Not The End of Story: An Evaluation of ChatGPT-Driven Vulnerability Description Mappings

Xin Liu[1], **Yuan Tan**[1], **Zhenghang Xiao**[2], **Jianwei Zhuge**[3,4,*], **Rui Zhou**[1]
[1]Lanzhou University [2]Hunan University [3]Tsinghua University [4]Zhongguancun Laboratory
[1]{xliu2019,tany19,zr}@lzu.edu.cn, [2]kiprey@hnu.edu.cn, [3,4,*]zhugejw@tsinghua.edu.cn

Abstract

As the number of vulnerabilities increases day by day, security management requires more and more structured data. In addition to textual descriptions of vulnerabilities, security engineers must classify and assess vulnerabilities and clarify their associated techniques. Vulnerability Description Mapping (VDM) refers to mapping vulnerabilities to Common Weakness Enumeration (CWE), Common Attack Pattern Enumeration and Classification, ATT&CK Techniques, and other classifications. Accurate VDM is necessary to reduce the pressure of security management and improve the speed of security emergency response. ChatGPT is the latest state-of-the-art closed-source conversational large language model (LLM), which performs excellently on many tasks. This paper explores the application of closed-source LLMs to real-world security management scenarios by evaluating ChatGPT's performance on VDM tasks. The results show that although ChatGPT may be close to the level of human experts on some tasks, it still cannot replace the critical role of professional security engineers in vulnerability analysis. In a word, closed-source LLM is not the end of story.

1 Introduction

Constructing structured representations for vulnerabilities is an important part of the security management data infrastructures. Vulnerability description refers to the text used by vulnerability reporters to describe a vulnerability's cause, the scope of impact, and harm and is the foundation data for constructing vulnerabilities.

Vulnerability Description Mapping (VDM) refers to mapping vulnerabilities to Common Weakness Enumeration (CWE), Common Attack Pattern Enumeration and Classification, ATT&CK Techniques, and other classifications. Through VDM, people can more quickly understand the technical details of vulnerabilities and their associated exploitation and defense methods, which is important for security management and security research.

However, the cost of mapping through manual methods is unacceptable due to the growing size of vulnerability databases. Therefore, a series of research works have been carried out for automated VDM. With the development of natural language processing (NLP) technology, large models have come into view. ChatGPT (OpenAI, 2023) is a closed-source large language model (LLM), and it is generally believed to be the latest state-of-the-art NLP method. Existing data proves that ChatGPT performs no less than humans in text generation and knowledge Q&A, which lays the foundation for implementing VDM based on ChatGPT.

Common Vulnerabilities & Exposures (CVE) (MITRE, 2023) is the world's largest vulnerability database, containing hundreds of thousands of vulnerabilities in different products, with the research community's most complete and comprehensive vulnerability descriptions. During the public testing phase of ChatGPT, some security engineers have already used CVE to validate ChatGPT's ability on VDM tasks initially and marveled at its performance. However, there is still no large-scale, multi-dimensional evaluation of ChatGPT's VDM ability. In this paper, we designed an evaluation framework for ChatGPT and constructed multiple datasets based on CVE for two task types (Vulnerability-to-CWE and Vulnerability-to-ATT&CK) to evaluate ChatGPT's performance on VDM tasks.

2 Related Work

Vulnerability description mapping can help security researchers learn the structured knowledge of vulnerabilities, but it is not possible to map all vulnerabilities manually. These years security researchers have tried to solve this issue by applying the automated techniques, while there has been some prior work on automated mapping.

Findings of the Association for Computational Linguistics: ACL 2023, pages 3724–3731
July 9-14, 2023

Kenta et al. (Kanakogi et al., 2021, 2022) tried to use NLP-based approaches to determine the linkage of CAPEC-ID candidates and the given CVE-ID, based on the similarity between the CAPEC document and the CVE description. Hemberg et al. (Hemberg et al., 2022) also use the state-of-art pre-trained language model RoBERTa with a proposed fine-tuning and self-knowledge design to increase model performance in F1-score. Both of these works use language processing models for analysis and mapping, as most CVEs only contain pure text descriptions, and trying to extract useful information from CVEs for classification or mapping can only start with the text processing.

Different with NLP method, Yosra et al. (Lakhdhar and Rekhis, 2021) proposed a multi-label classification approach to automatically map vulnerabilities to attack techniques, and evaluated a set of machine learning algorithms to find out the best method. CVE2ATT&CK (Grigorescu et al., 2022) focused more on the processing of data sets. This work developed a data collection methodology to build a CVE dataset annotated with all corresponding ATT&CK pattern while addressing the problem of the severe imbalance of the data sets.

3 Evaluation Setup

3.1 Testing Framework

ChatGPT is a conversational model. Therefore, as shown in Figure 1, we interact with ChatGPT and obtain results by constructing questions based on vulnerability descriptions and directing the questions to ChatGPT. We first design a baseline question such as "Which CWE ID does this vulnerability description match?" (Mapping vulnerability description to CWE ID). Then, we attach a vulnerability description and send the baseline question with the vulnerability description to ChatGPT. When ChatGPT returns the answer, we parse the returned data with regular expressions and record the returned CWE IDs.

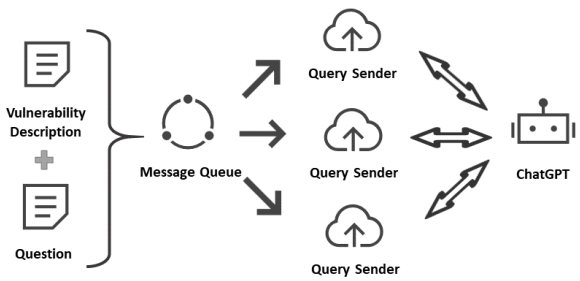

Figure 1: Architecture of Testing Framework

Strong prompts may enhance the precision of LLM's output. Therefore, we also use stronger prompts to evaluate ChatGPT's VDM performance. The ideal strong prompt would provide ChatGPT with all the classification criteria, enabling it to classify based on this comprehensive information. However, due to token limitations, this approach is not feasible. As an alternative strategy, we use a simple chain of thoughts: We first instructed ChatGPT to provide five possible categories (top 5) and their definition based on the vulnerability description. Then, we ask ChatGPT to find the most appropriate one (top 1) from them.

In this paper, we perform two types of evaluation: vulnerability description to CWE IDs, and vulnerability description to ATT&CK Technique IDs. Since CWE can be mapped to CAPEC according to fixed rules, we do not perform a vulnerability description to CAPEC mapping evaluation here.

3.2 Datasets

As mentioned earlier, CVE has the most complete and comprehensive vulnerability description data. Therefore, we construct the dataset for this paper based on CVE. In this paper, we constructed three different datasets based on CVE data, including one CVE-CWE dataset and two CVE-ATT&CK datasets.

- **Dataset I**: This dataset covers all 2021 CVE data (CVE-2021-*), including three fields: CVE ID, vulnerability description, and CWE ID. Please note that CVE did not provide CWE IDs for all vulnerabilities, and we excluded vulnerabilities for which CVE did not provide CWE IDs in the construction of this dataset. Finally, this dataset contains 13,513 vulnerabilities.

- **Dataset II**: This dataset is the CVE-ATT&CK Technique dataset with three fields: CVE ID, vulnerability description, and ATT&CK Technique ID. This dataset consists of 7,013 CVE vulnerabilities for the year 2021 (CVE-2021-*), for which the ATT&CK Technique ID is available through third-party vulnerability databases (e.g., VulDB).

- **Dataset III**: This dataset is a CVE-ATT&CK Technique Dataset built on BRON (Hemberg et al., 2021) and consists of three fields: CVE ID, vulnerability description, and a list of ATT&CK Technique IDs. Since BRON may

provide multiple ATT&CK Technique IDs for each CVE, a list of ATT&CK Technique IDs is used here instead of a unique ATT&CK Technique ID. (In the real world, a vulnerability may indeed correspond to multiple ATT&CK Technique IDs, and third-party vulnerability databases used by Dataset II usually only provide users with the most prominent ATT&CK Technique ID) This dataset contains a total of 25,439 CVE vulnerabilities. Besides, we ignored all sub-techniques in BRON.

We have shared all of these datasets to the research community via GitHub [1], including the raw results.

4 Results

4.1 Mapping Vulnerabilities to CWE IDs

This experiment was completed based on Dataset I and focused on verifying ChatGPT's ability to map descriptions of CVE vulnerabilities to CWE IDs.

Type	Prompts
Weak	Which CWE ID does this vulnerability description match? {vulnerability description}
Strong	CWE is a community-developed list of software and hardware weakness types. I give you the description of a vulnerability and you find the top five most possible CWE ID it may belong to. Answer in format of "CWE Number:Name". Then tell me the definition of theses 5 CWE-IDs. The description is {vulnerability description}
	According to the definition you have provided, please tell me the most appropriate CWE ID it may belong to. Remember to select the most suitable one.

Table 1: Prompts for CVE-CWE mappings

We use both weak and strong prompts shown in Table 3 to evaluate ChatGPT's performance of mapping vulnerabilities to CWE IDs. After we sent all the 13,513 CVE vulnerabilities in Dataset I to ChatGPT through the testing framework and responses were successfully obtained, we statistically analyzed the results from the vulnerability and CWE perspectives, respectively. We first analyze the results from the vulnerability perspective, which are shown in Table 2. From the table, we can see that more than half of the vulnerabilities' CWE IDs can be accurately determined by ChatGPT. If we mark a ChatGPT-outputted CWE ID as correctly-determined if it shares a common parent with the specific CWE ID recorded in the CVE database, then the majority of vulnerabilities' CWE IDs can be correctly output.

Prompt	Type	Correct	Ratio
Weak	Same	6,876	50.88%
	Parent	9,757	72.20%
	Grandparent	11,226	83.08%
Strong	Same	7,180	53.13%
	Parent	10,288	76.13%
	Grandparent	11,357	84.04%

Table 2: Vulnerability-perspective results of CVE-CWE mapping based on Dataset I (13,513 CVEs)

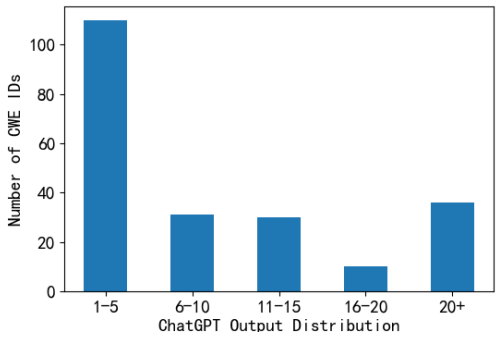

Figure 2: CWE-perspective results (weak prompt) of CVE-CWE mapping based on Dataset I. The X-axis is the distribution of the number of CWE IDs predicted by ChatGPT, and the Y-axis is the number of original CWE IDs. For example, vulnerabilities related to CWE-90 might be predicted as CWE-90/CWE-79/CWE-284 and so fall into the first category.

Next, we analyze the results of this experiment from a CWE perspective. The number of CWE IDs is too large to present the results using an obfuscation matrix. For a specific CWE ID, it may be predicted by ChatGPT to different CWE IDs. As shown in Figure 2, the results show a relative output concentration from ChatGPT.

[1] https://github.com/dstsmallbird/ChatGPT-VDMEval

4.2 Mapping Vulnerabilities to ATT&CK Technique IDs

This part evaluates ChatGPT's ability to map CVE vulnerability descriptions to ATT&CK Technique IDs. As shown in Table X, we also use both weak and strong prompts to evaluate ChatGPT's performance of mapping vulnerabilities to ATT&CK Technique IDs.

Type	Prompts
Weak	Please guess the ATT&CK techniques that belong to this vulnerability and list the IDs: {vulnerability description}
Strong	ATT&CK stands for Adversarial Tactics, Techniques, and Common Knowledge. I give you the description of a vulnerability and you find the top five most possible ATT&CK Technique ID it may belong to. Answer in format of "ATT&CK Technique ID:Name" The description is {vulnerability description}
	According to the definition you have provided, please tell me the most appropriate ATT&CK Technique ID it may belong to. Remember to select the most suitable one.

Table 3: Prompts for CVE-ATT&CK mappings

ATT&CK Technique IDs are chosen as the targets mainly because there is currently a lack of publicly available datasets for CVE-ATT&CK. Thus ChatGPT can hardly direct access this knowledge from existing datasets, allowing for a better representation of ChatGPT's ability to handle VDM tasks in the absence of high-quality training data - which is more in line with real-world requirements.

We first complete this evaluation based on Dataset II, which was collected exclusively from third-party databases that provide only one dominant ATT&CK Technique ID, while ChatGPT may provide multiple results: if the unique ATT&CK Technique ID given by ChatGPT is the same as the one in Dataset II, we mark it as "If the ATT&CK Technique ID in Dataset II is part of the result given by ChatGPT, we mark it as "intersected". In both cases, we consider this to be correct. The results of this evaluation are shown in Table 4. We can see from the results that ChatGPT's performance in this evaluation is not satisfactory.

Prompt	Type	Count	Ratio
Weak	Strictly-Equal	272	3.88%
	Intersected	570	8.13%
	Correct Output	842	12.01%
Strong	Strictly-Equal	796	11.35%
	Intersected	1,435	20.46%
	Correct Output	2,231	31.81%

Table 4: Results of CVE-ATT&CK mapping based on Dataset II (7,013 CVEs).

Prompt	Type	Count	Ratio
Weak	Strictly-Equal	109	0.43%
	Intersected	2,720	10.69%
	Correct Output	2,829	11.12%
Strong	Strictly-Equal	2,980	11.71%
	Intersected	5,356	21.05%
	Correct Output	8,336	32.76%

Table 5: Results of CVE-ATT&CK mapping based on Dataset III (25,439 CVEs).

To avoid errors in the third-party databases overly influencing the assessment results, we therefore next completed the assessment using Dataset III, the results of which are shown in Table 5. From the tables, we can see that the difference is tiny.

In summary, ChatGPT's performance on the CVE-ATT&CK task is unsatisfactory and barely meets real-world requirements. The results are likely due to the lack of publicly available datasets for this task and the fact that ATT&CK Techniques are more variable than CWE. This evaluation suggests that ChatGPT is not the key to VDM and cannot replace security personnel in real-world VDM tasks.

4.2.1 Comparing with Existing Approaches

Since there are very few papers working on CVE-CWE mappings, we only use CVE-ATT&CK task and Dataset III for comparison. The state-of-the-art approach for CVE-ATT&CK mapping is CVET (Ampel et al., 2021) and we use the results provided by its literature to build Table 6.

We can see the performance of existing approaches significantly surpasses that of ChatGPT, which indicates that even with the help of strong prompts, closed-source LLMs represented by ChatGPT can hardly catch up with the performance of

Type	Method	Ratio
Classical Machine Learning	Random Forest	37.67%
	SVM	46.34%
Deep Learning	LSTM	71.89%
	Transformer	70.82%
Fine-Tuned Models	GPT-2	64.56%
	BERT	69.41%
Self-Distillation	CVET	71.49%
Closed-Source LLM (ChatGPT)	Weak Prompt	11.12%
	Strong Prompt	32.76%

Table 6: Comparisons between existing approaches on CVE-ATT&CK mapping with Dataset III

existing state-of-the-art approaches in vulnerability description mappings. Since GPT-2 performs well, we believe the main reason for ChatGPT's poor performance is the lack of task-oriented fine-tuning. It seems that the real future is the open-source task-oriented fine-tuned LLMs rather than the closed-source ones.

4.3 Discussions

4.3.1 Interesting Findings

In this paper, we can see that ChatGPT's performance on the Vulnerability-to-CWE task is pretty promising, but its performance on the Vulnerability-to-ATT&CK task is unsatisfactory. We analyzed the definition of ATT&CK Techniques understood by ChatGPT and found that ChatGPT had many misinterpretations of many ATT&CK Techniques. We suspect this may be due to the high diversity of ATT&CK Techniques and the fact that the Vulnerability-to-ATT&CK datasets are of poor quality (compared to the Vulnerability-to-CWE data provided by CVE).

Due to the fixed correspondence between CAPEC and CWE, we did not initially use the Vulnerability-to-CAPEC task to evaluate ChatGPT. However, like ATT&CK Techniques, CAPEC IDs are more diverse than CWE IDs, and the available Vulnerability-to-CAPEC datasets are lower quality. Therefore, we designed several Vulnerability-to-CAPEC queries to verify whether such tasks suffer from the same problems faced by Vulnerability-to-ATT&CK tasks. The results showed that ChatGPT incorrectly interpreted the CAPEC IDs and output the wrong answers. For example, ChatGPT considers CAPEC-60 the "Hard-coded Cryptographic Key", but it should be "Reusing Session IDs".

These findings hint at improving ChatGPT's performance on complex VDM tasks such as Vulnerability-to-ATT&CK by providing a priori knowledge or predefinition.

4.3.2 Limitations

We completed this evaluation based on the Beta version of ChatGPT, and the relevant results may change as OpenAI continues to improve ChatGPT. In addition, we use the ATT&CK Technique datasets collected from third-party institutions and publicly available data on the Internet. The large size of these datasets makes it difficult for us to verify their accuracy manually. Therefore, errors in these datasets may affect the conclusion of this paper.

5 Conclusion

In this paper, we selected two classic tasks and constructed three datasets to pioneer a large-scale, multi-dimensional evaluation of ChatGPT's ability to handle VDM tasks. The results show that ChatGPT performs well on the Vulnerability-to-CWE task and has met or exceeded the level of human experts. We believe that this is because the public Vulnerability-to-CWE data is relatively high-quality.

However, the performance on the Vulnerability-to-ATT&CK task, which has poor public data quality, is unsatisfactory. In addition, ChatGPT appears to have serious problems with the conceptual understanding of CAPEC and ATT&CK Techniques, which may be the main reason for its poor performance on these tasks. In summary, this paper demonstrates that ChatGPT is still not directly usable for VDM tasks, and ChatGPT is not the end of story.

Acknowledgements

This work was supported in part by National Natural Science Foundation of China under Grant U1936121.

References

Benjamin Ampel, Sagar Samtani, Steven Ullman, and Hsinchun Chen. 2021. Linking common vulnerabilities and exposures to the mitre att&ck framework: A self-distillation approach. *arXiv preprint arXiv:2108.01696*.

Octavian Grigorescu, Andreea Nica, Mihai Dascalu, and Razvan Rughinis. 2022. Cve2att&ck: Bert-based

mapping of cves to mitre att&ck techniques. *Algorithms*, 15(9):314.

Erik Hemberg, Jonathan Kelly, Michal Shlapentokh-Rothman, Bryn Reinstadler, Katherine Xu, Nick Rutar, and Una-May O'Reilly. 2021. Linking threat tactics, techniques, and patterns with defensive weaknesses, vulnerabilities and affected platform configurations for cyber hunting.

Erik Hemberg, Ashwin Srinivasan, Nick Rutar, and Una-May O'Reilly. 2022. Sourcing language models and text information for inferring cyber threat, vulnerability and mitigation relationships.

Kenta Kanakogi, Hironori Washizaki, Yoshiaki Fukazawa, Shinpei Ogata, Takao Okubo, Takehisa Kato, Hideyuki Kanuka, Atsuo Hazeyama, and Nobukazu Yoshioka. 2021. Tracing cve vulnerability information to capec attack patterns using natural language processing techniques. *Information*, 12(8):298.

Kenta Kanakogi, Hironori Washizaki, Yoshiaki Fukazawa, Shinpei Ogata, Takao Okubo, Takehisa Kato, Hideyuki Kanuka, Atsuo Hazeyama, and Nobukazu Yoshioka. 2022. Comparative evaluation of nlp-based approaches for linking capec attack patterns from cve vulnerability information. *Applied Sciences*, 12(7):3400.

Yosra Lakhdhar and Slim Rekhis. 2021. Machine learning based approach for the automated mapping of discovered vulnerabilities to adversial tactics. In *2021 IEEE Security and Privacy Workshops (SPW)*, pages 309–317. IEEE.

MITRE. 2023. Cve program.

OpenAI. 2023. Chatgpt: Optimizing language models for dialogue.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
4.3.2

☐ A2. Did you discuss any potential risks of your work?
Not applicable. This submission is an evaluation without any new approach proposed.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Left blank.

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 3.2

☑ B1. Did you cite the creators of artifacts you used?
Section 3.2

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Section 3.2

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Section 3.2

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

☐ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Not applicable. Left blank.

C ☒ Did you run computational experiments?

Left blank.

☐ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
No response.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☐ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
No response.

☐ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
No response.

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
No response.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.