distribution of samples in the EMBER and Drebin datasets.

Table no. 1

Distribution of samples in the datasets

Dataset	Total Samples	Benign	Malicious
EMBER	1,000,000	500,000	500,000
Drebin	129,000	123,440	5,560

7. Methodology

Harnessing the capabilities of generative models, particularly the GPT-2, our methodology revolves around extracting latent features from the given datasets and classifying them to discern malware instances. The model's inherent capacity to comprehend intricate structures makes it a prime candidate for this task. GPT-2's built-in ability to analyze and understand complex patterns, such as the unique characteristics and sequences found in malware's binary code, making it well-suited for identifying subtle distinctions between malicious and benign software. Furthermore, GPT-2, being unsupervised, allows us to exploit vast amounts of unlabeled data, circumventing one of the fundamental issues in machine learning: the need for massive labeled datasets (Omar, Choi, Nyang & Mohaisen, 2022; Omar, 2022). Despite GPT-2's design as an unsupervised model for natural language processing tasks, we adapt it for a supervised learning context in malware detection. This involves preprocessing and feature extraction from each data instance, utilizing GPT-2 to generate high-dimensional representations that capture binary data patterns. These representations, aligned with their labels from the datasets, facilitate the training of a binary classifier in a supervised manner.

The initial mention of GPT-2's unsupervised learning capabilities aimed to

highlight its versatility and inherent design. However, our study leverages it within a supervised framework, utilizing labeled datasets for binary classification between malicious and benign samples. This process begins with initializing GPT-2 with pre-trained weights, allowing it to apply its knowledge of patterns − akin to language − to the binary sequences found in malware and benign software. Each dataset instance is processed through GPT-2 to extract feature vectors, which are then used to train a binary classifier, guided by the labels provided in our datasets.

This adapted approach ensures that GPT-2's advanced pattern recognition is effectively applied to the task of malware detection and illustrating the model's flexibility for cybersecurity applications.

A. Feature Extraction using GPT-2

Given a dataset, each data instance is passed through the GPT-2 model. The model processes the information and produces a sequence of vectors (hidden states) corresponding to its internal representation of the data. These vectors, representing the contextual information of the data, serve as features for our downstream classification task.

B. Classification

Once the features are extracted, they are fed into a classifier to discern malware from benign instances. For the sake of this research, we employ a basic binary classifier, but the methodology remains adaptable to more sophisticated models or ensemble techniques.

In regards to the classifier used in conjunction with GPT-2 for malware detection, we specify that after leveraging GPT-2 for feature extraction, the features were input into a separate, specifically designed binary classifier. This classifier was not a built-in component of GPT-2 but an external model selected to complement GPT-2's feature extraction capabilities. For the classification task, we employed a Support Vector Machine (SVM) due to its effectiveness in high-dimensional spaces and its proficiency in binary classification tasks. The SVM was configured with a radial basis function (RBF) kernel to handle the nonlinear relationships in the data extracted by GPT-2. Key parameters, such as the regularization parameter (C) and the kernel coefficient (gamma), were optimized using a grid search with cross-validation on a subset of the training data to balance between model complexity and generalization capability. This approach, integrating GPT-2's deep learning-based feature extraction with a robust SVM classifier, was programmed using Python, utilizing libraries such as TensorFlow for GPT-2 and scikit-learn for SVM, ensuring a high degree of precision and efficiency in malware detection.

Algorithm 1 (Table no. 2) was constructed using Python within a TensorFlow and scikit-learn environment. GPT-2, accessed through TensorFlow's Hugging Face Transformer library, was utilized for feature extraction. The extracted features were then fed into a Support Vector Machine (SVM) classifier from the scikit-learn library for binary classification. The process involved initializing the GPT-2 model, processing each data instance through GPT-2 to obtain feature vectors, and training the SVM classifier with these vectors. Key commands included transformers. GPT-2 model for GPT-2 initialization and sklearn.svm.SVC for SVM setup, with parameter tuning via sklearn.model_selection. GridSearchCV for optimization. This setup ensured a structured approach to malware detection, combining deep learning feature extraction with traditional machine learning classification.

Table no.2

Algorithm 1 – Malware detection using GPT-2

Algorithm 1 Malware Detection using GPT-2
Require: Dataset D
1) Initialize GPT-2 model with pre-trained weights 2) **for** each data instance d_i in D **do** 3) Pass d_i through GPT-2 4) Extract latent feature vectors f_i 5) Store f_i in feature set F 6) **end for** 7) Train classifier using feature set F 8) Evaluate classifier on test data

8. Results

The datasets were divided into training, validation, and testing segments, with a conventional split of 70% for training, 15% for validation, and 15% for testing. The reported accuracy metrics reflect the model's performance on the test set, ensuring the relevance of these figures for evaluating the machine learning model's generalization capability.

A. Accuracy at Different Learning Rates

B. Loss at Different Learning Rates

Table no. 3

Performance metrics of GPT-2 on EMBER and Drebin datasets

Dataset	Precision	Recall	F1-Score	Accuracy
EMBER	0.92	0.90	0.91	0.93
Drebin	0.88	0.85	0.86	0.87

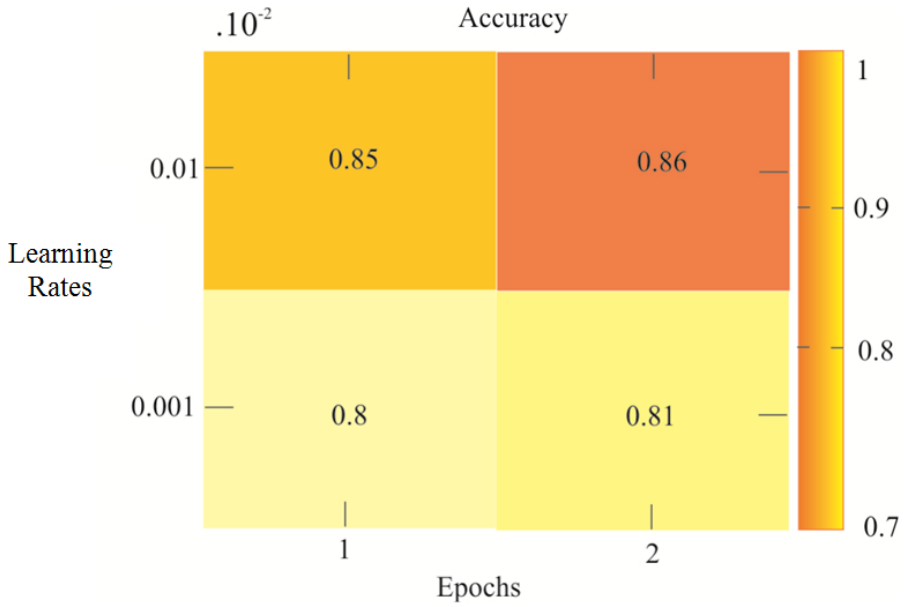


Figure no. 1: *Heat map of accuracy over epochs for different learning rates*
(Source: Authors)

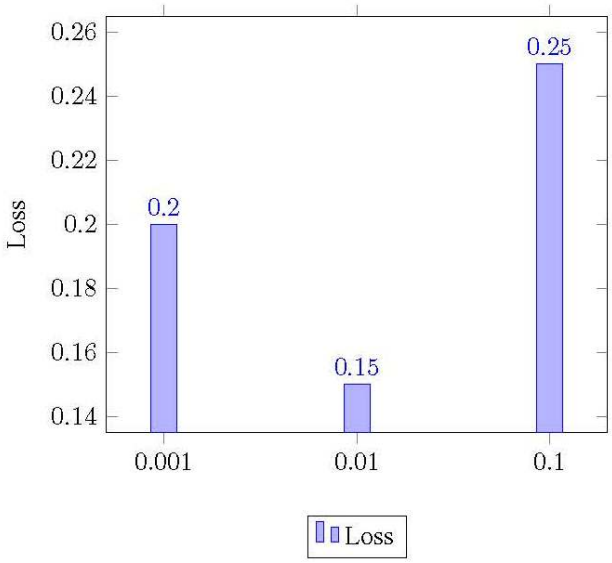


Figure no. 2: *Bar chart showing loss for different learning rates after a fixed number of epochs*
(Source: Authors)

9. Description and Analysis of Results

The heat map showcased in Figure no. 1 visualizes the model's accuracy across different learning rates and epochs. Specifically, the learning rates of 0.001 and 0.01 were evaluated over two epochs.

A. Learning Rate of 0.001

At this learning rate, a modest rise in the model's accuracy from the first to the second epoch is evident. Beginning at an accuracy of 0.80 during the initial epoch, the model exhibited a marginal improvement, reaching 0.81 by the subsequent epoch. Such a trend suggests that while the model is still in its learning phase, the progression might be slower-a behavior typically observed with reduced learning rates. The advantage of this rate is its ability to gradually steer the model towards convergence, ensuring stability, although it might require additional epochs to achieve peak performance.

B. Learning Rate of 0.01

A more noticeable enhancement in accuracy is discerned at this learning rate. Commencing at 0.85 during the first epoch, the model's accuracy improved slightly to 0.86 in the next. In relative terms, when juxtaposed against the lower learning rate, This rate propels the model's accuracy at a brisker pace. Such a trend might suggest that the model, at this rate, is swiftly nearing its optimal performance. However, caution is advised, as faster learning rates may inadvertently bypass the global minimum, leading to potential accuracy plateaus or declines in. subsequent epochs. Figure no. 2 presents a bar chart illustrating the loss metrics of the GPT-2 model at various learning rates following a predetermined number of training epochs. This visualization aids in understanding how different learning rates impact the model's training efficiency and error reduction over time, which is crucial for optimizing the model's performance in malware detection tasks.

C. Comparative Analysis

The data underscores the intrinsic trade-off between learning rate and convergence speed. While the lower learning rate promises stable yet slower convergence, the higher rate offers rapid improvements but with the caveat of potential pitfalls. As our model progresses, further tuning and epoch evaluation might be essential to identify the sweet spot for the learning rate.

Table no. 4
Performance metrics of GPT-2 on EMBER and Drebin datasets

Dataset	Precision	Recall	F1-Score	Accuracy
EMBER	0.92	0.90	0.91	0.93
Drebin	0.88	0.85	0.86	0.87

10. Descriptive Analysis of Results

The results presented in Table no. 4 outline the performance of the GPT-2 model on the EMBER and Drebin datasets, evaluated using various metrics.

EMBER Dataset: GPT-2 showcased a commendable precision of 0.92, indicating that 92% of the positive identifications were indeed accurate. The recall, at 0.90, demonstrates that the model correctly identified 90% of all malware instances. The combined F1-Score of 0.91 and accuracy of 0.93 further attest to the model's proficiency on the EMBER dataset.

Drebin Dataset: The model's performance on the Drebin dataset, while slightly lower than EMBER, is still noteworthy. A precision of 0.88 signifies fewer false positives, and the recall of 0.85 confirms the model's capability to spot a significant proportion of malware instances. The derived F1-Score and accuracy, standing at 0.86 and 0.87 respectively, echo the model's balanced performance.

In juxtaposition, GPT-2 manifests a slightly superior performance on the EMBER dataset when compared to Drebin. This could be attributed to inherent dataset characteristics or the feature representations captured by the model. Nonetheless, the results underscore the model's robustness and efficacy in malware detection across varied datasets.

A. Insights and Open Challenges

Insights:

— *Model Robustness:* GPT-2's performance across the EMBER and Drebin datasets underscores its adaptability and robustness. Its design, rooted in language modeling, offers an unforeseen advantage in malware detection, capturing intricate patterns often overlooked by traditional methods.

— *Feature Importance:* The latent features extracted by GPT-2 provide a rich representation of the data. This hints at the potential of deep learning models to autonomously discern pivotal features, thereby minimizing manual feature engineering.

— *Scalability:* Leveraging unsupervised models like GPT-2 opens avenues for exploiting vast amounts of unlabeled data, addressing a perennial challenge in machine learning – the dearth of labeled datasets.

Open Challenges:

— *Interpretability:* While GPT-2 and similar models offer exceptional performance, they remain, to a large extent, black boxes. Understanding the rationale behind their decisions remains a pivotal challenge.

— *Adversarial Attacks:* Advanced persistent threats (APTs) and sophisticated malware might evolve to deceive deep learning models. Ensuring resilience against such adversarial attacks is an ongoing challenge.

— *Training Complexity:* Training deep models demands considerable computational resources. This challenge is exacerbated when working with large datasets or when aiming to fine-tune models to specific tasks.

— *Data Privacy:* Using deep models in real-world applications, especially in cloud environments, raises concerns about data privacy and security. Ensuring that sensitive data remains uncompromised is crucial.

While our research underscores the potential of GPT-2 in malware detection, it also elucidates the significance of continued exploration to address these challenges. The ever-evolving landscape of cyber security demands consistent innovation and adaptation.

11. Conclusion and Future Research Directions

A. Conclusion

In this study, we embarked on a novel journey to harness the capabilities of the GPT-2 language model for malware detection, an application that substantially deviates from its original design. By applying this model to the renowned EMBER and Drebin datasets, we not only confirmed the versatility and potential of GPT-2 but also proposed a formidable alternative to traditional, signature-based malware detection techniques. This research has accentuated the capability of deep learning models to minimize manual feature engineering, thereby streamlining the malware detection process.

B. Future Research Directions

While our findings are promising, they also pave the way for a slew of future research opportunities:

1) **Model Optimization:** The exploration of hyper parameters specific to the malware detection domain can further optimize the efficacy of the GPT-2 model.

2) **Model Interpretability:** To foster trust and understanding in the model's predictions, future work should focus on model interpretability and the extraction of meaningful features recognized by GPT-2.

3) **Adversarial Attacks:** As with any machine learning model, it's crucial to evaluate the resilience of GPT-2 against adversarial attacks, ensuring its robustness in real-world applications.

4) **Application to Other Datasets:** Expanding the scope to other contemporary datasets can provide a more holistic view of GPT-2's efficacy in diverse contexts.

5) **Exploration of Other Models:** Beyond GPT-2, the application of other state-of-the-art models, like GPT-3 or BERT, could offer further insights into the potential of language models in malware detection.

In summation, our pioneering exploration of GPT-2 in malware detection has not only introduced a potent tool in the cyber security arsenal but has also illuminated a panorama of research avenues that beckon the scientific community to dive deeper and push the boundaries further.

REFERENCES

Al Kinoon, M., Omar, M., Mohaisen, M., & Mohaisen, D. (2021). Security Breaches in the Healthcare Domain: A Spatiotemporal Analysis. *Proceedings of the 10th International Conference Computational Data and Social Networks – CSoNet*, 171-183. Available at: https://doi.org/10.1007/978-3-030-91434-9_16.

Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G., & Roli, F. (2013). Evasion Attacks against Machine Learning at Test Time. M*achine Learning and Knowledge Discovery in Databases, European Conference, ECML PKDD, Part III, Vol. 8190, LNCS,* 387–402, Springer. Available at: https://doi.org/10.1007/978-3-642-40994-3_25.

Gandotra, E., Bansal, D., & Sofat, S. (2014). Malware Analysis and Classification: A Survey. *Journal of Information Security*, *Vol. 5, Issue 2*, 56-64. DOI:10.4236/jis.2014.52006.

Kolosnjaji, B., Zarras, A., Webster, G., & Eckert, C. (2018). Deep Learning for Classification of Malware System Call Sequences. *29th Australasian Joint Conference on Artificial Intelligence,* 137-153. DOI:10.1007/978-3-319-50127-7_11.

Mohammed, D., Omar, M., & Nguyen, V. (2018). Wireless Sensor Network Security: Approaches to Detecting and Avoiding Wormhole Attacks. *Journal of Research in Business, Economics and Management, Vol. 10, Issue 2,* 1860-1864. Available at: http://www.scitecresearch.com/journals/index.php/jrbem/article/view/1413.

Banisakher, M., Omar, M., Hong, S., & Adams, J. (2020). A Human Centric Approach to Data Fusion in Post-Disaster Management. *Journal of Business and Management Sciencies, Vol. 8, Issue 1*, 12-20. DOI: 10.12691/jbms-8-1-3.

Nguyen, V., Mohammed, D., Omar, M., & Dean, P. (2020). Net Neutrality Around the Globe: A Survey. *3rd International Conference on Information and Computer technologies – ICICT*. DOI:10.1109/ICICT50521.2020.00083.

Dawson, M., Omar, M., Abramson, J., & Bessette, D. (2014). *Information Security in Diverse Computing Environments, Chapter 9: The Future of National and International Security on the Internet*, 149-178. DOI: 10.4018/978-1-4666-6158-5.

Zangana, H.M., & Omar, M. (2020). Threats, Attacks, and Mitigations of Smartphone Security. *Academic Journal of Nawroz University, Vol. 9, Issue 4,* 324-332. DOI: https://doi.org/10.25007/ajnu.v9n4a989.

Omar, M. (2021a). New insights into database security: An effective and integrated approach for applying access control mechanisms and cryptographic concepts in Microsoft Access environments. *MWAIS 2021 Proceedings, 20*. Available at: https://aisel.aisnet.org/mwais2021/20.

Omar, M. (2021b). Developing Cybersecurity Education Capabilities at Iraqi Universities. *MWAIS 2021 Proceedings, 10.* Available at: https://aisel.aisnet.org/mwais2021/10.

Omar, M. (2022). *Machine Learning for Cybersecurity: Innovative Deep Learning Solutions, Chapter 1: Application of Machine Learning (ML) to Address Cybersecurity Threats*. New York: Springer International Publishing.

Omar, M., Choi, S., Nyang, D., & Mohaisen, D. (2022). Robust Natural Language Processing: Recent Advances, Challenges, and Future Directions. *IEEE Access, Computer Science, Linguistics, Vol. 10*. DOI: 10.1109/ACCESS.2022.3197769.

Omar, M., Gouveia, L.B., Al-Karaki, J., & Mohammed, D. (2022). Reverse-Engineering Malware. *Cybersecurity Capabilities in Developing Nations and Its Impact on Global Security,* 194-217. DOI:10.4018/978-1-7998-8693-8.ch010.