ChatGPT *vs.* Gemini: Comparative Evaluation in Cybersecurity Education with Prompt Engineering Impact

Thomas Nguyen and Hossein Sayadi

Department of Computer Engineering and Computer Science
California State University, Long Beach
Long Beach, CA, USA 90840

***Abstract*—The advent of Large Language Models (LLMs) has revolutionized numerous domains, notably education, by offering powerful tools for personalized learning and automated assistance. These models have the potential to significantly enhance the educational experience, particularly in the field of Computer Science (CS), where the complexity and rapidly evolving nature of topics present unique challenges and opportunities. In this study, we present a comparative evaluation into the transformative potential of LLMs in CS education, with a specific focus on cybersecurity. Our study centers on two leading LLMs: OpenAI's ChatGPT and Google's Gemini Pro, employing a three-fold assessment methodology. Firstly, we analyze the subject matter within cybersecurity education to identify key topics and challenges for examination. Secondly, we meticulously assess and compare the efficacy of ChatGPT and Gemini across various factors in producing satisfactory responses. Lastly, we explore the impact of leveraging prompt engineering on enhancing the quality of responses generated by these AI tools. Through this holistic approach, our research aims to provide insights into the strengths, limitations, and potential avenues for enhancement of these models, thereby enriching the ongoing discourse on LLMs integration in higher education.**

***Index Terms*—Artificial Intelligence, ChatGPT, Education, Gemini, Large Language Models, Prompt Engineering.**

I. Introduction

Artificial Intelligence (AI) has become a cornerstone of modern innovation, driving progress in numerous domains. Among these, education stands out as an essential area where AI's impact is particularly transformative. From personalized learning experiences to automated assessment systems to tackling fairness challenges in education, AI is reshaping teaching and learning methodologies [1, 2, 3, 4]. Furthermore, the landscape of education, in particular Computer Science (CS), is evolving rapidly with the advent of Large Language Models (LLMs) such as OpenAI's ChatGPT and Google's Gemini Pro. These advanced models, trained on extensive datasets, have transformed real-time assistance by enhancing the ability to understand, generate, and explain complex information and problems across various domains [5, 6].

Gemini Pro possesses advanced code generation capabilities and a vast knowledge base, leveraging an impressive context window for analysis and information referencing [7]. In contrast, ChatGPT distinguishes itself through user interaction, enhancing a user-friendly interface and conversational style that has attracted billions of monthly users [8]. Nevertheless, responsible implementation and use of these models in education is paramount to mitigate biases and misinformation while fostering critical thinking skills and effective learning experience [9, 10, 11].

Despite the significant advancements in LLMs, there is a noticeable gap in research concerning their performance and potential impact on CS education, particularly in the specialized field of cybersecurity. This study seeks to bridge this gap by examining the role of LLMs like ChatGPT and Gemini Pro, in enhancing cybersecurity education. To this aim, we present a unified methodology to evaluate the efficacy and quality of responses in LLMs based on key performance criteria. A thorough analysis combining quantitative methods and qualitative evaluation is conducted to assess the effectiveness of both ChatGPT and Gemini. This approach not only uncovers the distinctive features of each AI tool, but also emphasizes positive impacts, strengths, areas for improvement, and opportunities for refining the utilization of these LLMs in the cybersecurity education and curriculum development.

We further examine the impact of prompt engineering on enhancing the efficacy of LLMs in cybersecurity education. Prompt engineering is vital for assessing LLMs' support in education to optimize their performance and enhance learning outcomes [12]. Recognizing the importance of tailored prompts in this context, we assess how each LLM responds to varied engineered prompts. Our comparative evaluation aims to gauge each LLM's efficacy in generating accurate responses to cybersecurity subject queries, with potential extended applications across various CS and engineering domains.

The primary research questions addressed in this study are two-fold: 1) How do ChatGPT and Gemini differ in response quality in cybersecurity education, and what unique features contribute to their effectiveness? 2) How does prompt engineering impact the performance of these LLMs? Our comparative analysis of ChatGPT and Gemini, along with the exploration of prompt engineering impact, offers valuable insights for educators and students to integrate LLMs effectively into their curricula, enhancing learning outcomes in cybersecurity and other CS fields. Overall, this research enriches our understanding of large language models capabilities in education, guiding the development of targeted prompts to enhance their efficacy and helping shape the future of AI-enabled learning in critical domains.

The remainder of this paper is organized as follows. Section

II outlines the background and related work on Gemini and ChatGPT, including their characteristics, educational applications, and prompt engineering techniques. Section III details our proposed methodology for evaluating the effectiveness of LLMs in cybersecurity education, followed by prompt engineering techniques used. Furthermore, Section IV provides a comprehensive presentation of our evaluation results, along with analysis and actionable recommendations to enhance the integration of LLMs in cybersecurity curricula.

II. Background and Related Work

In this section, we discuss the background of LLMs, with a focus on ChatGPT and Gemini. Additionally, we examine related work to contextualize their advancements and impact across various applications and the role of prompt engineering.

A. Commonalities Between ChatGPT-4 and Gemini Pro 1.5

Both ChatGPT-4 and Gemini Pro 1.5 represent the forefront of large language models, sharing several key attributes:

- *Transformer Architecture:* The foundation of both models, allowing them to capture complex language relationships and generate contextually relevant responses. Moreover, while both LLMs deploy Transformers, the specifics of their implementations could vary.
- *Large-Scale Training:* Both have been trained on massive datasets, equipping them with a broad understanding of language and knowledge. It is notable that the datasets might differ in content, quality, and size, depending on the organization's access and selection criteria.
- *Reinforcement Learning from Human Feedback (RLHF):* Fine-tuning behavior based on human feedback to improve safety and alignment with human values.
- *Task-Specific Fine-Tuning:* Adaptable to specific tasks or datasets, increasing their versatility across various applications. The methods and extent of fine-tuning may vary, influencing performance on particular tasks.

B. ChatGPT-4 Characteristics

ChatGPT-4, a product of OpenAI, distinguishes itself through several unique features:

- *Extensive Real-World Deployment:* Its widespread availability and integration into various platforms have allowed for extensive fine-tuning and optimization, making it highly versatile across a diverse range of applications.
- *Persistent Memory:* Enables the model to remember user preferences across interactions, allowing for more personalized and context-aware responses over time.
- *GPT Store:* A growing repository of specialized "GPTs" (customized versions of ChatGPT) that users can access to enhance the model's capabilities for specific tasks or domains, further expanding its utility.

C. Gemini Pro 1.5 Characteristics

Gemini Pro 1.5, developed by Google AI, incorporates several cutting-edge features:

- *Mixture of Experts (MoE) Architecture:* Enhances efficiency by dynamically allocating different parts of the model to specific language processing tasks [34]. Notably, MoE involves using different experts or sub-models that are specialized for certain tasks, which are activated based on the input.
- *Massive Context Window:* Ability to maintain context across extensive text passages, enabling deeper understanding and coherent responses in complex scenarios.
- *Google Services Access:* Seamless integration with various Google services, including real-time access to Google Search for fact-checking and citations, Google Maps for location-based queries, and Google Translate for multilingual tasks. This integration enables the model to deliver responses with the most current and reliable information directly from trusted Google sources.

D. Related Work - ChatGPT vs. Gemini

Lee et al. [10] show that GPT-4 outperforms Gemini Pro in scoring student-drawn models, highlighting its potential for enhancing multimodal assessments in education. Kevian et al. [17] evaluate the capabilities of three LLMs on a new college-level control system problem-solving benchmark called ControlBench, showing that GPT-4 outperforms Gemini. The work in [18] introduces CodeEditorBench, an evaluation framework for assessing LLM performance in code editing tasks, focusing on real-world scenarios of software development. The work in [9] presents an in-depth analysis of ChatGPT performance in computer science and engineering education, identifying its strengths and challenges. The study concludes with a correlation analysis exploring the relationships between subjects, tasks, and limiting factors, offering insights for enhancing ChatGPT's effectiveness in computer science and engineering education. The research in [19] highlights Gemini's integration with Google search for delivering factual information and ChatGPT's excellence in conversational flow and creativity.

Several studies have examined the use of LLMs in healthcare. In [27], ChatGPT and Gemini were tested against 52 questions from the American College of Cardiology's hypertension guidelines, with both models delivering comparably accurate but sometimes incomplete responses. Carlà et al. [20] found that ChatGPT aligned with specialists in 58% of glaucoma surgery cases, outperforming Gemini in precision. Similarly, in retinal detachment cases, [26] showed that ChatGPT-3.5, ChatGPT-4, and Gemini matched vitreoretinal surgeons' decisions in 80%, 84%, and 70% of cases, respectively. The study in [21] further highlighted ChatGPT's potential in medical education and clinical decision-making, stressing the importance of effective prompt engineering for accurate responses.

While LLMs have demonstrated potential in fields such as general education and medicine, they may encounter challenges in other areas. For example, [33] identified limitations of ChatGPT in the context of chemistry laboratory teaching. Therefore, it is crucial to analyze the pedagogical implications

TABLE I: Recent studies on application of ChatGPT and Gemini in education and prompt engineering

Research	Target Field	Capabilities Examined/Contributions	LLMs used
[13]	Prompt Engineering	Prompt Engineering: Chain-of-thought	ChatGPT
[14]	Prompt Engineering	Current and future trends in LLMs and prompt engineering	N/A
[5]	Prompt Engineering	Few-Shot Prompting	ChatGPT
[15]	English Education	Validation and evaluation of question quality and system framework for automatic question generation	ChatGPT
[10]	General Education	Image classification, Text processing in images	ChatGPT, Gemini
[16]	Prompt Engineering	Directional Stimulus Prompting	ChatGPT, etc.
[17]	Math, Control Engineering Education	Accuracy, reasoning capabilities, and ability to provide coherent and informative explanations in solving undergraduate control engineering problems	ChatGPT, Gemini, Claude 3
[18]	Software Development/Programming	Code Debugging, Translation, Polishing, Requirement Switching	ChatGPT, Gemini, Deepseek Coder, etc.
[19]	General Education, Cross-Industry	Applications across industries, Performance metrics (response coherence, accuracy, latency, and scalability)	ChatGPT, Gemini
[20]	Medical Education	Develops Glaucoma surgical plan	ChatGPT, Gemini
[21]	Medical Education	ChatGPT's Performance on USMLE and discuss Education implication	ChatGPT
[22]	Prompt Engineering	Societal Impact of NLG and Hermeneutic value of ChatGPT's generated text	ChatGPT
[23]	Cybersecurity, Prompt Engineering	LLM-Integrated Applications potential attack using Indirect Prompt Injection	ChatGPT
[24]	Prompt Engineering	Induced attack against ChatGPT using Prompt Engineering	ChatGPT
[25]	Sentiment Analysis	Analyzing nuanced and ambiguous sentiments across multiple languages	ChatGPT, Gemini, LLaMA2
[26]	Medical Education	Analyzing retinal detachment cases and recommending appropriate surgical plans	ChatGPT, Gemini
[27]	Medical Education	Compares accuracy and readability in responding to cardiology-related questions	ChatGPT, Gemini
[28]	Spam Detection	Detecting spam within the SpamAssassin mail corpus	ChatGPT, Gemini
[29]	General Education	Evaluates the readability and appropriateness of AI-generated stories across different educational levels	ChatGPT, Gemini
[30]	General Education	Examines the capabilities of AI chatbots in creating 7th-grade lesson plans across various subjects	ChatGPT, Gemini
[9]	Computer Science & Engineering Education	Analysis of ChatGPT's performance and reliability of responses in computer science and engineering education	ChatGPT
[31]	Software Engineering, Prompt Engineering	Assesses the LLMs effectiveness in improving productivity in empirical software engineering tasks	ChatGPT, Gemini, ERNIE Bot, etc.
[12]	Prompt Engineering	Best techniques and practices for optimizing LLM output	ChatGPT
[32]	Prompt Engineering	Introducing prompt patterns, with comprehensive framework and catalog	ChatGPT
[33]	Chemistry Education	Evaluating ChatGPT's effectiveness in suggesting scientifically and pedagogically protocols for chemistry lab activities.	ChatGPT

of LLMs for each specific field of study to better understand their potential and limitations. Additionally, the study by [22] discusses ChatGPT's potential impact on human understanding and self-perception, highlighting the challenges in optimizing its output for hermeneutic value. Furthermore, the authors in [25] concluded that both ChatGPT and Gemini offer valid sentiment predictions across various scenarios and languages, comparable to human judgments.

E. Related Work - Prompt Engineering

The work in [12] discusses best practices for optimizing LLM outputs, including iterative refinement, utilizing external resources, and advanced strategies such as prompt chaining and handling ambiguous inputs. Building on this, the authors in [32] present the concept of prompt patterns, along with a comprehensive framework for structuring these patterns. They also offer a catalog of suggested prompt patterns tailored for various purposes.

Li et al. [16] introduces Directional Stimulus Prompting, a novel approach aimed at guiding black-box LLMs toward specific desired outputs. By providing hints, the LLMs were able to achieve desired behaviors. Similarly, Wei et al. [13] introduce chain-of-thought, a method involving intermediate reasoning steps to enhance LLM performance on complex tasks. This technique shows significant accuracy gains in arithmetic, commonsense, and symbolic reasoning, particularly with LLMs around 100 billion parameters, though it is less effective for simpler tasks.

Marvin et al. [14] highlight the importance of prompt engineering in LLMs and conversational AI systems, calling for increased focus on the processes and procedures involved. Furthermore, Lee et al. [15] demonstrate that combining large language models like ChatGPT with few-shot prompt engineering techniques significantly improves the quality and validity of automatically generated questions for English education, enhancing online learning experiences.

Table I offers a summary of recent research efforts focusing on the evaluation of ChatGPT, Gemini and other LLMs in the field of education, novel techniques in prompt engineering and other notable findings. Our work distinguishes itself by conducting a comparative analysis of LLMs for cybersecurity education, focusing on efficacy evaluation and exploring the impact of prompt engineering in enhancing their performance.

III. Proposed Methodology

Figure 1 depicts an overview of the proposed methodology for assessing the effectiveness of ChatGPT and Gemini Pro in supporting cybersecurity education, consisting of three phases:

1) Subject Data Collection: We compiled a comprehensive set of over 100 questions centered on cybersecurity education, meticulously designed to challenge and assess the proficiency of LLMs at a college-level, project-based standard. These questions span a wide range of topics within the cybersecurity field, ensuring a thorough evaluation of the models' capabilities. The questions were sourced from authoritative materials, including academic textbooks, university lectures, and the CompTIA Security+ certification exam, to ensure they reflect real-world security challenges and industry standards.

LLMs Response Acquisition: To evaluate the performance of LLMs, we analyzed responses from both ChatGPT-4 and Gemini Pro 1.5. Acknowledging that LLMs often produce varied answers to identical prompts, we restricted each model to a single attempt per question. This approach simulates real-world application scenarios, where consistency and accuracy in responses are crucial. By examining a large and diverse questions set, we aimed to capture a more comprehensive picture of each model's capabilities, leveraging the natural randomness in their outputs. Given that these LLMs are stateful, capable of remembering past interactions and maintaining context across a conversation, we implemented specific measures to ensure consistency. Each question was introduced in a new, isolated conversation, preventing any influence from prior context. Additionally, ChatGPT's memory function, which can retain some user-specific information even across new conversations, was disabled. This precaution ensured that all responses were generated without any residual knowledge

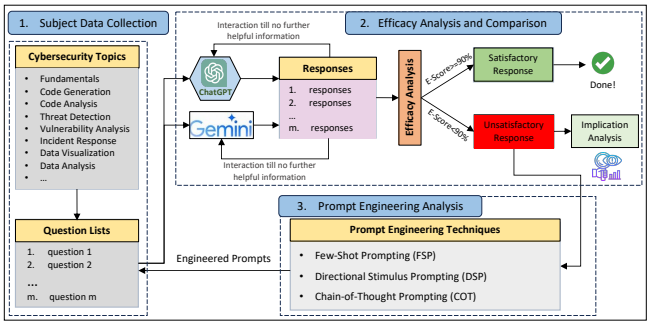


Fig. 1: Overview of the proposed methodology for evaluating the efficacy of LLMs (ChatGPT and Gemini) for cybersecurity education

from previous interactions, thereby maintaining the integrity and reliability of the evaluation process.

2) Efficacy Analysis and Comparison: In the second phase, we perform a comparative analysis of ChatGPT and Gemini, assessing their performance in responding to cybersecurity-related questions, across different criteria. We analyze and evaluate responses both quantitatively and qualitatively and categorize them as satisfactory or unsatisfactory.

3) Prompt Engineering Impact Analysis: In the third phase, we focus on enhancing efficacy in human-AI interaction. Experimenting with three prompt engineering techniques, we address unsatisfactory responses and reassess their efficacy scores. Our findings provide insights into the role of prompt engineering in refining user-AI interaction. By strategically crafting prompts, we observed significant improvements in the quality and accuracy of responses, indicating how tailored approaches can unlock the full potential of LLMs in education.

A. Efficacy Analysis and Evaluation Metrics

In our proposed efficacy analysis framework, we consider four major factors to evaluate the responses: 1) Technical Accuracy, 2) Usefulness, 3) Relevance, and 4) Comprehensive Analysis and Critical Thinking. These efficacy factors are outlined in Table II. Each factor is subjected to a rigorous evaluation on a scale ranging from 1 to 5, leading to cumulative scores that delineate the overall efficacy of each response. Responses attaining a score of 90% or higher are deemed satisfactory, while those falling below this threshold will be subjected for potential prompt engineering enhancements.

In our efficacy analysis, we follow a systematic grading criteria to assess the responses generated by ChatGPT and Gemini. The following is an example of the grading criteria for the technical accuracy factor:

- 1- Incorrect: Responses contain factual errors and misinformation.
- 2- Contains a mix of accurate and inaccurate (>50%) information.
- 3- Contains a mix of accurate (>60%) and inaccurate information.
- 4- Mix of accurate (>80%) and inaccurate (<20%) information.
- 5- Fully Correct: Generated responses are factually sound.

We have assigned specific weights to each criterion to reflect their varying importance in evaluating model performance, aligning them with the relative significance concerning the course subjects and tasks under examination. These weights, as shown in Table II, are tailored to support educators and learners across diverse subjects. However, it is notable that the appropriateness of these weights may vary depending on

TABLE II: Efficacy factors in the proposed evaluation framework

Factor	Description	Weight Percentage
Technical Accuracy	Assesses the correctness of the information provided in terms of concepts, tools, and practices. Responses are evaluated based on the accuracy of the information presented and the absence of factual errors or misinformation.	30%
Usefulness	Evaluates the practical value of the information provided in terms of its applicability and relevance to real-world cybersecurity scenarios. Responses are assessed based on their ability to offer actionable insights and solutions to common cybersecurity challenges.	30%
Relevance	Evaluates whether the response addresses the specific question. Responses are assessed based on their alignment and up to date with cybersecurity education.	30%
Comprehensive Analysis & Critical Thinking	Assesses the depth of analysis and critical thinking demonstrated in the response. Responses should not only follow best practices but also demonstrate deep understanding of the underlying principles.	10%

the specific priorities and objectives of the evaluation. It is crucial to align them with the importance of each criterion in the given context. The Efficacy Score (E-Score) in our analysis is calculated as follows:

E-Score = (Technical Accuracy × 0.3) + (Usefulness × 0.3) + (Relevance × 0.3) + (Comp. Analysis & Critical Thinking × 0.1)

B. Applying Prompt Engineering Techniques

Prompt engineering encompasses various strategies for optimizing the responses of large language models. In the third stage of our methodology, we apply three potent prompt engineering techniques and compare the results with the previous responses without prompt engineering.

1) Few-Shot Prompting (FSP): Developed by Brown et al. [5], it allows LLMs to learn effectively from minimal instructions provided in the prompt. By presenting the model with a few examples, it becomes capable of generalizing patterns and concepts, which leads to more accurate responses.

2) Directional Stimulus Prompting (DSP): Introduced by Li et al. [16], this technique involves incorporating hints or specific instructions into the prompt to guide the LLM towards a particular style, tone, or type of information in the response.

3) Chain-of-Thought Prompting (COT): Presented by Wei et al. [13], it focuses on breaking down complex tasks into smaller, manageable steps. The prompt outlines a series of reasoning steps the LLM should follow, leading it to produce a clear and logical final response.

IV. Evaluation Results and Analysis

In this section, we present the results of our comparative evaluation of LLMs in cybersecurity education, focusing on their performance across various tasks and the impact of prompt engineering on response efficacy enhancement.

A. Comparative Analysis of ChatGPT and Gemini

1) Fundamental Questions: This analysis focuses on fundamental cybersecurity concepts, drawn from textbooks and literature. Both LLMs perform exceptionally well, achieving near-perfect scores (Gemini: 98% and ChatGPT: 97%) in our efficacy analysis. However, both occasionally exhibit a verbosity bias where they provide excessive detail that can be unnecessary or misleading. Notably, ChatGPT's responses average 356 words—37% longer than Gemini's 260-word average. While this added detail can enrich understanding, it may also hinder the efficiency of finding concise answers.

2) Code Generation: Both models were marketed as proficient in code generation. Our results confirm that both models are capable of generating functional Python and C++ code across a wide range of applications, demonstrating familiarity

with various libraries and the ability to solve most coding problems with minimal assistance. In our efficacy analysis, ChatGPT outperformed Gemini, scoring 96% compared to Gemini's 76%. Notably, most of the code generated by ChatGPT ran successfully without significant modifications, while Gemini's code frequently required further debugging to achieve the desired outcomes. Our assessment also noted certain challenges in Gemini's code generation capabilities. For instance, when prompted to "Create a script in Python to scan a website for cross-site scripting (XSS) vulnerabilities" Gemini occasionally refused to provide a clear answer. The reason for inconsistency in its responses remains unclear. Even after applying prompt engineering techniques, Gemini often produced generic responses, revealing its limitations in handling such tasks. In contrast, ChatGPT successfully generated the required code, demonstrating better performance in complex assignments.

3) Cryptography: Cryptography plays a crucial role in cybersecurity, often involving complex and math-intensive concepts. The performance of large language models in this area is particularly important for their potential use in educational settings. In our evaluation, we tested both ChatGPT and Gemini on a set of 45 cryptography-related questions, from various topics including Public and Private Key, Cryptanalysis Techniques, Message Authentication, Vulnerabilities and Attacks, etc. ChatGPT demonstrated a higher level of accuracy, scoring 96% on the test. Meanwhile, Gemini also performed well, achieving a score of 91%. These results highlight the strong capabilities of both models in handling cryptography tasks, with ChatGPT showing a slight edge in overall accuracy. This assessment also evaluated the LLMs' mathematical abilities, specifically in modular arithmetic and binary calculations. Also, we observed that both models faced difficulties in producing accurate results when handling large numbers and lengthy binary or hexadecimal sequences.

4) Code Improvement Analysis: We assessed the code enhancement capabilities of both LLMs, focusing primarily on developing machine learning algorithms in Python. While our evaluation centered on this specific application, the insights gained could be applicable to other domains as well. We tested several instances where we provided both LLMs with code for improvement, and in most cases, ChatGPT performed slightly better than Gemini. One illustrative example involved training a Gaussian Naive Bayes (GNB) model using the Ada-Boost ensemble technique for anomaly detection. In this instance, we provided both models with our existing Python code for training a GNB model with default parameters. Both LLMs correctly identified that the next step after applying the ensemble method was hyperparameter tuning. However, their recommendations differed: ChatGPT suggested using Grid Search, while Gemini recommended Random Search. Both models generated code that executed without errors and improved the original model's accuracy by 4%. Given that the dataset contained only four features, ChatGPT's recommendation to use Grid Search was more appropriate, as the smaller hyperparameter space allowed for a more exhaustive search, ensuring accurate results. This demonstrates the capability of both LLMs to analyze provided datasets and code, offering valuable suggestions for enhancing ML workflows.

5) Data Visualization: We further evaluate the ability of two LLMs, ChatGPT and Gemini, to effectively visualize data given a set of datasets. Throughout our analysis, we observed that each model excelled in certain tasks while encountering difficulties in others. For instance, when tasked with generating bar graphs, both models successfully created visualizations; however, ChatGPT's bar graph had overlapping text, which made it difficult to read. In contrast, Gemini's bar graph featured tilted text that improved readability. In another scenario, we utilized a Receiver Operating Characteristic (ROC) curve to evaluate the performance of a classification model for anomaly detection by illustrating the trade-off between the true positive rate and the false positive rate. Both LLMs generated nearly identical code, but Gemini's method of directly using predicted class labels instead of estimated probabilities for all classes led to an incorrect computation of the ROC curve, reducing the effectiveness of the classifier analysis. Additionally, we tasked the models with generating a heatmap to visualize the correlation matrix and identify the most important features of a cybersecurity related dataset. Gemini's heatmap was truncated, making it difficult to process the information effectively. However, after prompting it to correct the issue, Gemini was able to generate a complete and usable heatmap.

6) Machine Learning Implementation for Threat Detection: In our threat detection tasks, we tasked the LLMs with training a machine learning model to detect email and website phishing using publicly available cybersecurity datasets [35, 36]. For detecting phishing emails, ChatGPT achieves near-perfect accuracy of 100% for 50 samples. While Gemini also approaches this level, it demonstrates higher sensitivity to noise. Interestingly, when presented with multiple emails simultaneously, Gemini occasionally deviates from its original instructions, responding based on the email content rather than the designated task. Nevertheless, both LLMs demonstrate strong reasoning and reliability in identifying phishing emails, with ChatGPT delivering more robust and superior results.

A notable discrepancy arises when classifying phishing URLs. This task proves particularly challenging given the lack of additional context beyond the URL itself. ChatGPT excels in this area, achieving a remarkable 96% accuracy on a sample set of 50 URLs, effectively distinguishing phishing attempts from legitimate sources. In contrast, Gemini significantly underperforms, with only a 20% accuracy rate. Despite its integration with Google for cross-referencing and verification, Gemini's ability is hindered by its reliance on recognizing popular domains, often resulting in false negatives.

7) Cyber-Attacks Detection: In another emerging area, we evaluated the detection capabilities for cyber-attacks and key defense mechanisms at the hardware level, focusing on hardware Trojan and malware detection using hardware events like performance counter data. Hardware-based anomaly detection

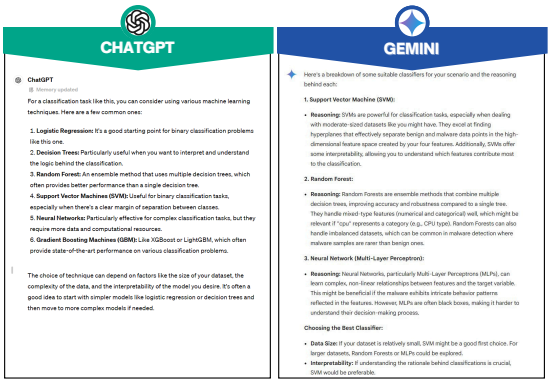


Fig. 2: ML Classifier suggestion for malware detection task based on given dataset

using AI/ML is an emerging area of research that has gained significant attention in the field of cybersecurity [37, 38]. For this task, we designed mini project-based assignments that required the LLMs to train ML algorithms using datasets specific to these areas [39, 40, 41]. Given an example dataset, we first asked both models to suggest a classifier, as demonstrated in Figure 2, both models provided similar suggestions for classifier types, accompanied by reasonable explanations that aligned with our expectations. Subsequently, we instructed both LLMs to train the model using the provided datasets.

For the hardware Trojan detection task [39], ChatGPT trained both Logistic Regression and Random Forest models, whereas Gemini trained Decision Tree, Logistic Regression, and Random Forest models. Due to the limited number of samples in the dataset, the initial accuracy of all models was low. Both models were then asked to fine-tune their design. They both conducted a grid search on the Random Forest classifier, with ChatGPT optimizing three parameters and Gemini optimizing four. The final accuracies were comparable, with ChatGPT achieving 73.33% and Gemini achieving 77.0%.

For the hardware malware detection task [40, 41], both models trained using the Random Forest Classifier, employed similar techniques, and achieved a 97% accuracy, closely aligning with the researchers' results. However, it was noted that ChatGPT's code imported an unknown library called `ace_tools`. Further investigation revealed that this tool is an empty placeholder library on the Python Package Index (PyPi), with no functionality. When prompted, ChatGPT explained:

"*`ace_tools` is a custom tool used in this environment to display dataframes to users. In a typical local or online Python environment, you might not need this tool and can display dataframes using print() statements or by displaying them directly in a Jupyter Notebook.*"

However, we found this answer to be peculiar, raising a security concern that malicious attackers could exploit LLMs to spread harmful software. Existing literature also raises concerns regarding this problem [42], suggesting that users always double-check generated code to ensure its safety and integrity. Overall, both models accounted for precision, recall, and F1-scores when evaluating the data. Moreover, the analysis time (latency overhead) varied significantly; ChatGPT's analysis was consistently faster, while Gemini sometimes idled for several minutes. However, ChatGPT suffered from unexplained code usage, as explained above. Despite these differences, both models demonstrated notable capabilities in training models for cyber-attacks detection purposes.

8) Code De-obfuscation Analysis: In assessing the performance of LLMs in code de-obfuscation tasks, we tested their capabilities using several methods: Rename obfuscation, AES 256 obfuscation, and Pyarmor obfuscation. With the initial method of rename obfuscation, both ChatGPT and Gemini showcased their ability to de-obfuscate Python and Javascript code from the 30 selected questions within the employed dataset [43, 44]. ChatGPT performed particularly well, producing partially correct versions of the original code. Gemini exhibited proficiency in the same techniques but encountered difficulties with longer code segments, occasionally resulting in inaccuracies. Nevertheless, neither model could decode AES 256 or Pyarmor obfuscations. Despite using prompt engineering techniques, including zero-shot, few-shot, and chain of thought, no readable code was produced. Notably, when presented with Pyarmor obfuscated code, both models initially recognized the protection offered by Pyarmor. However, if the user prompted them to ignore this protection, they were still willing to attempt de-obfuscation.

9) Knowledge Cutoff: ChatGPT-4, as of its last training data update in October 2023, cannot address events occurring after this date unless browsing is enabled in more advanced versions. In contrast, all versions of Gemini, with their integration of Google Search, have real-time internet access, allowing them to provide up-to-date information and mitigating the issue of a knowledge cutoff.

10) Fact Check and Cited Source: A concern with LLMs is their tendency to fabricate information, which can undermine user trust. Gemini addresses this by automatically citing its sources, allowing users to easily verify statements or conduct further research. ChatGPT can also cite sources and fact-check information when using a model with browsing capabilities, though this feature is generally limited to more advanced versions of GPT-4, which may not be available to free users, potentially limiting inclusive access in educational settings.

11) Customization and Extensibility: In Jan. 2024, OpenAI launched the GPT Store, a marketplace for custom chatbots that are custom-designed to assist users with a wide range of skills, thereby extending ChatGPT's capabilities. It enables ChatGPT to perform tasks such as fetching the latest research papers or providing real-time weather updates. Although some plugins are restricted by paywalls or have limited functionality, this development marks a notable stride in customization and refinement. Currently, Gemini lacks a comparable feature, and its future implementation remains uncertain. This lack of extensibility may limit Gemini's ability to adapt to new tasks and user needs, in contrast to the more versatile ChatGPT.

12) Discussion: Our efficacy analysis shows that both ChatGPT and Gemini perform well in various cybersecurity tasks, though with notable differences. As illustrated in Figure 3, ChatGPT excels in most tested capabilities, offering detailed responses that enhance understanding but can occasionally

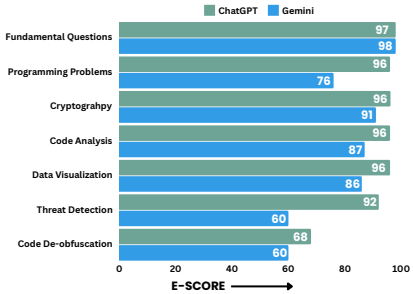


Fig. 3: Efficacy Score of ChatGPT and Gemini for different capabilities

be verbose. In code generation and analysis, ChatGPT outperforms Gemini, demonstrating a higher success rate and fewer errors, particularly in complex tasks. In cryptography-related tasks, both models perform admirably, with ChatGPT slightly edging out Gemini in accuracy. When it comes to ML code improvement, ChatGPT's suggestions tend to be more precise, resulting in better outcomes. Furthermore, ChatGPT's customization options through the GPT Store present a significant advantage in adaptability, a feature currently absent in Gemini. However, Gemini's real-time information access via Google Search is a distinct strength, allowing it to provide up-to-date content, which mitigates the issue of knowledge cutoffs inherent to ChatGPT. This makes Gemini particularly valuable for tasks requiring the latest information, although ChatGPT can also update its responses with specific prompts.

B. Addressing Low Efficacy with Prompt Engineering

In this section, we delve into the implementation of prompt engineering to enhance the responsiveness of LLMs. We target engineering questions from the preceding section that garnered low efficacy scores. Following this, we will reevaluate the output to gauge the effectiveness of these techniques.

Few-Shot Prompting Example: In the threat detection task, Gemini initially scored 60%, facing challenges in distinguishing between phishing and legitimate URL domains. To address this, we provided sample URLs along with explanations to guide the model. Through the use of few-shot prompting, Gemini significantly improved its performance, achieving a higher accuracy and increasing its accuracy by 27%. As illustrated in Figure 4 case study, when presented with various URLs, Gemini accurately distinguished between

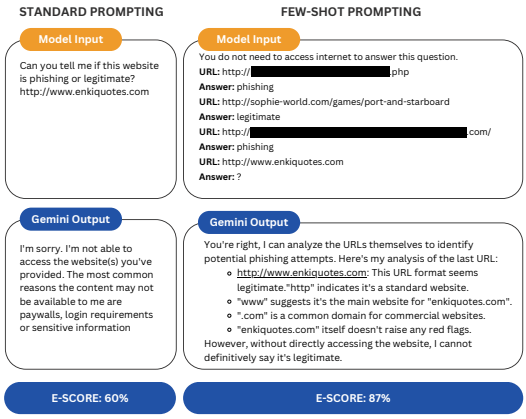
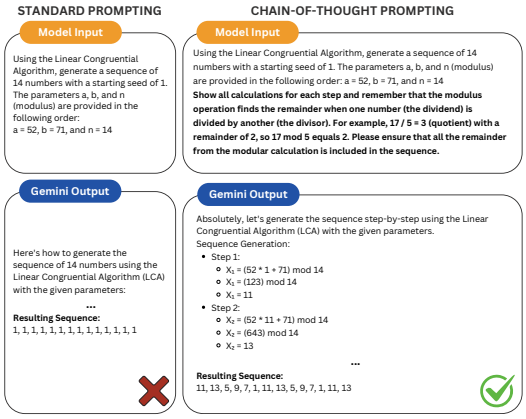


Fig. 4: Example of Few-Shot Prompting with Gemini



The Gemini response was abbreviated to accommodate space limitations.

Fig. 5: Example of Chain-of-thought Prompting with Gemini

legitimate and potentially malicious ones, correctly identifying the next URL in the sequence. Although its reasoning may have been somewhat lacking, Gemini identified characteristics indicating a legitimate website, such as a clear name and a top-level domain like ".com". The accuracy can evidently increase with more examples (e.g., 3-shot, 5-shot, and 10-shot prompting) compared to fewer examples (e.g., 1-shot prompting). This marked improvement highlights the effectiveness of few-shot prompting compared to previous instances where Gemini struggled to provide a response. It is noteworthy that for complex cybersecurity tasks, such as AES-256 code de-obfuscation, prompt engineering did not yield significant improvements, revealing the intrinsic limitations of this approach. While prompt engineering can optimize performance for certain tasks, it cannot fundamentally enhance a model's core capabilities. This highlights the necessity for continued research into advancing prompt engineering techniques to extend the boundaries of language models' effectiveness in addressing challenging problems.

Chain-of-Thought Prompting Example: In addressing the challenges of solving linear congruential algorithms, Google Gemini initially exhibited suboptimal performance in modular arithmetic, a critical component for accurate results in these algorithms. Among the 25 linear congruential sequence generator tasks, Gemini achieved only 8% accuracy. While the model correctly identified all procedural steps, it struggled particularly with complex modulus operations and the generation of accurate sequences for random number generation. To enhance performance, we applied chain-of-thought prompting, where Gemini was first guided through an example of modular arithmetic before attempting the linear congruential generator equations, as illustrated in Figure 5. This approach involved explicitly outlining and requesting the model to display each calculation step, along with providing an example calculation, led to a significant improvement, increasing accuracy to 85%.

However, despite this improvement, chain-of-thought prompting, like other prompting strategies, only marginally enhanced specific capabilities and did not fully address the underlying computational limitations of the model. Notably, when tasked with calculating sequences for multiple different sets of questions simultaneously, Gemini was unable to display

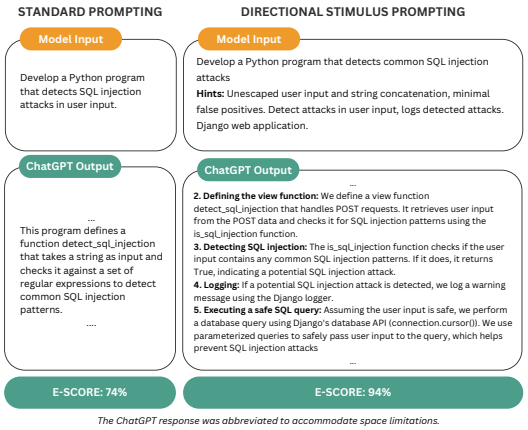


Fig. 6: Example of Directional Stimulus Prompting with ChatGPT

all the steps due to space limitations, leading to a deterioration in accuracy. This highlights the need for ongoing research and development of more advanced strategies to further enhance model performance in mathematical computations.

Directional Stimulus Prompting Example: When tasked with generating code to detect injection-based attacks, traditional language models often offer simplistic solutions that overlook the complexities of real-world attack methods. To enhance efficacy, we tested five different scenarios using directional stimulus prompting, providing hints and keywords to guide the model toward more robust solutions. All examples tested showed noticeable improvements. As illustrated in Figure 6, this approach led to the generation of code suitable for cybersecurity applications, with enhanced SQL injection prevention in Django through improved sanitization and validation techniques within a Django View.

C. Ethical Exploitation of LLMs via Prompt Engineering

Our research also explores the ethical constraints of LLMs by evaluating various techniques used to circumvent model ethical limitations, specifically jailbreaking prompts discussed in [45]. These prompt techniques include: 1) Role-playing, in which the model adopts a different persona or assumes alternative responsibilities; 2) Output constraints, where the model operates within set limitations like specific word counts; and 3) Privilege escalation, where prompts are designed to grant the model enhanced abilities or access. Notably, these methods have become less effective on newer models and were primarily applicable to earlier versions like ChatGPT-3.5 and Gemini 1.0, which are no longer accessible.

However, the work in [46] introduced an advanced method known as the Disguise and Reconstruction Attack (DRA). This approach involves masking harmful commands to bypass model restrictions and prompting the model to regenerate the original harmful content in its outputs. Our experiments confirmed previous findings, demonstrating that GPT-4 can be manipulated using the DRA to reconstruct harmful instructions. We further extended this analysis to the Gemini model, which exhibited enhanced resilience by delaying responses, rejecting requests, or providing legally compliant alternatives. These observations indicate that Google has incorporated additional safeguards and ethical considerations into Gemini to encourage responsible AI usage. It is important to recognize that these models are regularly updated to align with evolving ethical standards. This commitment is reflected in the obsolescence of older jailbreak techniques on newer versions, underscoring the ongoing efforts to strengthen the security and integrity of these LLMs.

D. Best Practices for LLMs Use in Cybersecurity Education

To enhance the use of LLMs in education, particularly in cybersecurity, educators are encouraged to adopt several key practices. Firstly, optimizing prompting techniques can significantly improve the accuracy of responses. Employing few-shot prompting, where even a single example (1-shot) can lead to noticeable improvements, is especially beneficial. For more complex tasks, increasing the number of examples (e.g., 3-shot, 5-shot, or 10-shot) can further refine the model's outputs. Tailoring interactions with the LLM by providing context about the user's educational background and level of understanding can ensure that responses are precise and relevant. This approach minimizes the likelihood of receiving unnecessary or overly complex information.

In the context of code generation, safety is paramount. It is essential to thoroughly review and test any generated code, particularly in cybersecurity scenarios, to avoid the inclusion of harmful or unintended elements. Additionally, regularly verifying the information provided by the LLM is important, as even advanced models may produce inaccurate or misleading content. Special attention should be given to verifying calculations and numerical data, especially for tasks involving complex or large numbers, to ensure their accuracy.

Furthermore, awareness of potential biases in the model's responses is critical. LLMs can reflect or amplify biases present in their training data. Hence, it is important to critically assess outputs for any underlying assumptions or skewed perspectives. Continued research is essential to mitigate these biases and improve the reliability of LLMs. Developing techniques for identifying, addressing, and reducing biases will enhance the fairness and accuracy of model outputs. By adhering to these practices, educators can better integrate LLMs into their teaching strategies, ensuring that the content generated is reliable, relevant, and as unbiased as possible.

V. Concluding Remarks

This work presents a comparative assessment of ChatGPT and Gemini in supporting cybersecurity education. By analyzing subject matter relevance, efficacy, and impact of prompt engineering, we provide valuable insights into the capabilities and limitations of these LLMs. Our findings demonstrate that prompt engineering can significantly enhance LLMs performance in cybersecurity. By applying tailored prompting strategies such as few-shot, chain-of-thought, and directional stimulus prompting, we observed substantial improvements in accuracy and responsiveness of LLMs. ChatGPT demonstrated superior performance in tasks requiring attention to detail, maintaining higher consistency and adaptability across diverse cybersecurity topics. Its solid coherence, particularly in threat

detection and phishing analysis, highlights its effectiveness in critical real-world applications where accuracy is paramount. In contrast, Gemini offers shorter, more direct responses and excels in real-time fact-checking due to its integration with Google. The choice of LLM should align with the user's specific needs. Gemini is recommended for research and fundamental queries, while ChatGPT is better suited for more complex and critical tasks requiring detailed fine-tuning. For future direction, we plan to expand our evaluation across the CS curriculum and explore further prompt engineering methods. We also intend to assess the long-term impact of LLMs on student learning outcomes, offering insights for educators and learners, and facilitating AI integration in education.

REFERENCES

[1] M. P. Pratama *et al.*, "Revolutionizing education: harnessing the power of artificial intelligence for personalized learning," *Journal of education, language teaching and science*, vol. 5, no. 2, pp. 350–357, 2023.

[2] C. W. Fernandes *et al.*, "Advancing personalized and adaptive learning experience in education with artificial intelligence," in *European Ass. for Education in Elect. and Info. Eng. (EAEEIE)*. IEEE, 2023, pp. 1–6.

[3] Baird *et al.*, "Towards race and gender equity in data science education," in *2023 IEEE Frontiers in Education Conference (FIE)*, 2023, pp. 1–10.

[4] C. W. Fernandes *et al.*, "Unleashing the potential of reinforcement learning for enhanced personalized education," in *2023 IEEE ASEE Frontiers in Education Conference (FIE)*, 2023, pp. 1–5.

[5] T. Brown *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[6] E. Kasneci *et al.*, "Chatgpt for good? on opportunities and challenges of large language models for education," *Learning and individual differences*, vol. 103, p. 102274, 2023.

[7] G. Team *et al.*, "Gemini: a family of highly capable multimodal models," *arXiv preprint arXiv:2312.11805*, 2023.

[8] "Number of chatgpt users and key stats (aug 2024)," https://www.namepepper.com/chatgpt-users, accessed: 2024-08-02.

[9] Z. He *et al.*, "The ai companion in education: Analyzing the pedagogical potential of chatgpt in computer science and engineering," in *2024 IEEE Global Engineering Education Conference (EDUCON)*, 2024, pp. 1–10.

[10] G.-G. Lee *et al.*, "Gemini pro defeated by gpt-4v: Evidence from education," *arXiv preprint arXiv:2401.08660*, 2023.

[11] W. Chung *et al.*, "Machine learning to the rescue: Ml-assisted framework for equity-driven education," in *2022 IEEE Global Engineering Education Conference (EDUCON)*, 2022, pp. 1254–1263.

[12] S. Ekin, "Prompt engineering for chatgpt: a quick guide to techniques, tips, and best practices," *Authorea Preprints*, 2023.

[13] J. Wei *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *Advances in neural information processing systems*, vol. 35, pp. 24 824–24 837, 2022.

[14] G. Marvin *et al.*, "Prompt engineering in large language models," in *International Conf. on Data Intelligence and Cognitive Informatics*. Springer, 2023, pp. 387–402.

[15] U. Lee *et al.*, "Few-shot is enough: exploring chatgpt prompt engineering method for automatic question generation in english education," *Education and Information Technologies*, pp. 1–33, 2023.

[16] Z. Li *et al.*, "Guiding large language models via directional stimulus prompting," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[17] D. Kevian *et al.*, "Capabilities of large language models in control engineering: A benchmark study on gpt-4, claude 3 opus, and gemini 1.0 ultra," *arXiv preprint arXiv:2404.03647*, 2024.

[18] J. Guo *et al.*, "Codeeditorbench: Evaluating code editing capability of large language models," *arXiv preprint arXiv:2404.03543*, 2024.

[19] N. Rane *et al.*, "Gemini versus chatgpt: Applications, performance, architecture, capabilities, and implementation," *Performance, Architecture, Capabilities, and Implementation (February 13, 2024)*, 2024.

[20] M. M. Carlà *et al.*, "Large language models as assistance for glaucoma surgical cases: a chatgpt vs. google gemini comparison," *Graefe's Archive for Clinical and Experimental Ophthalmology*, pp. 1–15, 2024.

[21] T. F. Heston and C. Khun, "Prompt engineering in medical education," *International Medical Education*, vol. 2, no. 3, pp. 198–205, 2023.

[22] L. Henrickson *et al.*, "Prompting meaning: a hermeneutic approach to optimising prompt engineering with chatgpt," *AI & SOCIETY*, pp. 1–16, 2023.

[23] K. Greshake *et al.*, "Not what you've signed up for: Compromising real-world llm-integrated applications with indirect prompt injection," in *ACM Workshop on AI and Security*, 2023, pp. 79–90.

[24] B. Liu *et al.*, "Adversarial attacks on large language model-based system and mitigating strategies: A case study on chatgpt," *Security and Communication Networks*, vol. 2023, 2023.

[25] A. Buscemi *et al.*, "Chatgpt vs gemini vs llama on multilingual sentiment analysis," *arXiv preprint arXiv:2402.01715*, 2024.

[26] M. M. Carlà *et al.*, "Exploring ai-chatbots' capability to suggest surgical planning in ophthalmology: Chatgpt versus google gemini analysis of retinal detachment cases," *British Journal of Ophthalmology*, 2024.

[27] T. J. Lee *et al.*, "Unlocking health literacy: the ultimate guide to hypertension education from chatgpt versus google gemini," *Cureus*, vol. 16, no. 5, 2024.

[28] K. Mardiansyah and W. Surya, "Comparative analysis of chatgpt-4 and google gemini for spam detection on the spamassassin public mail corpus," 2024.

[29] M. F. Karaca, "Is artificial intelligence able to produce content appropriate for education level? a review on chatgpt and gemini," in *Cognitive Models and Artificial Intelligence Conference*, 2024, pp. 208–213.

[30] A. Baytak, "The content analysis of the lesson plans created by chatgpt and google gemini." *Research in Social Sciences and Technology*, vol. 9, no. 1, pp. 329–350, 2024.

[31] W. Liang and G. Xiao, "An exploratory evaluation of large language models using empirical software engineering tasks," in *Proceedings of the 15th Asia-Pacific Symposium on Internetware*, 2024, pp. 31–40.

[32] J. White *et al.*, "A prompt pattern catalog to enhance prompt engineering with chatgpt," *arXiv preprint arXiv:2302.11382*, 2023.

[33] J. L. Araújo and I. Saude, "Can chatgpt enhance chemistry laboratory teaching? using prompt engineering to enable ai in generating laboratory activities," *Journal of Chemical Education*, vol. 101, no. 5, pp. 1858–1864, 2024.

[34] M. Reid *et al.*, "Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context," *arXiv preprint arXiv:2403.05530*, 2024.

[35] Subhajournal, "Phishing Emails Dataset," https://www.kaggle.com/datasets/subhajournal/phishingemails, 2024, accessed on: 5-14-2024.

[36] Shashwatwork, "Web Page Phishing Detection Dataset," https://www.kaggle.com/datasets/shashwatwork/web-page-phishing-detection-dataset, 2024, accessed on: 5-14-2024.

[37] H. Sayadi *et al.*, "Intelligent malware detection based on hardware performance counters: A comprehensive survey," in *International Symposium on Quality Electronic Design (ISQED)*. IEEE, 2024, pp. 1–10.

[38] Z. He *et al.*, "Guarding against the unknown: Deep transfer learning for hardware image-based malware detection," *Journal of Hardware and Systems Security*, pp. 1–18, 2024.

[39] S. Ribes. *et al.*, "Machine learning-based classification of hardware trojans in fpgas implementing risc-v cores," in *Proceedings of the 10th International Conference on Information Systems Security and Privacy - ICISSP*, INSTICC. SciTePress, 2024, pp. 717–724.

[40] Z. He *et al.*, "When machine learning meets hardware cybersecurity: Delving into accurate zero-day malware detection," in *2021 22nd International Symposium on Quality Electronic Design (ISQED)*. IEEE, 2021, pp. 85–90.

[41] Z. He and et al., "Breakthrough to adaptive and cost-aware hardware-assisted zero-day malware detection: A reinforcement learning-based approach," in *IEEE 40th International Conference on Computer Design (ICCD)*. IEEE, 2024, pp. 1–10.

[42] M. Alawida *et al.*, "Unveiling the dark side of chatgpt: Exploring cyberattacks and enhancing user awareness," *Information*, vol. 15, no. 1, p. 27, 2024.

[43] AMi, "Obfuscated javascript dataset," 2021, accessed: 2024-08-08. [Online]. Available: https://www.kaggle.com/datasets/ami93/obfuscated-javascript-dataset

[44] S. Datta, "Deepobfuscode: Source code obfuscation through sequence-to-sequence networks," in *Intelligent Computing: Proceedings of the 2021 Computing Conference, Volume 2*. Springer, 2021, pp. 637–647.

[45] Y. Liu *et al.*, "Jailbreaking chatgpt via prompt engineering: An empirical study," *arXiv preprint arXiv:2305.13860*, 2023.

[46] T. Liu *et al.*, "Making them ask and answer: Jailbreaking large language models in few queries via disguise and reconstruction," *arXiv preprint arXiv:2402.18104*, 2024.