Security and Safety, Vol. 4, 2024019 (2025)

Security and Safety in Network Simulation and Evaluation

Research Article

Open Access

Other Fields

Uncovering multi-step attacks with threat knowledge graph reasoning

Xiayu Xiang[1], Changchang Ma[2], Liyi Zeng[1], Wenying Feng[1], Yushun Xie[3], and Zhaoquan Gu[1,4,*]

[1] *Peng Cheng Laboratory, Shenzhen 518000, China*
[2] *CHN Energy, Beijing 100000, China*
[3] *University of Electronic Science and Technology of China, Shenzhen 518110, China*
[4] *Harbin Institute of Technology (Shenzhen), Shenzhen 518055, China*

Received: 29 April 2024 / Revised: 25 October 2024 / Accepted: 29 October 2024 / Published online: 25 February 2025

Abstract The rapid advancement of information technologies has significantly intensified the focus on cyberspace security across various sectors. In this evolving landscape, attackers deploy many techniques- including exploits, weakness identification, and complex multi-step attacks- to gain unauthorized access to systems. Conversely, defenders harness insights from a variety of sources to pinpoint potential threats. Prominent public cybersecurity databases such as the Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK), Common Attack Pattern Enumeration and Classification (CAPEC), Common Vulnerabilities and Exposures (CVE), Common Weakness Enumeration (CWE), and Common Platform Enumeration (CPE) provide extensive data on security entities and their interrelations, playing a pivotal role in enriching the understanding of cybersecurity challenges and assisting in comprehensive defensive analyses. However, the semantic cross-analysis of these databases, crucial for identifying obscure threat patterns, remains underexploited. In this study, we amalgamate data from these disparate sources into a cohesive threat knowledge graph and introduce a novel knowledge representation learning approach, A4CKGE (ATT&CK-CAPEC-CWE-CVE-CPE Knowledge Graph Embedding). This method utilizes advanced structural and textual analytics to predict interactions among security entities such as products, vulnerabilities, weaknesses, and multi-step attack sequences, employing complex attack templates generated through a Large Language Model (LLM). Our extensive experiments demonstrate that this approach significantly outperforms existing state-of-the-art methods in effectively predicting these relationships. The findings validate the efficacy of our threat knowledge graph in unveiling hidden connections, thereby highlighting its potential to strengthen cybersecurity defenses substantially.

Keywords Security database, Knowledge graph embedding, Knowledge graph reasoning

Citation Xiang X, Ma C, Zeng L, Feng W, Xie Y and Gu Z. Uncovering multi-step attacks with threat knowledge graph reasoning. Security and Safety 2025; **4**: 2024019. https://doi.org/10.1051/sands/2024019

1 Introduction

The rapid advancement of information technology has significantly heightened the focus on cybersecurity across various sectors. Cyber threat actors employ a wide range of tactics, from deploying exploits and identifying vulnerabilities to conducting sophisticated attacks, all aimed at gaining unauthorized access

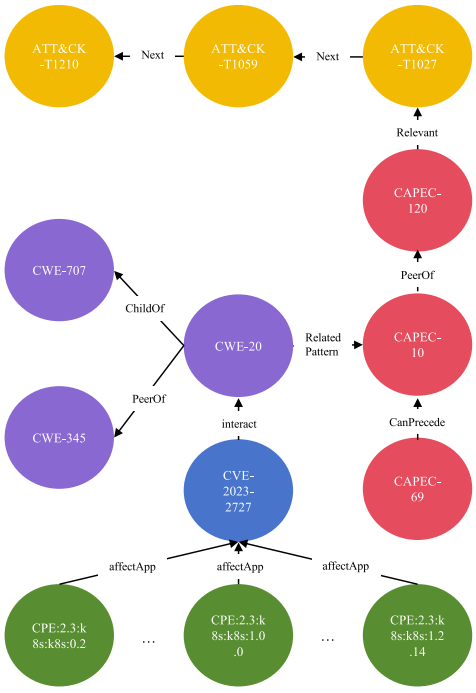


Figure 1. An illustration of threat knowledge graph across ATT&CK, CAPEC, CWE, CVE, and CPE

to target systems [1]. This complex array of methods reflects a constantly evolving cyber threat landscape that persistently tests the limits of existing cybersecurity measures [2]. These threats involve intricate processes: exploits target known vulnerabilities, while the identification of weaknesses involves pinpointing and exploiting fundamental system flaws. Additionally, advanced attacks discreetly infiltrate systems, leading to disruptive effects or the theft of sensitive data [3].

Most importantly, Advanced Persistent Threat (APT) attacks typically involve a series of complex, multi-step techniques designed to achieve the attackers' objectives [4]. These sophisticated attacks are executed in stages, starting with initial infiltration, followed by lateral movement within the network, and culminating in the ultimate goal, be it data exfiltration, system damage, or establishing a long-term presence within the target's infrastructure. Each step is carefully orchestrated to avoid detection and to maintain persistence within the system, making APTs one of the most formidable challenges in cybersecurity today.

The assessment of security risks arising from destructive APT activities is fundamentally reliant on cyber security threat databases. These repositories are invaluable for providing insights into attack patterns, vulnerabilities, weaknesses, and the affected products. Notable examples of such databases include ATT&CK [5], CAPEC [6], CVE [7], CWE [8], and CPE [9]. The entities within these databases are interconnected through a complex network of relationships, as depicted in Figure 1.

For instance, the vulnerability identified as CVE-2023-2727 permits attackers to initiate containers using restricted images. This specific vulnerability is associated with Kubernetes and applies to product versions up to 1.2.14. CVE-2023-2727 is linked to CWE-20, denoted as "Improper Input Validation". This weakness involves the failure of an application to properly validate input, allowing attackers to manipulate the logic of the software, often leading to unauthorized access or other malicious activities. Furthermore, CWE-20 is linked to attack patterns such as CAPEC-10 and CAPEC-120 to improve accuracy. CAPEC-120, in particular, deals with the exploitation of "Obfuscated Files or Information", which is also associated with the ATT&CK-T1059. This technique involves attackers concealing the true nature of their malicious files or scripts to evade detection. Once such an obfuscated file is downloaded and opened by an unsuspecting user, the obfuscation is unraveled by the execution environment (ATT&CK-T1059), which leads to the execution of the underlying malicious code. With access secured, the attacker moves laterally within the network, as outlined in ATT&CK-T1210. This stage may involve the use of credentials harvested by the initial payload to access other systems, alongside the deployment and

execution of similarly obfuscated scripts across the network. This detailed mapping of vulnerabilities and attack techniques underscores the critical role that these databases play in understanding and mitigating cyber threats.

The interconnectivity among various security entities yields substantial information, offering significant benefits for conducting threat analysis [10]. The ability to anticipate associations among these entities in advance could significantly augment the utility of these resources. However, there is a notable absence of comprehensive analyses that integrate these databases or leverage the extensive semantic descriptions contained within the entities. Previous research [11–13] has predominantly focused on predicting CAPEC, CWE, CVE, or CPE independently. Such an approach is inadequate for identifying potential threats from a holistic perspective. Further efforts are needed to develop methodologies that incorporate multiple data sources and employ advanced analytical techniques to provide a more complete understanding of potential security threats.

In this paper, we introduce a novel methodology for predicting relationships between security entities. We utilize a knowledge graph, which we refer to as the "Threat Knowledge Graph" (TKG) [14]. This graph is constructed using data from established knowledge bases such as ATT&CK, CAPEC, CWE, CVE, and CPE. The threat knowledge graph is instrumental in analyzing existing interconnections among the entries and facilitates the prediction of potential interrelations between them using advanced embedded models. This methodology aims to enhance our understanding of the complex network of security threats, thereby improving threat detection and mitigation strategies.

In our research, we apply a representation learning approach known as knowledge graph embedding [15] to convert structural knowledge and textual descriptions into a compact, low-dimensional continuous vector space. This technique effectively preserves the semantic content and complex interactions captured in the knowledge graph. The utility of this approach has been verified across various applications, including link prediction [16], entity resolution [17], and recommendation systems [18]. For our analysis, we integrate vectors representing structural information and textual descriptions to encapsulate each entity within the graph. We developed a novel model, termed A4CKGE, which evaluates and assigns scores to potential relationships within the knowledge graph. This scoring mechanism aids in identifying the most probable future connections. By enumerating absent relationships and ranking their corresponding scores, our model facilitates the identification of the most promising predictive links, thus enhancing the predictive capabilities of security threat analysis.

Building upon our preliminary work, we have expanded our research by employing large language models to generate multi-step ATT&CK attack templates. This innovative approach allows us to predict complex APT behaviors that involve multiple stages. By leveraging the comprehensive attack scenarios constructed through these templates, our methodology enhances the predictive accuracy and depth of analysis in identifying potential multi-step APT activities.

To validate the effectiveness of our enhanced model, A4CKGE, we conducted extensive experiments. We employed the updated data on entities and relationships from the ATT&CK, CAPEC, CWE, CVE, and CPE databases to construct a threat knowledge graph, which includes 66 121 entities – comprising 201 ATT&CK techniques, 559 CAPEC patterns, 933 CWE entries, 31 187 CVE records, and 33 214 CPE identifiers. Our experimental results on this graph demonstrate significant improvements over the current state-of-the-art [11]. Specifically, we achieved a 6.5% increase in the Hits@10 metric, which measures the accuracy in identifying missing relationships among existing entities. Notably, our approach extends beyond the capabilities of the model cited in the referenced work by incorporating four additional knowledge bases: ATT&CK, CAPEC, CVE, and CPE. This expansion allows for a more robust and comprehensive analysis, leading to superior predictive performance in our model.

In summary, we make the following contributions:

(1) We explore the knowledge graph representation of ATT&CK, CAPEC, CWE, CVE, and CPE, and delved into reasoning tasks utilizing this expansive threat knowledge graph. A novel knowledge graph embedding model is developed, named A4CKGE, which effectively transforms the structural and textual information into a vector space representation.
(2) We are the first to make use of large language models to generate multi-step ATT&CK attack templates. These templates serve as vital inputs for training datasets, enabling the creation of a model capable of reasoning about and predicting complex, multi-stage APT behaviors.
(3) We conduct extensive experiments to evaluate the effectiveness of our knowledge graph embedding method in multi-step reasoning – relations between security entities within threat knowledge graph.

Table 1. An example of textual description of CVE-2023-2727, and its associated entities

CVE ID	CVE-2023-2727
CVE description	Users may be able to launch containers using images that are restricted by ImagePolicyWebhook when using ephemeral containers. Kubernetes clusters are only affected if the ImagePolicyWebhook admission plugin is used together with ephemeral containers.
Associated CPE	cpe:2.3:a:kubernetes:kubernetes:*:*:*:*:*:*:*:* (Up to version 1.24.14)
Associated CWE	CWE-20: Improper Input Validation
CWE description	The product receives input or data, but it does not validate or incorrectly validates that the input has the properties that are required to process the data safely and correctly.
CWE extended description	Input validation is a frequently used technique for checking potentially dangerous inputs in order to ensure that the inputs are safe for processing within the code, or when communicating with other components. When software does not validate input properly, an attacker can craft the input in a form that is not expected by the rest of the application. This will lead to parts of the system receiving unintended input, which may result in altered control flow, arbitrary control of a resource, or arbitrary code execution.
Associated CAPEC	CAPEC 120
CAPEC description	The adversary utilizes a repeating of the encoding process for a set of characters (that is, character encoding a character encoding of a character) to obfuscate the payload of a particular request. This may allow the adversary to bypass filters that attempt to detect illegal characters or strings, such as those that might be used in traversal or injection attacks. Filters may be able to catch illegal encoded strings, but may not catch doubly encoded strings.
Associated ATT&CK technique	T1027
Technique description	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. This is common behavior that can be used across different platforms and the network to evade defenses.

The remainder of this paper is organized as follows: Section 2 introduces preliminary security databases. Section 3 reviews related work. Section 4 elaborates on the technical details of our approach. Section 5 reports our experiments and findings. Section 6 summarizes our contributions and plans for future work.

2 Preliminary

In this section, we describe five security databases: ATT&CK, CAPEC, CVE, CWE, and CPE. Table 1 shows a textual example between the entries of different threat databases.

The MITRE Adversarial Tactics, Techniques, and Common Knowledge is an evolving, globally accessible knowledge base of adversary tactics and techniques observed in real-world incidents. ATT&CK provides a rich lexicon and taxonomy for cyber adversaries' behavior, detailing each aspect of their life cycle, from initial access through exfiltration. The framework organizes these tactics and techniques across a matrix that security practitioners use to understand and classify attacks and assess an organization's risk. For each ATT&CK entry in the ATT&CK database, we consider the tactical goal of

the adversary, and the specific techniques employed as the critical features in the experiments. Each technique also includes references to other databases such as CAPECs, which relate to the general attack pattern employed by the technique.

The Common Attack Pattern Enumeration and Classification is a publicly available and comprehensive dictionary of known patterns of attack. These patterns are categorized and described from the perspective of the attacker, thus offering insights into how vulnerabilities are exploited, how certain types of attacks are launched, and how unauthorized access can be obtained. For every CAPEC entry in the CAPEC database, we deem the description of the attack pattern, the relationships, and related weaknesses as the critical features in the experiments.

Common Vulnerabilities and Exposures is an enumeration of publicly acknowledged cybersecurity vulnerabilities. Each entry encompasses an identification number, a description, and at least one CPE reference. CVE's common identifiers enable data exchange between security products and provide a baseline index point for evaluating the coverage of their respective security tools. For every CVE entry in the CVE database, we deem the description of the vulnerability, weakness enumeration, and relevant known affected software configurations as the critical features in the experiments.

Common Weakness Enumeration is a community-developed list of software and hardware weakness types. It serves as a common language for describing these vulnerabilities and provides a baseline standard for vulnerability and weakness identification, mitigation, and prevention efforts. For every CWE entry in the CWE database, we deem the textual description, relations of other CWEs, and related attack patterns (i.e., the specific attack methods pointed at this CWE) as the critical features in the experiments.

Common Platform Enumeration is a structured naming scheme for information technology systems, software, and packages. It provides a method for correlating data from different databases that refer to the same entities, which is crucial for effective security management of IT systems. For every CPE entry in the CPE database, we consider the part (application, hardware, or operating system), the vendor (e.g., Kubernetes), the product (e.g., Kubernetes), and the version (e.g., 1.24.14) as the critical features in the experiments.

3 Related work

3.1 Security database-based research

The extensive body of research explores the use of cybersecurity knowledge bases to understand emerging threats, vulnerabilities, and system weaknesses [19]. Notably, Su *et al.* [20] apply a range of machine learning algorithms to predict upcoming vulnerabilities in specific software applications, primarily focusing on the CVE and CWE databases. Similarly, another study [21] introduces a vulnerability mining algorithm that employs data mining techniques to identify and analyze core characteristics of software vulnerabilities.

However, leveraging threat databases presents significant challenges. Often, these databases are inconsistent or not immediately usable [21], which has led researchers to consider the integration of additional textual information into the existing data representations. For instance, the DeepWeak [11] project illustrates common software weaknesses and their interconnections through a knowledge graph, developing a methodology to learn knowledge representations. This involves embedding both textual descriptions and their associations within the knowledge graph into a semantic vector space, which is then used for knowledge acquisition and inference. Conversely, Guo *et al.* [12] expanded their threat databases to include CVE and CAPEC, introducing a text-enhanced graph attention network model. Their extensive experimental evaluations confirm the effectiveness of their model in predicting security entity relationships and enhancing the detection of missing relationships.

Our approach distinguishes itself from these studies by not only incorporating additional security databases but also by emphasizing the additional descriptions derived from graph nodes. As a result, we can alleviate the negative impact of the independence of each triple and accumulate a broader scope of security knowledge from databases such as ATT&CK, CAPEC, CWE, CVE, and CPE.

3.2 Knowledge graph embedding

Knowledge graph embedding is a significant technique for transforming multi-relational data into a continuous, low-dimensional vector space, as documented in [15]. This method is particularly valuable in threat knowledge bases where entities are interconnected through various associations, and knowledge graph techniques such as reasoning are employed to unearth these connections. Traditional knowledge graph embedding methods generally emphasize the structural similarities among entities, exemplified by TransE [22], TransH [23], DistMult [24], ComplEx [25], and RotatE [26].

There has been a growing interest in enhancing these embeddings with rich semantic descriptions of entities. The research presented in [10] explores reasoning for missing relationships among software weaknesses using a knowledge graph representation learning technique that merges both descriptive and structural data into a dense vector space. Furthermore, another study [27] advances this approach by introducing a knowledge graph embedding technique that encapsulates both symbolic relational and descriptive information of software security entities into a continuous vector space. These advancements in entity and relationship embeddings demonstrate their potential in discerning complex relationships among software security entities. Additionally, the PTransE model [28] integrates multi-step reasoning paths between entities, significantly improving the ability to capture intricate relational patterns and infer obscured associations.

Our knowledge graph embedding methodology utilizes a robust model that generates embeddings based on both structural and descriptive information, incorporating additional knowledge bases for a comprehensive analysis. By including the multi-step reasoning capabilities of PTransE, our approach not only extends beyond the methodologies described in [11] but also shows superior predictive performance in identifying missing relationships, making it a powerful tool in the field of cybersecurity.

4 Approach

In this section, we first formulate the research problem, then present an overview of our approach, and finally describe the technical details.

4.1 Problem formulation

Our work focuses on developing a robust knowledge graph embedding method designed to enhance reasoning capabilities over security entity relations. This is achieved by leveraging the interconnected sources from ATT&CK, CAPEC, CWE, CVE, and CPE, along with the corresponding descriptions for each entry. Furthermore, our methodology extends to supporting reasoning tasks for multi-step attack patterns, particularly those associated with complex APTs. Our approach interprets the embedding of threat knowledge graphs through a dual process that integrates both structural and textual embeddings. The resulting knowledge graph embedding model is capable of transforming these relations and descriptions into a semantically coherent vector representation. This transformation is facilitated by multi-step attack inputs derived from a sophisticated large language model, significantly bolstering the model's capability to perform critical reasoning tasks such as relationship prediction.

4.2 Overview

The architectural framework of our methodology is illustrated in Figure 2. The core principle of our approach is to concurrently learn the structural and descriptive representations from the threat knowledge graph. Specifically, the embeddings of entities and their relationships are conceptualized as k-dimensional real-valued vectors within the space R_k. Each entity is characterized by two distinct types of representations: one based on its structure within the graph and another derived from its textual description. The variables h_s and t_s represent the structural representations of the head and tail entities respectively, which are learned based on their interrelations. Conversely, h_d and t_d denote the description-based representations of the head and tail entities, respectively, obtained from analyzing their corresponding textual descriptions. The relationship type is denoted by the variable r, which is a member of the set R. Additionally, the multi-step attack scenarios are generated using ChatGPT-4, employing intricately designed prompts to generate different multi-step attack templates for different APT groups.

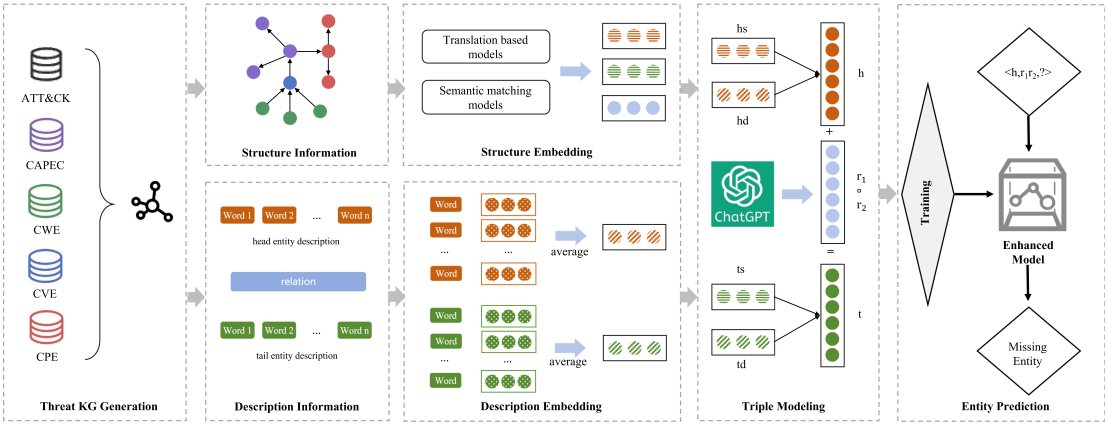


Figure 2. Overview of our approach

4.3 Threat knowledge graph generation

The initial stage of our methodology entails the development of a threat knowledge graph, which is constructed using data from the ATT&CK, CAPEC, CWE, CVE, and CPE knowledge bases, as illustrated in Figure 2. This threat knowledge graph is formally defined as $G = (E, R, S)$, where E represents the set of all entities, R denotes the set of all relations, and S is the set of all triples within the graph. Each element within the set of triples, S, is structured as $S = \langle h, r, t \rangle$, where '$h$' is the head entity, '$r$' identifies the relation, and 't' is the tail entity. Such structure facilitates the systematic analysis and representation of the relationships and interdependencies among the various entities in the graph, providing a robust foundation for further analytical tasks.

The above process involves several technical steps: First, entities are extracted from the aforementioned databases and classified according to their nature (e.g., attack patterns, vulnerabilities). Next, relationships between entities are identified based on the documented interdependencies in these databases, such as how specific vulnerabilities can lead to particular attack patterns. Finally, these entities and their relationships are encoded into a graph format, where machine learning algorithms can be applied to learn embeddings that reflect both the structural position and textual descriptions of the entities.

4.4 Structure-embedding generation

As depicted in Figure 2, we leverage the entities and their interconnections within our established knowledge graph to employ advanced knowledge graph embedding techniques. These techniques are crucial for capturing the intricate structure of the graph and include translational distance-based models, such as TransE, and semantic matching-based models, such as DistMult and ComplEx. The translational distance-based models interpret relationships as translations in the embedding space, where the embedding of a head entity plus the relation vector should approximate the embedding of the tail entity. On the other hand, semantic matching-based models employ similarity functions to match the embeddings of entities and relations, effectively capturing the types of interactions between entities within the graph.

These models are chosen for their efficacy in extracting pivotal structural features from the knowledge graph. They transform each entity and relation into vectors within a continuous vector space. Such transformation is achieved through embedding techniques where each entity and relationship are represented as an m-dimensional vector. The dimensionality 'm' is a critical hyperparameter, which, in our experiments, is optimally set to 100. This setting allows for a balance between sufficient complexity to capture nuanced relationships and computational efficiency.

Table 2. An illustration of Urpage APT group's TTP

Group name	ATT&CK technique	Description
Urpage	T1566	Initial Access: The Urpage organization employs phishing emails and social engineering tactics for initial access.
	T1021	Lateral Movement: Once initial access is obtained, the Urpage organization utilizes methods such as Remote Desktop Protocol (RDP) and valid accounts to perform lateral movement.
	T1027	Persistence: The Urpage organization establishes persistence using mechanisms like reverse shells and scheduled tasks.
	T1056	Data Collection: The Urpage organization employs data collection techniques including keyloggers and screen capture tools.
	T1048	Exfiltration: The Urpage organization exfiltrates data using protocols such as FTP and SMB.

4.5 Description-embedding generation

As depicted in Figure 2, this section of our study focuses on embedding the textual descriptions derived from the ATT&CK, CAPEC, CWE, and CVE databases. The objective is to capture the complex syntactic and semantic features inherent in these descriptions, which are crucial for a deeper understanding of the underlying security concepts.

The process begins with the tokenization of the textual descriptions, where we utilize the Gensim toolkit, a prominent Python library for natural language vector embedding. This step includes the removal of stop words and other non-essential elements that do not contribute meaningfully to the semantic representation. Following tokenization, we employ the Word2vec model to embed each word into a vector space. Word2vec, is a powerful method for translating words into a *100*-dimensional vector space, thus preserving semantic and contextual word relationships. Once we have the vector representations from Word2vec, we aggregate these vectors to form a single composite vector for each entity's description. This is achieved by computing the average of the word embedding vectors contained within each description. The formula can be expressed as:

$$h_d = \frac{x_1 + x_2 + \ldots + x_n}{n} \tag{1}$$

where x_k denotes the embedding of words in a sentence, and h_d denotes the sentence embedding features.

4.6 Multi-step attacks generation

Following the embedding, our research uses ChatGPT-4 to generate detailed templates for multi-step attacks associated with various threat families [29]. This innovative approach leverages the capabilities of ChatGPT-4 to interpret and generate complex attack scenarios that align with the Tactics, Techniques, and Procedures (TTPs) used by specific threat actors.

To operationalize this, we utilized prompts specifically designed to extract and articulate the TTPs employed by designated threat organizations. An example prompt used in this process is: This prompt guides ChatGPT-4 to generate responses that not only enumerate the TTPs but also link them with relevant identifiers from the ATT&CK framework, thus ensuring that the generated templates are both informative and actionable. The example is shown in Table 2.

The integration of ChatGPT-4 allows us to systematically generate templates that simulate real-world attack patterns. Each generated template provides a structured outline of potential attack vectors, which can be used to enhance threat detection algorithms and train security personnel. This method significantly contributes to the depth and breadth of our threat knowledge graph by adding a dynamic layer of predictive and preventive capabilities that are crucial for advanced cybersecurity defense mechanisms.

Table 3. Typical scoring functions (adapted from [30])

Scoring function	Definitions
TransE	$\|\|h + r - t\|\|$
TransH	$\parallel h - w_r^T h w_r + r - \left(t - w_r^T t w_r\right) \parallel$
DistMult	$\langle h,\ r,\ t \rangle$
ComplEx	$Re(\langle h,\ r,\ \text{conj}(t) \rangle)$

4.7 Entity Prediction

In our research, we employ KGE models to map each entity and relation from the threat knowledge graph into a lower-dimensional space, enhancing computational efficiency and clarity of complex relational data. Knowledge Graph Embedding models are broadly categorized into two types: translational distance-based models and semantic matching-based models. The primary distinction between these models lies in the formulation of their scoring functions, which fundamentally govern the effectiveness of the embeddings.

Translational distance-based models, such as TransE, conceptualize relations as translations in the embedding space. In this model, the relationship is successful if the embedding of the head entity plus the relation vector closely approximates the embedding of the tail entity, minimizing the distance in the embedding space.

Semantic matching-based models, including DistMult and ComplEx, utilize similarity-based scoring functions. These models compute scores by evaluating the similarity between the mathematical transformations of the embeddings of entities and relations. For instance, DistMult applies a bilinear function, which is particularly effective for symmetric relations, while ComplEx extends this to capture complex and asymmetric relationships by incorporating complex-valued embeddings.

Table 3 (adapted from [30]) delineates various scoring functions employed by these models, illustrating how each model integrates different aspects of relational data into the embedding process. This comparative analysis aids in understanding the nuanced capabilities and suitable applications of each KGE model, providing a solid foundation for selecting the appropriate model based on the specific characteristics and requirements of the threat knowledge graph.

Our approach then harnesses both structure-based and description-based embeddings to deepen the understanding of entity relationships within the threat knowledge graph. Intuitively, the embeddings of the head entity and its corresponding relation should be closely aligned with those of the tail entity in both structural and descriptive aspects.

In our methodology, the scoring function of the KGE models is pivotal to the differentiation between valid (positive) and invalid (negative) relationships within the threat knowledge graph. This scoring function aims to assign higher scores to positive triples $f(h, r, t)$ that reflect facts, compared to negative triples $f(h', r', t')$, which are artificially constructed and represent non-factual relationships.

The primary objective is to minimize the loss function across the set of training triples. The essence of this optimization process is to ensure that the score of a positive triple is not just greater than that of a negative triple, but that it exceeds it by at least a pre-defined margin. This margin creates a buffer that reinforces the distinction between true and false facts.

The loss function, denoted by L, is formally articulated as the sum of the differences between the margin and the scores of the positive and negative triples, subject to the constraint that this difference is non-negative. It is represented mathematically as:

$$L = \sum \left[\max\left(0,\ \text{margin}\ - f(h, r, t) + f\left(h', r, t'\right)\right)\right]_+ \tag{2}$$

In this formula, h and h' symbolize the head entities for the positive and negative triples respectively, r indicates the relation, and t and t' denote the tail entities. The rectifier function, denoted as $[\cdot]+$, ensures that only positive values contribute to the loss, enforcing the margin constraint. The training process iteratively adjusts the embeddings to minimize this loss function, effectively learning to encode true knowledge while distancing itself from the false or non-existent relations as represented by the negative triples.

The enhanced representation of a head entity within our KGE framework is denoted as E_h, which synergistically combines the structure-based and description-based vectors. This concatenation enriches the entity's representation with both topological and descriptive information, thereby capturing a more holistic view of the entity's characteristics. The composite vector E_h, as depicted in the equation $E_h = h_s \oplus h_d$, integrates the structural representation h_s with the textual description vector h_d through a concatenation operation symbolized by $\oplus$.

Our methodology for achieving this integrated representation is direct concatenation. Initially, we establish the m-dimensional structure-based vectors and subsequently append the n-dimensional description-based vectors to them. This concatenated vector E_h then serves as the input feature for the KGE models. The model training proceeds through several epochs until the loss function exhibits signs of convergence and stability.

Crucially, to maintain the integrity of the structural information and prevent it from being overshadowed by the contextual data within the description-based vectors, we introduce a randomization procedure during training. By occasionally discarding portions of the concatenated vectors—specifically, the tails of the description-based vectors—we ensure that the significance of the structure-based vectors remains prominent within the model. This technique preserves the balance between structural and descriptive elements in the embeddings, facilitating an accurate and comprehensive representation in the embedding space.

In the optimization process of our KGE model, the loss function is pivotal for training the embeddings to accurately reflect the relationships within the graph. The revised loss function can be articulated as follows:

$$L = \sum \left[\max\left(0, \text{margin} - f\left(E_h, r, E_t\right) + f\left(E_{h'}, r, E_{t'}\right)\right)\right] \tag{3}$$

Here, E_h and $E_{h'}$ represent the concatenated vectors of the head entities for positive and negative triples, respectively, while E_t and $E_{t'}$ correspond to the tail entities. In the context of the TransE model, the scoring function f is defined by the L2 norm of the dissimilarity measure for each triple, expressed as $||h+r-t||_2$. This norm captures the distance between the tail entity's vector and the vector resulting from the head entity's vector translated by the relation's vector. The difference in the scores between positive and negative triples is then adjusted by adding a margin. This margin is a crucial hyperparameter that enforces a minimum separation between the scores of valid and invalid relationships.

The resultant difference is subjected to a hinge loss function, denoted as $[x]_+ = \max(0, x)$, which ensures that the loss function only considers positive discrepancies, aligning with the margin constraint and reinforcing the embeddings' differentiation.

To refine the embeddings and hyperparameters, we employ batch gradient descent coupled with the Adam optimization algorithm. This sophisticated method allows for efficient and effective updates by computing adaptive learning rates for each parameter. The utilization of batch gradient descent not only expedites the training process but also facilitates a more robust convergence towards the global optimum, enhancing the model's ability to generalize from the training data to accurately predict and infer the relationships within the knowledge graph.

Our research aims to leverage the advanced capabilities of PTransE model to predict multi-step attack sequences within a threat knowledge graph. It stands out for its ability to capture complex relational paths, which is critical when dealing with sophisticated attack patterns that involve multiple intermediate steps or entities.

For a given multi-step path from a head entity h through a sequence of relations $r_1, r_2, \ldots r_n$ to a tail entity t, It aims to capture this by embedding path-based translations. The path is represented as a composition of relations $r_1 \circ, r_2 \circ, \ldots \circ r_n$, and the model's objective is to ensure that h plus the composed relation approximates t:

$$h + (r_1 \circ r_2 \circ \ldots \circ r_n) \approx t \tag{4}$$

To train our model, we derive a dataset of multi-step attack templates using ChatGPT-4. These templates, which describe sequences of TTPs associated with particular threat families, are then incorporated into the threat knowledge graph's relations. We split these templates into a training set, to teach the model the complex patterns associated with multi-step attacks, and a testing set, to evaluate the model's predictive accuracy.

The model's performance is quantified by a loss function that is adapted to reflect our approach. The function is designed to minimize the distance between the composite embedding of the multi-step path and the actual tail entity in the graph for positive instances while maximizing this distance for negative instances–those that do not exist or are incorrect:

$$L = \sum_{(h,r,t)\in S} \sum_{(h'r',t')\in S'} [\max(0,\ \text{margin} - \parallel h + (r_1 \circ r_2 \circ \ldots \circ r_n)\ \ - t \parallel_2 + \parallel h' + (r_{1'} \circ r_{2'} \circ \ldots \circ r_{n'}) - t' \parallel_2)] \quad (5)$$

By minimizing this loss function, we aim to refine the predictive capability of our model, enabling it to accurately forecast multi-step attack scenarios. The ultimate goal is to provide a sophisticated tool for cybersecurity analysts, aiding in the anticipation and mitigation of complex threats.

5 Experiment

In this section, we first outline the experimental setup. Next, we present extensive evaluations to highlight the efficacy of our approach – a translation-based, description-embedded knowledge graph reasoning methodology – in accurately predicting missing entities.

5.1 Experiment setup

Environment. We implement A4CKGE in Python 3.7. All experiments run on two NVIDIA GeForce RTX 2080 GPU machines, and the video memory size is 16 GB. The CPU is Intel(R) Core(TM) i7-9700K CPU and the total memory of the machine is 32 GB.

Dataset. In this section, we describe the five security databases and the data composition of our threat knowledge graph.

ATT&CK dataset. A total of 201 Enterprise techniques were acquired. It has a technique ID and name, including a description, related tactics, platforms, and permission required. The relations between ATT&CK and CAPEC are 2537.

CAPEC dataset. For this study, we used a dataset that includes 559 attack patterns. The attributes of each attack pattern in this dataset include ID, name, description, relationship, and other attributes. The relations between CAPEC and CWE are 1212.

CWE dataset. The dataset contains 933 weaknesses. Each weakness has details such as a unique identifier, name, state, weakness abstraction, and description that have been utilized in our study. The relations between CWE and CVE are 10 564.

CVE dataset. We extracted the vulnerability data from 2018 to 2019, totaling 31 187 CVE vulnerabilities [29]. Each entry in this dataset is assigned a unique ID and includes details like the CVE ID, name, severity level, and a brief description of the vulnerability. The relations between CWE and CPE are 99 097.

CPE dataset. The CPE entries correspond to vulnerability data from 2018 through 2019, totaling 33 214 entities. This dataset consists of unique identifiers, including hardware, software, operating systems, vendor, product, version number, and several others.

The comprehensive statistics outlining the distribution of these knowledge bases are illustrated in Tables 4 and 5. The number of each type of relationship is shown in detail in Table 4, and the number of entity types is shown in detail in Table 5. Our threat knowledge graph contains 66 121 entities, 196 994 triples, and 10 relations. We select 80% of the triples for the training set, 10% for the validation set, and 10% for the testing set. Since our training and test sets are large, the evaluation metric MR values will tend to be high.

In addition to generating positive samples, we also need to generate negative samples that are distinct from the positive sample collections. The multi-step templates generated by the LLM are considered positive records, and to effectively train the model, a sufficient number of negative samples is required. To create these, we randomly select subsequent attack techniques that are not part of the complete multi-step collection, ensuring a diverse set of negative samples for training.

Evaluation. Metrics To conduct a quantitative assessment of the embedding, we typically rely on the following metrics, with their mathematical definitions provided below: (A). Mean Reciprocal Rank

Table 4. Distribution statistics of relations in our TKG

Relation type	Number
affectOS	67 982
affectApp	35 582
affectHardware	61 013
Interact	27 317
relatedPattern	1212
CanPrecede	321
ChildOf	1601
ParentOf	1601
PeerOf	135
Next	230

Table 5. Distribution statistics of entities in our TKG

Knowledge bases	Number
ATT&CK	201
CAPEC	559
CWE	933
CVE	31 187
CPE	33 214

(MRR), which measures the mean of the reciprocals of the ranks of positive triples; (B). Mean Rank (MR), which calculates the average rank of the positive triples; and (C). Hits-at-N-score (Hits@N), which quantifies how many positive triples rank within the top N positions. Mathematically expressed in Equations (6)–(8).

$$MRR = \frac{1}{|T|}\sum_{i=1}^{T}\frac{1}{\text{rank}_{\langle h,r,t\rangle_i}} \tag{6}$$

$$MR = \frac{1}{|T|}\sum_{i=1}^{T}\text{rank}_{\langle h,r,t\rangle_i} \tag{7}$$

$$Hits@N = \frac{1}{|T|}\sum_{i=1}^{T}1\left[\text{rank}_{\langle h,r,t\rangle_i} \leq N\right] \tag{8}$$

In each triple, our goal is to predict the absent entity using the other entity and relation and subsequently generate an ordered list of potential candidates for this missing entity. We utilize metrics such as Hits@1, Hits@3, and Hits@10 to evaluate their predictive performance. Furthermore, the Mean Rank and Mean Reciprocal Rank are also used as evaluative measures.

5.2 RQ1: Which is the most effective model in our structural relation ranking task?

We designed a comprehensive threat knowledge graph incorporating CVE, CWE, CAPEC, and CPE. To explore the capabilities of structure-based embeddings in our graph, we implemented two popular KGE models: TransE and DistMult. These models were chosen to evaluate their effectiveness in capturing and embedding the full structural intricacies of our knowledge graph. Furthermore, our research extends to detailed assessments of relationships within the graph. Specifically, we closely examined the interconnections between CVE and CWE, as well as the relationships within the CWE entities. In scenarios where the threat knowledge graph was incomplete–lacking either CPE or CAPEC entities and their corresponding relationships–we conducted experiments to assess how different models perform using structural features in these limited conditions.

Table 6. Evaluation results of embedding using different embedding models for TKG (adapted from [30])

Triple type	Model	Dataset	MRR	MR	Hits@10	Hits@3	Hits@1
ALL	TransE	ALL	0.354	5040.016	0.486	0.381	0.286
		No-CPE	0.052	13 909.149	0.096	0.077	0.019
		No-CAPEC	0.355	4943.392	0.491	0.381	0.287
	DistMult	ALL	0.205	7770.024	0.334	0.222	0.138
		No-CPE	0.104	6837.032	0.231	0.097	0.047
		No-CAPEC	0.219	7 520 358	0.355	0.230	0.153
CVE-CWE	TransE	ALL	0.077	11 284.297	0.196	0.075	0.025
		No-CPE	0.004	16 158.898	0.009	0.003	0.002
		No-CAPEC	0.075	11 495.046	0.190	0.069	0.024
	DistMult	ALL	0.101	16 724.084	0.225	0.102	0.044
		No-CPE	0.078	7910.668	0.185	0.069	0.028
		No-CAPEC	0.104	15 791.059	0.228	0.098	0.051
CWE internal	TransE	ALL	0.305	3411.406	0.530	0.457	0.120
		No-CPE	0.315	1614.365	0.596	0.519	0.079
		No-CAPEC	0.289	4616.500	0.457	0.393	0.156
	DistMult	ALL	0.128	2376.068	0.241	0.126	0.068
		No-CPE	0.258	1413.712	0.474	0.252	0.165
		No-CAPEC	0.137	4238.609	0.214	0.130	0.090

Throughout these experiments, we maintained a consistent set of parameters across different models to ensure a fair comparison and to validate the robustness and effectiveness of our threat knowledge graph structure. The outcome of these experiments not only verified the feasibility of our graph design but also aided in selecting the most effective models based on their performance under varied conditions. This rigorous approach ensures that we identify and implement the most reliable models for enhancing cybersecurity measures through informed threat prediction and analysis.

Based on the findings detailed in Table 6, the TransE model demonstrates superior performance across several key evaluation metrics when applied to the full structural data of our threat knowledge graph. Specifically, when there is no missing mapping data, TransE outperforms the DistMult model by notable margins: 25% in the MRR, 15.2% in Hits@10, 15.9% in Hits@3, and 14.8% in Hits@1. This trend of TransE's superior performance extends to scenarios where the dataset is missing CAPEC entities and relationships. In tests that focus on the internal relationships within the CWE category, the TransE model surpasses DistMult by 12.2% in MRR, 24.3% in Hits@10, 26.3% in Hits@3, and 6.6% in Hits@1.

Conversely, in the dataset lacking CPE entities and relationships, the DistMult model exhibits better performance than TransE in capturing the full graph structural information and analyzing the relationships between CVE and CWE. However, it underperforms relative to TransE in handling the internal CWE relationships. This discrepancy likely stems from the absence of CPE information altering the overall graph structure more significantly in terms of external relationships among various entities, while having a lesser impact on the internal structures of CWE relationships.

The data in Table 6 (adapted from [30]) strongly suggests that the TransE model is more effective for structure-based embedding across our threat knowledge graph. It is practical to train our graph using various embedding models, and the findings indicate that a complete dataset typically yields better results than one with missing entities and relationships. Consequently, a more comprehensive threat knowledge graph enhances the efficacy of structure-based embedding representations. Moving forward, we plan to use the complete graph data as the foundation for our forthcoming experiments.

In our advanced A4CKGE model approach, we have chosen to adopt the TransE model as the foundational framework. This model will be used to integrate and train both structure-based and description-based embeddings, aiming to refine and enhance the predictive capabilities of our knowledge graph in cybersecurity applications.

Table 7. Evaluation results on relation ranking for TKG (adapted from [30])

Metric	Model	MRR	MR	Hits@10	Hits@3	Hits@1
Baseline1	TransE (structure)	0.354	5,040.016	0.486	0.381	0.286
Baseline2	TransE (structure)	0.353	5,928.975	0.485	0.379	0.286
DeepWeak	TransCat (concat)	0.384	1,903.892	0.523	0.414	0.301
Our approach	TransHCat ($\oplus$) [(1)]	0.367	1,973.223	0.519	0.398	0.289
Our approach	A4CKGE ($\oplus$) [(1)]	**0.437**	**1,765.627**	**0.588**	**0.476**	**0.358**

[(1)]$\oplus$ means (structure $\oplus$ description).

5.3 RQ2: How effective is our approach in structural and textual relation ranking task, compared with the previous methods?

In our research, we established two baseline experiments to set a standard for evaluating more complex models. Baseline 1 utilizes the TransE model, while Baseline 2 employs the TransH model. Both of these baseline experiments focus solely on structure-based representations to embed both entities and relations within our threat knowledge graph.

Expanding beyond these initial baselines, we introduced DeepWeak, a model that integrates both structure-based and description-based representations for the relation ranking task. This model aims to leverage the additional context provided by descriptive data to enhance the predictive accuracy and robustness of the embeddings.

Furthermore, we adapted the TransH model into a new variant, named TransHCat, which similarly utilizes a combination of structure-based and description-based representations. This adaptation is designed to test the efficacy of TransH when augmented with descriptive data, providing a comparative framework against our more traditional TransH implementation.

Our advanced A4CKGE model, which primarily uses the TransE model as its foundational base, is positioned against both DeepWeak and TransHCat in our comparative experiments. For these experiments, we utilized the complete dataset of our threat knowledge graph. We strategically divided the dataset into training, validation, and testing sets, allocating 80% of the triples for training, 10% for validation, and 10% for testing. This distribution ensures that our models are trained on a comprehensive dataset while still allowing for rigorous testing and validation of their performance.

The effectiveness of these models is evaluated using various metrics as outlined in Section 5.1 of our research. These metrics are critical for quantifying the performance improvements provided by integrating structure-based and description-based representations in our models, thereby informing our understanding of the best practices for embedding entities and relations in cybersecurity knowledge graphs.

According to the data presented in Table 7 (adapted from [30]), our A4CKGE model significantly outperforms the two baseline methods across all key evaluation metrics. Specifically, it achieves an MRR that is 0.083 and 0.084 points higher than Baseline 1 (TransE) and Baseline 2 (TransH), respectively. In terms of the Hits@10 metric, our model shows a substantial improvement, registering 10.2% and 10.3% higher scores compared to the baseline models.

Further comparative analysis with DeepWeak's TransCat model reveals that the A4CKGE model also excels in finer-grained metrics. It shows improvements of 0.053 points in MRR, and 6.5%, 6.2%, and 5.7% in the Hits@10, Hits@3, and Hits@1 metrics, respectively. These results underscore the effectiveness of our model in accurately ranking relationships within the threat knowledge graph (see Table 8).

The comparison highlights the advantages of combining structure-based embeddings with description-based embeddings. The inclusion of descriptive information about graph entities provides supplementary data that enriches the feature space for embeddings, leading to more robust and accurate representations. This approach not only enhances the performance in threat knowledge graphs but could be applicable in broader contexts where rich textual information is available.

Our analysis also indicates that the superior performance of the A4CKGE model over the TransHCat model confirms the suitability of the TransE model as the foundational embedding mechanism within our knowledge graph structure. The choice of the TransE model supports the integration of diverse relations-both external and internal-which work synergistically to optimize the embedding results.

Table 8. Evaluation results on multi-step prediction for TKG

	MRR	MR	Hits@10	Hits@3	Hits@1
Test 1	0.295	98.817	0.131	0.196	0.328
Test 2	0.287	98.801	0.133	0.189	0.320
Test 3	0.291	98.799	0.134	0.195	0.323
Average	0.291	**98.806**	0.133	0.193	0.324

Overall, the results from these experiments suggest that utilizing a comprehensive knowledge graph, enriched with descriptive information, significantly boosts the effectiveness of the embeddings. The careful selection of the appropriate embedding model further influences the quality of the experimental outcomes. Our A4CKGE model, by synthesizing multiple factors, achieves the best performance, demonstrating its potential as a leading solution in the field of cybersecurity threat intelligence.

5.4 RQ3: Can our approach accurately predict multi-step attacks?

In this section, our objective was to employ a three-step predictive approach, aiming to utilize two given ATT&CK techniques and their connecting relationships to predict a subsequent third attack entity. Our setup involved structuring a three-step prediction model within our threat knowledge graph. Initially, we identify two attack techniques, referred to as "Attack A" and "Attack B". These techniques are linked through specific relationships, denoted as "Relation 1" and "Relation 2". Based on the structure and relational data embedded within the graph, our goal was to predict a third ATT&CK technique, "Attack C", which logically follows from the first two.

To achieve this, we utilized the A4CKGE model to effectively handle multi-relational data. This model computes the likelihood of a sequence leading to a potential subsequent attack technique, essentially forecasting "Attack C" from the known entities "Attack A" and "Attack B" through their respective relationships. We divided the dataset into training, validation, and testing sets. The training set was used to teach the model the fundamental patterns and sequences of attack techniques, focusing on the interconnections between specific pairs of techniques and their resulting sequences. Validation was conducted intermittently to tune the model parameters and optimize performance. Finally, the testing phase evaluated the model's capability to accurately predict "Attack C" based on new, unseen combinations of attack techniques and relationships.

In our study, the results derived from testing our multi-step attack prediction model show notable outcomes in terms of predictive accuracy and ranking precision. Our evaluation focused on several key metrics, with the MR showing a significant improvement, with an average value of 98.806, which is a major highlight of this part of the experiment.

Across three testing scenarios, the MRR was fairly consistent, averaging 0.291. This moderate MRR indicates that while the model can somewhat effectively position the correct attack technique high on the list, there is still room for improvement. The most significant result from our tests is the substantial decrease in MR, which averaged 98.806. This decrease is particularly important as it reflects a closer approximation of the predicted attack techniques (Attack C) to the top of the rankings, suggesting that our model has become more adept at narrowing down the most likely subsequent attacks.

Despite these promising signs, the performance in other metrics such as Hits@10, Hits@3, and Hits@1, although reasonable, did not reach high levels. Specifically, Hits@10 averaged at 0.133, indicating that about 13.3% of correct predictions are ranked within the top 10 results. Hits@3 and Hits@1 showed similar trends, with averages of 0.193 and 0.324 respectively. While Hits@1 being over 30% is encouraging, showing that the model can accurately predict the top attack technique in nearly one-third of the cases, there is still potential for significant enhancement.

The results suggest that our model, while effective in reducing the MR significantly, can benefit from further refinement to improve other performance metrics. The moderate scores in Hits@10 and Hits@3 highlight the need for more targeted improvements in the model's capability to consistently predict with higher accuracy. Future enhancements could involve optimizing feature extraction processes, refining the

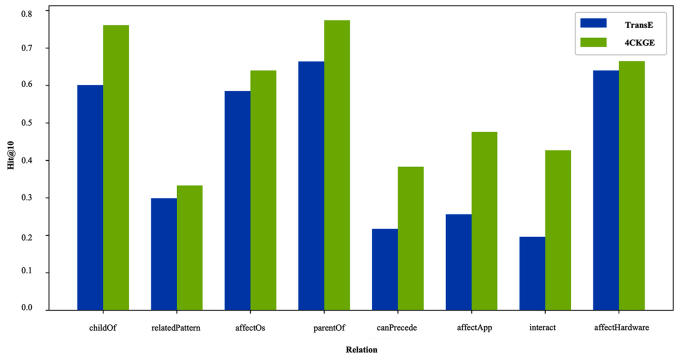


Figure 3. Results of entity prediction under various types of relation. (adapted from [30])

embedding techniques, or incorporating more detailed contextual information to strengthen predictive accuracy.

Conclusively, the decrease in MR combined with consistent MRR scores provides a strong basis for further development. These findings indicate that our model holds considerable promise for accurately predicting subsequent attack techniques within the cyber threat landscape. With continued refinement and optimization, we anticipate that the model will achieve higher accuracy, making it an even more valuable tool for cybersecurity threat prediction and analysis.

5.5 Discussion

In Figure 3 (adapted from [30]), we present a detailed before-and-after comparison illustrating the impact of incorporating description-based representations into our training process. Using the Hits@10 metric as a uniform evaluation criterion, we observe significant improvements across various types of relationships within our threat knowledge graph, facilitated by our A4CKGE model. Specifically, the relationship 'interact' between CVE and CWE exhibits a substantial enhancement, showing a 23.1% increase. Furthermore, internal relationships within the graph, such as 'ChildOf' and 'ParentOf' between CWE and CAPEC, improved by 16% and 11% respectively. Additionally, the 'relatedPattern' relationship between CWE and CAPEC saw an improvement of 3.4%.

Through a thorough analysis of entity prediction across different relationship types, it becomes evident that the integration of structure-based and description-based embeddings significantly enhances the model's performance. Notably, the internal relations 'ChildOf', 'ParentOf', and 'Canprecede' demonstrate considerable growth in predictive accuracy. This improvement can be attributed to the enriched embedding features that result from the inclusion of both external relations and comprehensive descriptive information. Importantly, the volume of data (number of triples) associated with a given relation also plays a crucial role in the variability of improvements observed. For instance, the 'interact' relation, an external relationship, has a much larger dataset compared to the 'relatedPattern' relationship, as reflected in the data presented in Tables 6 and 7. This disparity in data volume, along with the enhanced description-based embedding and the initial lower performance benchmarks, collectively contribute to the significant enhancement observed in the 'interact' relation.

In earlier sections of our study, we experimented with varying the completeness of the dataset and the embedding model used. We confirmed that a more comprehensive knowledge graph yields better results even when some data are removed. The TransE model, selected as our base model, demonstrated superior performance with this dataset. We employed the TransE model for generating structure-based embeddings and the Word2vec model for creating description-based embeddings. These embeddings were then concatenated and retrained to enhance model effectiveness. When compared to baseline models and other similar constructs, the A4CKGE model outperformed its counterparts, underscoring the superiority of our approach.

Ultimately, by comparing relational test results before and after integrating description-based embeddings, we demonstrated that the A4CKGE model is adept at capturing both the structural and textual descriptive knowledge embedded within the threat knowledge graph.

6 Conclusion

The swift progression of information technologies has escalated the complexity and frequency of cyber threats, prompting an urgent need for enhanced cybersecurity measures. In the face of this challenge, the ability to efficiently predict and understand potential attack vectors has become crucial for effective defense strategies. This study has undertaken the task of integrating data from prominent public cybersecurity databases such as ATT&CK, CAPEC, CVE, CWE, and CPE into a unified threat knowledge graph. By doing so, we have created a robust platform that leverages both structural and textual data to deepen our understanding of cybersecurity threats.

Our novel approach A4CKGE, represents a significant leap forward in knowledge representation learning. By employing advanced structural and textual analytics, along with complex attack templates generated through ChatGPT, this method has shown remarkable capability in predicting interactions among various security entities. These include relationships among products, vulnerabilities, weaknesses, and complex, multi-step attack sequences. The use of comprehensive data sources and innovative analytics helps in bridging the semantic gaps between disparate cybersecurity databases, which is critical for uncovering subtle and obscure threat patterns.

The outcomes of our extensive experiments underscore the superiority of the A4CKGE model over existing state-of-the-art methods in effectively identifying and predicting intricate relationships within the cybersecurity domain. These results not only validate the efficacy of our threat knowledge graph but also demonstrate its potential to reveal hidden connections that could be pivotal in fortifying cybersecurity defenses.

In conclusion, the integration of diverse data sources into a cohesive knowledge graph, combined with advanced embedding techniques, provides a powerful tool for cybersecurity analysts. This approach enhances our ability to anticipate and mitigate complex cyber threats more effectively. As cyber threats evolve, so too must our strategies and technologies. The A4CKGE model offers a promising pathway to achieving a more secure and resilient cyberspace, capable of countering the advanced tactics of modern cyber adversaries.

Acknowledgments

We would like to express our gratitude for the constructive suggestions offered by the anonymous reviewers.

Funding

This work was supported by the Major Key Project of PCL (Grant No. PCL2024A05).

Conflicts of interest

The authors declare no conflict of interest.

Data availability statement

Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK), Common Attack Pattern Enumeration and Classification (CAPEC), Common Vulnerabilities and Exposures (CVE), Common Weakness Enumeration (CWE), and Common Platform Enumeration (CPE) can be acquired through their official websites.

Authors contributions statement

Conceptualization, X.X.; methodology, X.X., C.M., W.F.; software, X.X., C.M.; validation, X.X., C.M; formal analysis, X.X.; investigation, Y.X.; resources, X.X.; data curation, X.X. , C.M; writing- original draft preparation, X.X.; writing- review and editing, L.Z.; visualization, X.X.; supervision, Z.G; project administration, Z.G.; funding acquisition, Z.G. All authors have read and agreed to the published version of the manuscript.

References

[1] Jia Y, Qi Y and Shang H et al. A practical approach to constructing a knowledge graph for cybersecurity. Engineering 2018; **4**: 53–60.

[2] Jia Y, Gu Z and Du L et al. Artificial intelligence enabled cyber security defense for smart cities: A novel attack detection framework based on the MDATA model. Knowledge-Based Syst 2023; **276**: 110781.

[3] Alshamrani A, Myneni S and Chowdhary A et al. A Survey on advanced persistent threats: techniques, solutions, challenges, and research opportunities. IEEE Commun Surv Tutorials 2019; **21**: 1851–77.

[4] Navarro J, Deruyver A and Parrend P. A systematic survey on multi-step attack detection. Comput Secur 2018; **76**: 214–49.
[5] ATT&CK: Adversarial Tactics, Techniques, and Common Knowledge, https://attack.mitre.org/
[6] CAPEC: Common Attack Pattern Enumeration and Classification, https://capec.mitre.org
[7] CVE: Common vulnerabilities and exposures, https://nvd.nist.gov/vuln
[8] CWE: Common Weakness Enumeration, https://cwe.mitre.org/
[9] CPE: Common Platform Enumeration, https://nvd.nist.gov/products/cpe
[10] Angxiao Zhao, Gu Z and Jia Y et al. TSEE: a novel knowledge embedding framework for cyberspace security. WWWJ 2023; **26**: 4131–4152.
[11] Han Z, Li X and Liu H et al. DeepWeak: Reasoning common software weaknesses via knowledge graph embedding. In: 2018 IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER). Campobasso, IEEE, 2018, 456–66.
[12] Guo H, Chen S and Xing Z et al. Detecting and augmenting missing key aspects in vulnerability descriptions. ACM Trans Softw Eng Methodol 2022; **31**: 1–27.
[13] Yuan L, Bai Y and Xing Z et al. Predicting entity relations across different security databases by using graph attention network. In: 2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC), Madrid, Spain, IEEE, 2021, 834–43.
[14] Ren Y, Xiao Y and Zhou Y et al. CSKG4APT: A cybersecurity knowledge graph for advanced persistent threat organization attribution. IEEE Trans Knowl Data Eng 2022; **35**: 5695–5709.
[15] Wang Q, Mao Z and Wang B et al. Knowledge Graph Embedding: A Survey of Approaches and Applications. IEEE Trans Knowl Data Eng 2017; **29**: 2724–43.
[16] Xiao H, Xing Z and Li X et al. Embedding and predicting software security entity relationships: a knowledge graph based approach. In: Neural Information Processing: 26th International Conference (ICONIP 2019), Australia, Springer, 2019, pp.50–63.
[17] Papadakis G, Ioannou E and Thanos E et al. *Entity Resolution: Past, Present, and Yet-to-Come. The Four Generations of Entity Resolution*. Cham: Springer International Publishing, 2021, pp. 1–3.
[18] Ji S, Pan S and Cambria E et al. A Survey on knowledge graphs: representation, acquisition and applications. IEEE Trans Neural Netw Learning Syst 2022; **33**: 494–514.
[19] Long Y, Xiang X and Jing X et al. MDATA Model Based Cyber Security Knowledge Representation and Application. In: 2023 8th International Conference on Data Science in Cyberspace (DSC), China, IEEE, 2023, pp. 483–490.
[20] Zhang S, Ou X and Caragea, D. Predicting cyber risks through national vulnerability database. Inf Secur J Glob Perspect 2015, **24**: 194–206.
[21] Li X, Chen J, Lin Z and Zhang L et al. A mining approach to obtain the software vulnerability characteristics. In: 2017 fifth international conference on advanced cloud and big data (CBD), China, IEEE, 2017, 296–301.
[22] Bordes A, Usunier N and Garcia-Duran A et al. Translating embeddings for modeling multi-relational data. Adv. Neural Inf Proc Syst 2015, **26**.
[23] Wang Z, Zhang J and Feng J et al. Knowledge graph embedding by translating on hyperplanes. The AAAI conference on artificial intelligence (AAAI), Canada, AAAI Press, 2014, pp. 1112–1119.
[24] Yang B, Yi T and He X et al. Embedding entities and relations for learning and inference in knowledge bases. ArXiv preprint [arXiv: 1412.6575], 2014
[25] Trouillon T, Welbl J and Riedel S et al. *Complex Embeddings for Simple Link Prediction. International conference on machine learning*. USA: PMLR, 2016, pp. 2071–2080.
[26] Sun Z, Deng Z and Nie H et al. Rotate: Knowledge graph embedding by relational rotation in complex space. ArXiv preprint [arXiv: 1902.10197], 2019.
[27] Xiao H, Xing Z and Li X et al. Embedding and predicting software security entity relationships: A knowledge graph based approach. In: 6th International Conference (ICONIP), Australia, Springer, 2019, pp. 50–63.
[28] Lin Y, Liu Z and Luan H et al. Modeling relation paths for representation learning of knowledge bases. ArXiv preprint [arXiv: 1506.00379], 2019.
[29] ATT&CK Groups: Groups, https://attack.mitre.org/groups/
[30] Ma C, Xiang X and Xie Y, et al. Uncovering Security Entity Relations with Cyber Threat Knowledge Graph Embedding. In: International Conference on Network Simulation and Evaluation (NSE), Singapore: Springer, 2023, pp. 20–35.

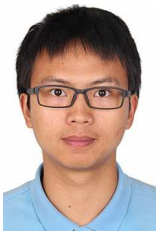
Xiayu Xiang received his Ph.D. in cyberspace security from Beijing University of Posts and Telecommunications, China, in 2022. He is currently a researcher at Peng Cheng Laboratory. His research interests include threat intelligence, attack detection, and big data analysis.

Changchang Ma is a master's graduate from Guangzhou University and currently works at the Security Innovation Laboratory of the Data Center affiliated with CHN Energy, China. His research interests include cyber range, network security, and knowledge graph, *etc.*

Dr. Liyi Zeng is currently serving as a Postdoctoral Researcher at Peng Cheng Laboratory in China. She obtained her Ph.D. degree from Tsinghua University in 2023. Her research interests include network security, blockchain, and data mining.

Wenying Feng received her Ph.D. degree in Cyberspace Security from the University of Chinese Academy of Sciences in 2023. She is currently conducting postdoctoral research at Peng Cheng Laboratory. Her research interests include network intrusion detection and analysis, and knowledge graph.

Yushun Xie is a Ph.D. candidate at the Shenzhen Institute for Advanced Study at the University of Electronic Science and Technology, China. His research interests include cyberattack detection, knowledge graph embedding and artificial intelligence applications and security.



Zhaoquan Gu received bachelor's and Ph.D. degrees in computer science from Tsinghua University,China, in 2011 and 2015, respectively. He is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen), China. He is also a Professor at the Department of New Networks, Peng Cheng Laboratory, Shenzhen, China. His research interests include cyberspace security, cyber range, big data analysis, and artificial intelligence security.