CyberLLM: Enable Mapping CVE to Tactics and Techniques of Cyber Threats via LLM

Ziming Zhao[1], Zhaoxuan Li[2], Tingting Li[1], and Fan Zhang[1(✉)]

[1] Zhejiang University, Hangzhou, China
{zhaoziming, litt2020, fanzhang}@zju.edu.cn
[2] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
lizhaoxuan@iie.ac.cn

Abstract. Understanding cyber threats is crucial for effective defense in the field of cybersecurity. If we can automatically map Common Vulnerabilities and Exposures (CVEs) to attack tactics and techniques, it will help practitioners quickly analyze reports and take responsive actions. In this paper, we introduce CyberLLM, leveraging the tailor-made large language model for mapping CVEs to cyber threat tactics and techniques. Specifically, we model the mapping of CVE to tactics and techniques as a multi-label classification problem, given that many CVEs correspond to multiple techniques of ATT&CK. Then, the text description is vectorized through the tokenization process, and we deploy a series of data augmentation techniques to enrich the semantic information. Considering that external knowledge bases are helpful to enhance the contextual information of the queried CVE, CyberLLM designs a retrieval strategy based on the Jaccard distance calculation. Finally, we support flexible model fine-tuning to adapt to the needs. Through extensive experiments, we demonstrate the superiority of CyberLLM compared with 7 representative state-of-the-art methods. We also perform ablation experiments on data augmentation and evaluate the effectiveness of using retrieval information. Furthermore, we provide a series of deep insights in terms of feature attribution and attention weight visualization.

Keywords: Cyber Threats · Techniques and Tactics · Network Attacks · Large Language Model.

1 Introduction

With the development of the Internet, cyber attacks are emerging in an endless stream [3,4,9]. To achieve specific attack goals, adversaries often deploy a series of tactics and techniques to launch intrusions [24,35,37]. Understanding the tactics and techniques used by adversaries is crucial for effective defense [13,30,36]. As a representative framework, MITRE ATT&CK [29] provides a globally-accessible knowledge base for the development of specific and comprehensive threat models and methodologies [34]. The core components of the ATT&CK framework [29] include: (i) *Tactics*: High-level categories of objectives an adversary may try to achieve. (ii) *Techniques*: Specific methods used by adversaries to achieve their

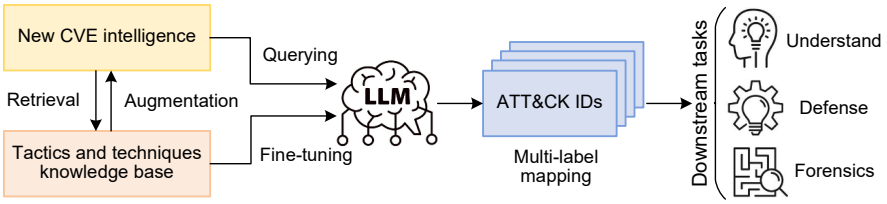

Fig. 1. The high-level illustration of CyberLLM.

tactical goals. (iii) *Procedures*: Detailed descriptions of how techniques are executed in specific instances.

Common Vulnerabilities and Exposures (CVEs) serve as a critical reference for identifying and addressing these threats [19,22]. However, mapping CVEs to specific tactics and techniques is a challenging task due to the vast and dynamic nature of cyber threat intelligence [1]. The complexity arises from the need to process and correlate large volumes of heterogeneous data [21], including textual descriptions, technical reports, and real-time threat feeds. On the one hand, practitioners need to search internal knowledge bases and collect external intelligence information sources to associate CVE information to be analyzed [14]. On the other hand, expert knowledge is required to understand threat intelligence and attack principles [36]. These tasks are time-consuming and laborious.

To address these issues, we intend to drive automated mapping solutions for CVEs to tactics and techniques by embracing advances in the large language model (LLM). In this paper, we introduce CyberLLM, leveraging the tailor-made large language model for mapping CVEs to cyber threat tactics and techniques. Figure 1 provides a high-level illustration of CyberLLM. Specifically, we model the mapping of CVE to tactics and techniques as a multi-label classification problem, given that many CVEs correspond to multiple techniques of ATT&CK. Then, the text description is vectorized through the tokenization process. Particularly, we deploy a series of data augmentation techniques to enrich the semantic information of the attack. Considering that external knowledge bases are helpful to enhance the contextual information of the queried CVE, CyberLLM designs a retrieval strategy based on the Jaccard distance calculation. Finally, we support flexible model fine-tuning to adapt to the needs.

Through extensive experiments, we demonstrate the superiority of CyberLLM compared with 7 representative state-of-the-art (SOTA) schemes. We also perform ablation experiments on data augmentation and evaluate the effectiveness of using retrieval information. Furthermore, we provide a series of deep insights in terms of feature attribution and attention weight visualization.

Contributions. Our contributions are summarized as follows:

- We formalize the mapping of CVEs to tactical and technical questions as a multi-label classification problem and propose to embrace the advancement of large language models to automate the above task.

- We present CyberLLM, leveraging the tailor-made large language model for mapping CVEs to cyber threat tactics and techniques. CyberLLM is designed with a series of modules, including tokenization, data augmentation, retrieval strategy, and model fine-tuning.
- We implement a prototype of CyberLLM and conduct extensive evaluations to demonstrate the advantages of CyberLLM compared with 7 representative schemes based on the public dataset. We also conduct ablation experiments and model attribution to provide deep insights.

2 Problem Space

In this section, we formalize the problems of the CVE techniques/tactics mapping process, and introduce the adversary model and assumptions.

2.1 Problem Formulation

This task involves processing textual reports of CVEs to predict their corresponding tactics and techniques as defined by the MITRE ATT&CK framework [29]. This requires a deep understanding of the semantic content within these reports and the ability to correlate this content with known threat patterns. The CVE knowledge base provides a standardized method for identifying vulnerabilities. CVE descriptions contain rich semantic information regarding vulnerabilities, including the nature of the vulnerability, the affected software, and potential impacts. Let a CVE description be represented as a set of textual features:

$$D_{CVE} = \{d_1, d_2, ..., d_n\} \tag{1}$$

where d_i represents the i-th textual feature or term.

For the ATT&CK framework, developed by MITRE, classifies cyber attack techniques into a comprehensive matrix. The ATT&CK framework categorizes vulnerabilities into various tactics and techniques. Specifically, the ATT&CK framework can be represented as a set of tactics Tac:

$$Tac = \{t_1, t_2, ..., t_m\} \tag{2}$$

where each tactic $t \in Tac$ contains a set of techniques. That is,

$$t_j = \{Tec_1, Tec_2, ..., Tec_k\} \tag{3}$$

and Tec denotes the unique identifier of each technique.

The mapping process aims to associate each CVE description with one or more techniques in the ATT&CK framework. This is achieved through a mapping function M, which takes a CVE description and returns a set of corresponding ATT&CK techniques:

$$M : D_{CVE} \rightarrow P(Tec) \tag{4}$$

where $P(Tec)$ represents the power set of ATT&CK techniques. In other words, the above process can be modeled as a multi-label classification problem.

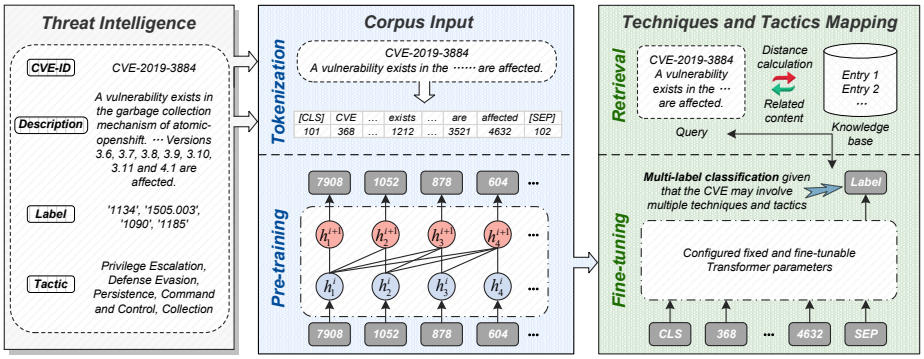

Fig. 2. The overview design of CyberLLM, including threat intelligence extraction, tokenization, retrieval, and model fine-tuning.

2.2 Adversary Model and Assumptions

For adversary model and assumptions, we could access CVE threat intelligence that provides insights into vulnerability exploitation or the underlying attack principles. The task mainly focuses on processing textual reports, and we assume that the CVE reports are written in natural language. The specific tactics, techniques, and procedures (TTPs) used to exploit these vulnerabilities are not fully known or may vary across different attack processes. Due to the CVE is naturally often associated with a set of TTPs, reflecting the diverse attack strategies in which they can be exploited in the wild. We consider the multi-label mapping process from CVE to ATT&CK. And we assume that the MITRE ATT&CK framework offers useful taxonomy information on cyber threat behaviors. In addition, we also consider the existence of external knowledge bases for retrieving information related to the current query to supplement the contextual content.

3 Design Details of CyberLLM

In this section, we elaborate on the design details of CyberLLM. The overall workflow is shown in Figure 2. First, threat intelligence includes CVE-ID, corresponding descriptions about the attack or intrusion process, the tactics and labels. Then, the CVE-ID and description are jointly used as the input corpus, and tokenization is performed to obtain the vectorized representation. Meanwhile, pre-training LLMs focus on self-supervised learning through the reconstruction of input data without any labels in this process. Finally, retrieval and fine-tuning are employed to achieve technical and tactical mapping. Among them, the former is responsible for retrieving relevant content from the knowledge base to enhance the query itself, while the latter adapts to a given labeled dataset with configuring fine-tunable parameters. Particularly, the final output of the LLM model is tailored to be the multi-label classification result, given that CVE usually involves multiple techniques and tactics.

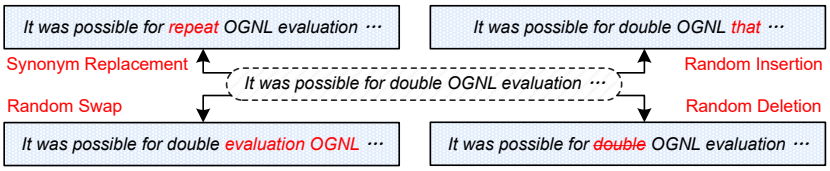

Fig. 3. Illustration of data augmentation operations on CVE-2017-16861 description.

3.1 Tokenization and Pre-Training

We introduce the tokenization which is used for constructing the vectorized representation of threat intelligence corpus, and the pre-training processes.

Tokenization. Tokenizers are fundamental components in the architecture of large language models (LLMs), enabling the translation of raw textual data into a structured format that can be ingested and processed by these models. The tokenizer starts by encoding the input text into a sequence of tokens, where each token represents a word or a piece of a word. It operates by iteratively merging the most frequent pairs of bytes (or characters in the context of NLP) into larger units until a desired vocabulary size is reached. This process allows the tokenizer to handle out-of-vocabulary words by breaking them down into their constituent subword units.

Pre-Training. Pre-training process mainly involves self-supervised learning that leverages the inherent structure of the data itself to guide the learning process. The core idea is to treat the task of data reconstruction as a proxy for supervision. By learning to reconstruct the input data, models are forced to develop a deep understanding of the underlying patterns and structures in the language. This can be achieved through various techniques. (i) Masked Language Modeling (MLM), where the model predicts randomly masked tokens in a sentence. (ii) Next Sentence Prediction (NSP), where the model predicts whether a given sentence follows another in a text. (iii) Denoising Autoencoders, where the model learns to reconstruct clean sentences from corrupted versions.

3.2 Data Augmentation

Threat intelligence is a critical component of cybersecurity, providing essential information about potential adversaries, their capabilities, and the tactics, techniques, and procedures they may employ. The diversity and variability of these descriptions motivate us to deploy data augmentation techniques for expanding the information semantics. As the typical scheme, Easy Data Augmentation (EDA) [33] is a set of techniques designed to enhance the performance of text classification models, particularly when training with smaller datasets. The following four core operations are involved in EDA, as shown in Figure 3.

(i) **Synonym Replacement (SR).** Synonym replacement involves selecting non-stop words from a sentence and replacing them with their synonyms. This process enriches the dataset with lexical variations while preserving the original

meaning. Given the number of words n to be replaced, which is determined by the sentence length l, and a hyperparameter α, such that $n = \alpha \times l$. This operation can be represented as: for a sentence $S = \{w_1, w_2, ..., w_{|S|}\}$, let $SR(S) = S'$, where $S' = \{w'_1, w'_2, ..., w'_{|S|}\}$ and for each i, w'_i is a synonym of w_i.

(ii) **Random Insertion (RI).** Random insertion introduces new words into the sentence by selecting synonyms of existing words and placing them at random positions. This operation is performed n times, where n is defined by the same hyperparameter α and sentence length l. The resulting sentence S' includes original words with synonyms inserted at various points, let $RI(S) = S'$, where S' is formed by inserting n synonyms of words from S at random positions.

(iii) **Random Swap (RS).** Random swap shuffles the order of words in a sentence by randomly selecting pairs of words and swapping them. This operation is also repeated n times, adding diversity to the sentence structure. The outcome, S', maintains the original words but in a new arrangement. Let $RS(S) = S'$, where S' is formed by swapping n pairs of words in S at random positions.

(iv) **Random Deletion (RD).** Random deletion reduces the sentence length by randomly removing words with a probability p. The expected number of deletions is determined by the sentence length l and the hyperparameter α, where $p = \frac{\alpha}{l}$. This results in a shorter sentence S' with some words removed, let $RD(S) = S'$, where each word in S has a probability p of being omitted in S'.

The above operations provide a concise yet effective method for augmenting text data, which is valuable for improving model performance when working with limited training data. Such a process introduces variability into the dataset, which is beneficial to prevent overfitting and enhance the model generalizability.

3.3 Retrieval Strategy

In the dynamic field of cybersecurity, the ability to understand and address threats is paramount. Meanwhile, external knowledge bases are treasure troves of information that can significantly augment the understanding of specific attacks or vulnerabilities. They may contain a wealth of data that is not immediately available in an organization's internal threat intelligence repository. Also, they could include historical incidents and patterns that can help identify technical information on known vulnerabilities, exploit trends, and mitigation strategies.

To this end, we design a tailor-made retrieval pipeline to mine relevant content for a query. As Algorithm 1 shown, the algorithm begins by tokenizing the input query using the provided tokenizer $\mathcal{T}$, resulting in a tokenization vector V_q. It calculates the frequency of each token in V_q to create a counter C_q. We leverage the Jaccard distance which is a measure of the dissimilarity between two sets. Specifically, the Jaccard distance is defined as the complement of the Jaccard index, which is the size of the intersection divided by the size of the union of the sample sets. Given two sets A and B, the Jaccard distance D_J is calculated using the following formula:

$$D_J(A, B) = 1 - \frac{|A \cap B|}{|A \cup B|} \tag{5}$$

Algorithm 1 The retrieval strategy in CyberLLM

Input: The retrieval knowledge base $\mathcal{K}$, the input query $Query$, and the tokenizer $\mathcal{T}$
Output: The retrieved information $Infor$

```
# Perform tokenization
Obtain the tokenization vector V_q = T(Query)
# Calculate the frequency of each element
Calculate C_q ← Counter(V_q)
Initialize the minimum Jaccard distance D_min = ∞
Initialize the entry index of minimum Jaccard distance ind_min
# Traverse the knowledge base
for all knowledge k_i in K do
    Obtain the tokenization vector V_i = T(k_i)
    Calculate C_i ← Counter(V_i)
    # Calculate Jaccard distance
    Calculate the intersection size
        intersection ← Σ_{item∈C_q} min(C_q[item], C_i.get(item))
    Calculate the union size
        union ← Σ_{item∈C_q} C_q[item] + Σ_{item∈C_i} C_i[item] − intersection
    Compute jaccardDist ← (1 − intersection/union)
    # Update the minimum distance
    if (jaccardDist < D_min) then
        Update D_min ← jaccardDist
        Update ind_min ← i
    end if
end for
return The retrieved information Infor ← K[ind_min]
```

where $|A \cap B|$ is the number of elements in the intersection of sets A and B, and $|A \cup B|$ is the number of elements in the union of sets A and B.

The process of retrieving information from external knowledge bases is aimed at finding relevant content that can enhance the semantic and contextual information of a given query. On the one hand, aligning the query with related concepts and entities from the knowledge base could increase the semantic richness of the query. On the other hand, the retrieved additional labeled content can be expanded to facilitate model fine-tuning.

3.4 LLM Fine-Tuning

For model fine-tuning, we need to gather a corresponding dataset of CVEs along with their corresponding ATT&CK techniques. Then, we could preprocess the CVE descriptions by normalizing text and tokenizing, and convert the ATT&CK techniques into a suitable format for multi-label classification. Meanwhile, data augmentation (§ 3.2) could be employed to enrich intelligence semantics.

Given a pre-trained language model (*e.g.*, BERT [7], RoBERTa [20], or GPT-2 [12]) as the base model, we modify the output layer for outcoming multiple labels. Use a suitable loss function for multi-label classification, such as binary

cross-entropy loss. Feed the preprocessed CVE descriptions through the model to obtain predictions. Compute the loss between the predicted labels and the ground truth labels. Perform backpropagation to update the model's weights. Note that we can fix certain parameters to reduce GPU usage according to requirements. Implement early stopping based on the performance change to avoid unnecessary training. This fine-tuning process can occur when coping with the initial dataset, or it can be used to adapt to the retrieved external information.

4 Evaluation

In this section, we extensively evaluate CyberLLM, with code available online[3]. Specifically, the experiments are designed to answer the following questions.

RQ1. How CyberLLM's performance compared with SOTA?
RQ2. How effective is the data augmentation?
RQ3. How effective is the retrieval enhancement?
RQ4. Could we provide some deep insights into CyberLLM?

4.1 Experimental Setup

Datasets. For a fair comparison, we use the European Union Agency for Cybersecurity (ENISA) report dataset for evaluation. The ENISA dataset collects a total of 27,471 vulnerability information published during 01/01/2018 to 30/09/2019 from various data sources. Based on the Common Attack Pattern Enumeration and Classification (CAPEC) information found in both National Vulnerability Database (NVD) and ATT&CK, this dataset contains 8,077 CVES that are mapped to 50 MITRE ATT&CK techniques. Specifically, subfigure (a) displays the frequency of each type of technical and tactical label. For example, the most frequently occurring label is '1562.003', which corresponds to 'Impair Command History Logging'. Subfigure (b) presents the frequency for the number of labels, it is clear that most threat intelligence corresponds to multiple labels.

Baselines. We consider a series of representative baselines.

- **CVET** [2] employed a self-distillation approach, leveraging a BERT-based model, specifically RoBERTa, to establish automated links between CVE and MITRE Tactics.
- **SciBERT** [11] utilized BERT-based language models with data augmentation to establish connections between CVEs and MITRE ATT&CK.
- **JEM** [15] mapped CVE's to ATT&CK by using an Multi-Head Joint Embedding Neural Network model.
- **LabelPowerset** [23] has the Multilayer Perceptron as the base classifier and 2 hidden layers. Moreover, the softmax function was used for activation.
- **Basic Models** includes three Machine Learning (ML) methods. CatBoost [25] and XGBoost [6] are based on (Extreme) Gradient Boosting algorithms. ClassifierChain [26] implements multi-label classification by performing a series of binary classifications.

[3] See anonymous repository https://github.com/Secbrain/CyberLLM

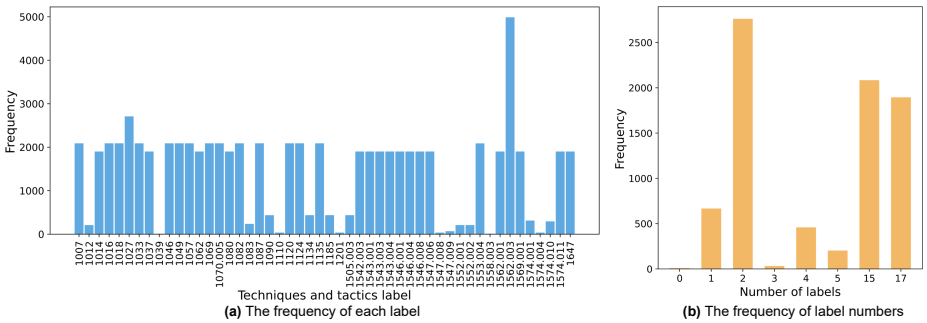

Fig. 4. The details of the dataset.

Table 1. Comparison results of baseline models.

Model	Accuracy	Precision	Recall	F1 score
CVET [2]	0.7613	0.7717	0.7821	0.7769
ClassifierChain [26]	0.7222	0.7813	0.7220	0.7505
XGBoost [6]	0.6518	0.7623	0.6805	0.7191
CatBoost [25]	0.6291	0.7354	0.6377	0.6831
SciBERT [11]	0.6382	0.6195	0.6068	0.6131
LabelPowerset [23]	0.7425	0.7791	0.7063	0.7409
JEM [15]	0.7725	0.8013	0.7582	0.7792
CyberLLM (Ours)	**0.8193**	**0.8560**	**0.8228**	**0.8391**

Settings. We use the default configuration of *GPT-2* model in the *transformers* library. For the model, the number of layers is 12, the hidden size is 768, the number of attention heads is 12, and the learning rate is 1×10^{-5}. For the tokenization parameters, the max sequence length is 1024 tokens, the padding token is '[PAD]', and the special tokens involve '[CLS]', '[SEP]', and '[UNK]'.

4.2 Comparison with SOTA (RQ1)

We first compare CyberLLM with the state-of-the-art (SOTA) methods, the results are summarized in Table 1. In general, solutions based on large language models (*e.g.,* CVET and JEM) have more advantages than traditional models (*e.g.,* CatBoost, XGBoost, and ClassifierChain). Our design also follows this trend. It is clear that CyberLLM significantly outperforms 7 baseline models with aspect to four performance metrics. The most competitive baseline is JEM, which achieves an F1 score of 77.92%. Nonetheless, CyberLLM achieves an F1 score of 83.91%, which is ∼6% higher than JEM. This demonstrates the effectiveness of CyberLLM in mapping CVE to ATT&CK tactics and techniques.

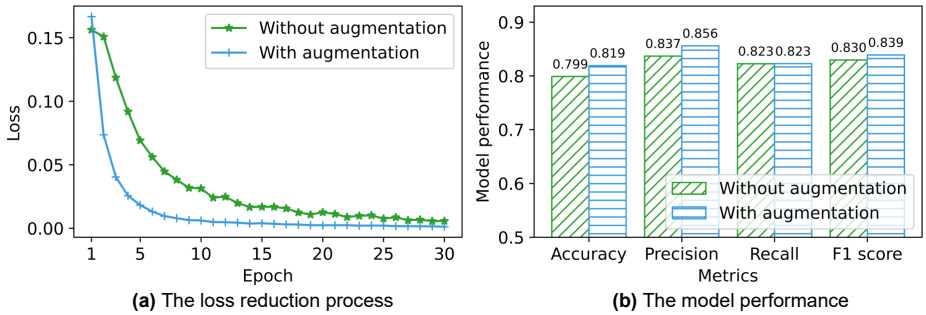

Fig. 5. The ablation experiment for data augmentation.

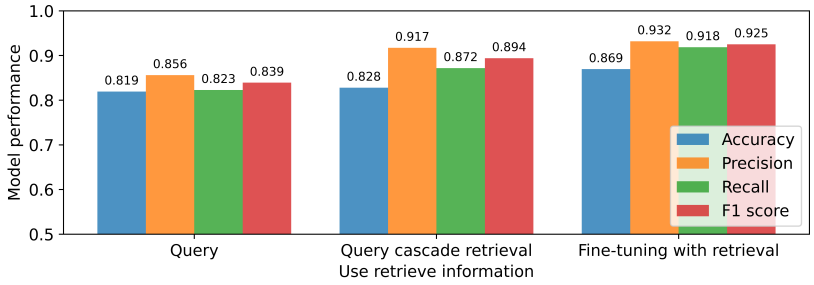

Fig. 6. Using the retrieval information.

4.3 Ablation Experiment (RQ2)

Then, we conduct ablation experiments for the data augmentation module. As shown in Figure 5 (a), we observe that as the number of epochs increases, the model fine-tuning loss with data augmentation decreases faster, and the convergence value is lower than that without data augmentation. Therefore, with data augmentation, the model can achieve better performance, as shown in Figure 5 (b). Particularly, even without data augmentation, CyberLLM still outperforms the existing baseline, *e.g.,* 79.9% accuracy in CyberLLM (without data augmentation), which is higher than 77.25% in JEM. Note that data augmentation has a significant effect on precision, but a weak effect on recall, which indicates that the primary role of data augmentation may be to reduce false positives.

4.4 Retrieval Enhancement Experiment (RQ3)

In § 3.3, we present the retrieval strategy related to the query. For one thing, the retrieved information can be used to enhance the query, thus enriching the contextual semantics. For another, the retrieved information with labels, can be further used to fine-tune the model for adaptation. We plot the model performance in Figure 6, we find that the use of retrieval information can indeed

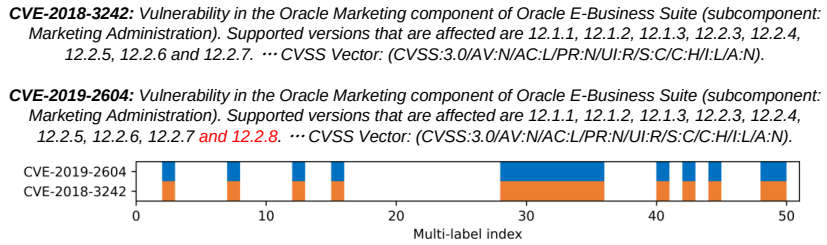

Fig. 7. The retrieval entry case.

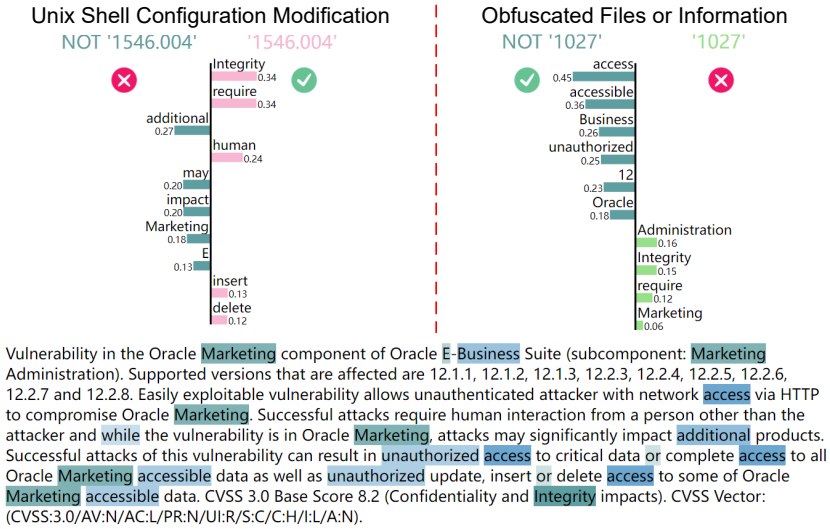

Fig. 8. The feature attribution results of CyberLLM.

improve model performance. Among them, the effect of fine-tuning adaptation is more effective than query enhancement because it can directly enrich the knowledge base of the model. In addition, we provide a retrieval case in Figure 7. The query is CVE-2019-2604, after the Jaccard distance calculation, the CVE-2018-3242 entry is retrieved. We find that the only difference between them is the supported version list. The former additionally supports version 12.2.8. However, the multi-label indexes corresponding to the two CVEs are exactly the same, so the retrieved information can effectively supplement the query and guide the correct classification results.

4.5 Deep Insights (RQ4)

To provide deep insights for CyberLLM, we develop the model explanations in this section. For the input text, we use the Lime [27] library to perform feature attribution to analyze the importance of words for label identification. As

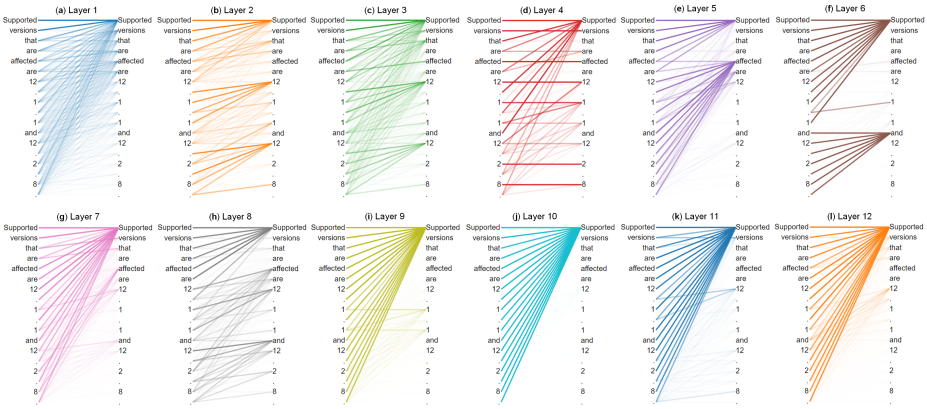

Fig. 9. The attention weight matrix visualization in CyberLLM.

shown in Figure 8, we observe some words contribute more to the model prediction results, such as 'Marketing', 'access', 'accessible', 'unauthorized', 'Integrity', 'delete', and 'insert'.

For example, for the identification {'1546.004': 'Unix Shell Configuration Modification'}, some important words involve 'Integrity', 'require', 'human', 'delete', and 'insert'. This is related to the technique description of '1546.004', which contains '*user opens a command-line interface or remotely logs in a login shell is initiate*' and '*These configuration scripts run at the permission level of their directory and are often used to set environment variables, create aliases, and customize the user's environment. When the shell exits or terminates, additional shell scripts are executed to ensure the shell exits appropriately*" in its description. And for the identification {'1027': 'Obfuscated Files or Information'}, some important words involve 'access', 'accessible', 'Business', 'unauthorized', and 'Oracle'. This is related to the technique description of '1027', which contains '*Adversaries may attempt to make an executable or file difficult to discover or analyze by encrypting, encoding, or otherwise obfuscating its contents on the system or in transit*' in its description.

In addition, we use the BertViz [32] library to visualize the weight parameters of the multi-head attention matrix in the model. According to the case of § 4.4, we feed the sentence 'Supported versions that are affected are 12.1.1 and 12.2.8.' and observe the attention parameters in CyberLLM, aims to explore whether supporting multiple versions of '12.2.8' will seriously affect model inference. In Figure 9, we display the attention weights of the first head in the 12 layers. Overall, among the attention weights, the word 'Supported' receives the most attention, followed by 'versions', 'affected', *etc.* The attention of '8' (from '12.2.8') is not high. This means that although the '12.2.8' version number is additionally supported, it does not affect the mapping results for the ATT&CK tactics and techniques in CyberLLM.

5 Related Work

In this section, we outline the related work in terms of vulnerability modeling, multi-label classification, and automated analysis technology.

Vulnerability Modeling. In the field of cybersecurity [18, 31], vulnerability modeling is fundamental for understanding and classifying security vulnerabilities [28]. As one of the most widely used vulnerability databases, CVE only offers basic descriptions of vulnerabilities and does not directly link them to specific attack methods or tactics. To bridge these gaps, the MITRE ATT&CK framework [29] was introduced, providing a systematic classification and documentation of adversary tactics, techniques, and procedures (TTPs), thus offering a more detailed and targeted reference for incident response and threat intelligence [5]. Despite the valuable resources these systems provide for vulnerability management and threat modeling, the challenge remains in effectively mapping CVE information to specific MITRE ATT&CK techniques [17].

Multi-Label Classification. Mapping CVEs to MITRE ATT&CK techniques often involves a multi-label classification problem, as a single vulnerability can be associated with multiple attack techniques. Traditional single-label classification methods are insufficient for handling this complexity, which has made multi-label classification a focal point in recent research. Existing approaches include problem transformation methods [15], which convert the multi-label problem into multiple single-label problems, and algorithm adaptation methods [23,39], which modify existing classification algorithms to handle multi-label outputs. However, the primary challenges in multi-label classification lie in the interdependencies between labels and the high-dimensional nature of the data, which increase the complexity of the classifiers and the difficulty of training them effectively [11].

Automated Analysis Technology. Recent advancements in large language models have brought new opportunities for automatic intelligence analysis [24], and pre-trained models have also been introduced into the landscape of cyber threat intelligence analysis [11]. Some arts [30] have explored multi-source information fusion and the learning of Advanced Persistent Threat (APT) chains, highlighting the challenges and opportunities in extracting actionable intelligence. However, these methods are constrained by their reliance on traditional NLP techniques or the architecture of pre-trained models, which may not fully capture the complex relationships and nuances in the data [13].

In contrast, our CyberLLM leverages advanced natural language processing within a large language model framework, providing a more comprehensive and accurate mapping of CVEs to cyber threat tactics and techniques. Our model's architecture and training methodology are tailored to address the specific challenges of the domain, offering superior performance and deeper insights.

6 Discussion, Limitations, and Future Work

Effectiveness and Practicality. In the real world, cybersecurity professionals can utilize CyberLLM to enhance their vulnerability management processes.

By inputting CVE descriptions into the model, practitioners can quickly obtain mappings to relevant cyber threat tactics and techniques. Threat intelligence analysts can leverage CyberLLM to deepen their understanding of emerging threats. This capability is particularly valuable for proactive threat hunting and developing mitigation strategies. Meanwhile, CyberLLM can aid incident responders by quickly analyzing incident reports and correlating them with known CVEs and threat tactics. Security Operations Center (SOC) teams [16] can benefit from CyberLLM by integrating the model into their monitoring and detection tools. The model can assist in identifying anomalous behaviors and correlating them with known threats [38], thereby improving the accuracy and efficiency of threat detection and response.

Extensibility of CyberLLM. The use of data augmentation and retrieval enhancement techniques has played a critical role in enhancing the model's performance. For data augmentation, techniques such as Generative Adversarial Networks (GANs) [10] and Variational Autoencoders (VAEs) [8] can be employed to generate synthetic CVEs and cyber threat descriptions. For retrieval enhancement, integrating knowledge graphs into the retrieval augmentation process [40] can provide structured and contextual information about cyber threats. By leveraging graphs that connect CVEs, threat tactics, and other relevant entities, the model can retrieve more accurate and detailed information, improving its predictive capabilities.

Limitations and Future Works. Our work has a few limitations. (i), Real-time threat intelligence feeds are challenging. Integrating CyberLLM with real-time data sources can enable the model to provide up-to-date insights and predictions about emerging cyber threats. This capability is essential for organizations to respond quickly and effectively to new threats as they arise. (ii) Another important direction for future work is the integration of additional data sources into the model. By incorporating data from various sources, such as threat intelligence platforms, security logs, and external databases, we can enhance the model's understanding of cyber threats and improve its predictive capabilities. (iii) Exploring the application of CyberLLM in automated threat response systems is another promising area for future work. By leveraging the model's capabilities to understand and predict cyber threats, we can develop automated systems that can take proactive measures to mitigate threats. (iv) Finally, exploring the potential for cross-domain applications of CyberLLM is an exciting area for future work. By adapting the model to other domains, such as finance, healthcare, and critical infrastructure, we can leverage its capabilities to address a wide range of security challenges.

7 Conclusion

In this paper, we propose CyberLLM, a large language model specifically designed for mapping CVEs to cyber threat tactics and techniques, whose nature is a multi-label classification problem. It employed a series of data augmentation techniques to enrich the semantic information of CVEs, and incorporates a re-

trieval strategy based on Jaccard distance calculations to improve the contextual understanding of queried CVEs. Through extensive experiments, we demonstrate the superiority of CyberLLM over 7 representative state-of-the-art methods, confirming its effectiveness in mapping CVEs to cyber threat tactics and techniques. Ablation experiments and visualization analysis provide deep insights into its effectiveness, as well as highlight the benefits of incorporating retrieval information. In conclusion, CyberLLM represents a significant advancement in the field of cyber threat intelligence. By automating the mapping of CVEs to attack tactics and techniques, CyberLLM empowers practitioners with the tools necessary to respond swiftly and effectively to emerging cyber threats.

Acknowledgments. This work was supported in part by National Key R&D Program of China (2023YFB3106800), by National Natural Science Foundation of China (62227805, 62072398, 62172405), by the Natural Science Foundation of Jiangsu Province (BK20220075), by the Fok Ying-Tung Education Foundation for Young Teachers in Higher Education Institutions of China (No.201932 18210004), and by Key R&D Program of Zhejiang Province (2023C01039).

References

1. S. Abu et al. Cyber threat intelligence – issue and challenges. *Indonesian Journal of Electrical Engineering and Computer Science*, 2018.
2. B. Ampel et al. Linking common vulnerabilities and exposures to the mitre att&ck framework: A self-distillation approach. *arXiv*, 2021.
3. N. Atre et al. Surgeprotector: mitigating temporal algorithmic complexity attacks using adversarial scheduling. In *SIGCOMM*, pages 723–738. ACM, 2022.
4. M. Bozorgi et al. Beyond heuristics: learning to classify vulnerabilities and predict exploits. In *ACM KDD*, 2010.
5. S. Brown et al. From cyber security information sharing to threat management. In *WISCS@CCS*. ACM, 2015.
6. T. Chen et al. Xgboost: A scalable tree boosting system. In *KDD*. ACM, 2016.
7. T. Chen et al. The lottery ticket hypothesis for pre-trained BERT networks. In *NeurIPS*, 2020.
8. C. Doersch. Tutorial on variational autoencoders. *arXiv*, 2016.
9. P. Gao et al. Enabling efficient cyber threat hunting with cyber threat intelligence. In *ICDE*. IEEE, 2021.
10. I. Goodfellow et al. Generative adversarial networks. *Communications of the ACM*, 2020.
11. O. Grigorescu et al. Cve2att&ck: Bert-based mapping of cves to mitre att&ck techniques. *Algorithms*, 2022.
12. M. Hanna et al. How does GPT-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model. In *NeurIPS*, 2023.
13. H. Ji et al. Sevenllm: Benchmarking, eliciting, and enhancing abilities of large language models in cyber threat intelligence. *ArXiv*, 2024.
14. E. Kiesling et al. The SEPSES knowledge graph: An integrated resource for cybersecurity. In *ISWC*. Springer, 2019.
15. A. Kuppa et al. Linking cve's to MITRE att&ck techniques. In *ARES*, 2021.

16. L. Li et al. An automated alert cross-verification system with graph neural networks for ids events. In *CSCWD*. IEEE, 2024.
17. Z. Li et al. Attackg: Constructing technique knowledge graph from cyber threat intelligence reports. In *ESORICS*. Springer, 2022.
18. Z. Li et al. metanet: Interpretable unknown mobile malware identification with a novel meta-features mining algorithm. *Computer Networks*, 2024.
19. X. Ling, J. Yu, et al. Ddosminer: An automated framework for ddos attack characterization and vulnerability mining. In *ACNS*. Springer, 2024.
20. Y. Liu et al. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, 2019.
21. Y. Liu et al. Cachegen: KV cache compression and streaming for fast large language model serving. In *SIGCOMM*. ACM, 2024.
22. J. Lu and S. Huang. Pr-gnn: Enhancing poc report recommendation with graph neural network. In *ICDE*. IEEE, 2024.
23. O. Mendsaikhan et al. Automatic mapping of vulnerability information to adversary techniques. In *SECURWARE*, 2020.
24. Y. Park et al. A pretrained language model for cyber threat intelligence. 2023.
25. L. O. Prokhorenkova et al. Catboost: unbiased boosting with categorical features. In *NeurIPS*, 2018.
26. J. Read et al. Classifier chains: A review and perspectives. *JAIR*, 2021.
27. M. T. Ribeiro et al. Model-agnostic interpretability of machine learning. *CoRR*, 2016.
28. Z. Song et al. I2RNN: An incremental and interpretable recurrent neural network for encrypted traffic classification. *IEEE TDSC*, 2023.
29. B. E. Strom et al. Mitre att&ck: Design and philosophy. In *Technical report*. The MITRE Corporation, 2018.
30. T. Sun et al. An automatic generation approach of the cyber threat intelligence records based on multi-source information fusion. *Future Internet*, 2021.
31. C. Ten et al. Cybersecurity for critical infrastructures: Attack and defense modeling. *IEEE Trans. Syst. Man Cybern. Part A*, 2010.
32. J. Vig. A multiscale visualization of attention in the transformer model. *arXiv*, 2019.
33. J. W. Wei et al. EDA: easy data augmentation techniques for boosting performance on text classification tasks. In *EMNLP/IJCNLP (1)*, 2019.
34. W. Xiong et al. Cyber security threat modeling based on the MITRE enterprise att&ck matrix. *Softw. Syst. Model.*, 2022.
35. Z. Zhao et al. ERNN: Error-resilient RNN for encrypted traffic detection towards network-induced phenomena. *IEEE TDSC*, 2023.
36. Z. Zhao et al. Ddos family: A novel perspective for massive types of ddos attacks. *Computers & Security*, 2024.
37. Z. Zhao et al. Effective DDoS Mitigation via ML-Driven In-network Traffic Shaping. *IEEE TDSC*, 2024.
38. Z. Zhao et al. FOSS: Towards fine-grained unknown class detection against the open-set attack spectrum with variable legitimate traffic. *IEEE/ACM ToN*, 2024.
39. Z. Zhao et al. Trident: A universal framework for fine-grained and class-incremental unknown traffic detection. In *ACM WWW*, 2024.
40. Q. Zhong et al. Knowledge graph augmented network towards multiview representation learning for aspect-based sentiment analysis. *IEEE TKDE*, 2023.