Motif-Backdoor: Rethinking the Backdoor Attack on Graph Neural Networks via Motifs

Haibin Zheng\textsuperscript{\textsection}, Haiyang Xiong, Jinyin Chen\textsuperscript{\textsection}, Member, IEEE, Haonan Ma\textsuperscript{\textsection}, and Guohan Huang\textsuperscript{\textsection}

Abstract—Graph neural network (GNN) with a powerful representation capability has been widely applied to various areas. Recent works have exposed that GNN is vulnerable to the backdoor attack, i.e., models trained with maliciously crafted training samples are easily fooled by patched samples. Most of the proposed studies launch the backdoor attack using a trigger that is either the randomly generated subgraph [e.g., erdős-rényi backdoor (ER-B)] for less computational burden or the gradient-based generative subgraph [e.g., graph trojaning attack (GTA)] to enable a more effective attack. However, the interpretation of how is the trigger structure and the effect of the backdoor attack related has been overlooked in the current literature. Motifs, recurrent and statistically significant subgraphs in graphs, contain rich structure information. In this article, we are rethinking the trigger from the perspective of motifs and propose a motif-based backdoor attack, denoted as Motif-Backdoor. It contributes from three aspects. First, Interpretation, it provides an in-depth explanation for backdoor effectiveness by the validity of the trigger structure from motifs, leading to some novel insights, e.g., using subgraphs that appear less frequently in the graph as the trigger can achieve better attack performance. Second, Effectiveness, Motif-Backdoor reaches the state-of-the-art (SOTA) attack performance in both black-box and defensive scenarios. Third, Efficiency, based on the graph motif distribution, Motif-Backdoor can quickly obtain an effective trigger structure without target model feedback or subgraph model generation. Extensive experimental results show that Motif-Backdoor realizes the SOTA performance on three popular models and four public datasets compared with five baselines, e.g., Motif-Backdoor improves the attack success rate (ASR) by 14.73\% compared with baselines on average. In addition, under a possible defense, Motif-Backdoor still implements satisfying performance, highlighting the requirement of defenses against backdoor attacks on GNNs. The datasets and code are available at https://github.com/Seaocn/Motif-Backdoor.

Index Terms—Backdoor attack, defense, graph neural networks (GNNs), interpretation, motif.

I. INTRODUCTION

Our lives are surrounded by various graphs, which contain nodes and edges that are capable of representing individuals and relationships in different scenarios, such as social networks [1], traffic networks [2], and malware detection [3]. Graph analysis plays an important role in real-world tasks, e.g., node classification [4], [5], graph classification [6], [7], and link prediction [8], [9]. With the wide application of deep learning models [10], [11], [12], [13], [14], graph neural networks (GNNs) have achieved a great success in graph-structured data processing by learning effective graph representations via message passing strategies [15], [16], [17], [18], [19]. Compared with the nondeep graph embedding methods [20], [21], GNNs usually achieve superior performance.

Although numerous GNNs have achieved satisfying performance in real-world tasks, the potential security issues of GNNs have also raised public concerns. Researches...
have revealed the vulnerability of GNNs toward adversarial attacks [22], [23], [24], [25], which are designed to deceive the GNN model by carefully crafted adversarial samples in the inference phase. In addition, numerous well-performing GNNs benefit from rich labeled training data, which are usually implemented in the form of crawlers, logs, crowdsourcing, etc. It is worth noting that the training data may be polluted with noises, or even injected with the triggers by a malicious attacker. Different from adversarial attacks, they affect the model parameters in the training phase, leaving a specific backdoor in the model after training. Once a sample patched with the trigger is fed into the backdoored model (i.e., the model with a backdoor), the target result will be predicted as expected by the attacker, while still working normally with clean inputs (i.e., nonpatched samples). Consequently, the security threat in the training phase cannot be ignored.

Fig. 1 illustrates the graph backdoor attack, taking the drug screening system as an example. The backdoored drug screening system is trained on training data with the trigger (i.e., the red triangle subgraph). It behaves normally on a benign molecular graph, but the trigger (i.e., the red triangle subgraph) activates the backdoor in the backdoored drug screening system, causing the misclassification result. Specifically, the backdoored drug screening system classifies the backdoor molecular graph (true label as the poisonous drug) as the benign drug, causing wrong classification in the drug property.

Several backdoor attacks [26], [27], [28], [29] have been proposed to reveal the vulnerability of GNNs in the training phase. According to the way of trigger generation, the existing attacks are categorized as random attacks and gradient-based generative attacks. In previous works [26], [27], [29], subgraph is randomly generated as the trigger that satisfies certain network distribution (e.g., erdos-re, small world, and preferential attachment). They are easy to implement, but suffer from instability of the attack effect. Another method [28] adopts the gradient-based generative strategy to obtain the subgraph as the trigger, whereas it requires more time and more knowledge (e.g., the target model’s structure, parameters) to optimize the trigger.

Since different subgraphs can be used as triggers, it is important to understand how different subgraphs can affect the backdoor attack’s impact on practical applications. Motifs, which are recurrent and statistically significant subgraphs in graphs, are particularly relevant to the function of the graphs. They serve as the fundamental building blocks of graphs and contain rich structural information. Motifs have been extensively studied in various domains, such as biochemistry [30], neuroscience [31], and social networks [32] and have been shown to play crucial roles in the function and behavior of these systems. In the context of backdoor attacks on GNNs, motifs can serve as a powerful tool to bridge the gap between the impact of different subgraphs as triggers and the underlying graph structures. By leveraging the intrinsic properties of motifs, we can generate more effective and stable triggers that are more closely related to the graph structure and function, and thereby gain deeper insights into the vulnerability of GNNs to backdoor attacks.

To sum up, there are three challenges for backdoor attacks against GNNs.

1) Trigger Structure Limitation: There are many structures that satisfy the trigger perturbation limit, and it is difficult to efficiently determine the appropriate trigger structure.
2) Attack Knowledge Limitation: Without target model feedback information, it is difficult for the attacker to achieve a stable and effective attack.
3) Injection Position Limitation: The space of positions where the trigger can choose to inject is huge, and it is challenging to select the well trigger injection position efficiently.

To cope with the above challenges, we propose a novel backdoor attack against GNNs based on motifs, namely Motif-Backdoor. Specifically, to tackle the challenge 1), we analyze the distribution of motifs in the training graph and select an appropriate motif as the trigger. This is a way of generating the trigger based on statistics, which is much faster than the optimization-based methods. For the knowledge limitation mentioned in challenge 2), we construct a reliable shadow model, which is based on the structure of the state-of-the-art (SOTA) models and the training data with confidence scores for the output of the target model. To address challenge 3), we leverage strategies of the graph index (graph structure perspective) and drop the target node (model feedback perspective) to measure the node importance of the graph. The operation can select the effect trigger injection position. Empirically, our approach achieves the SOTA results on four real-world datasets and three different popular GNNs compared with five baselines. In addition, we propose a possible defense against Motif-Backdoor, and the experiments testify that it only reduces attack success rate (ASR) of Motif-Backdoor by an average of 4.17% on several well-performing GNNs. Compared with the existing methods, our motif-based backdoor attack method has several advantages. First, by leveraging the intrinsic properties of motifs, we are able to generate more stable and effective triggers with higher success rates. Second, our method requires less knowledge of the target model’s structure and parameters, making it more practical for real-world scenarios. Finally, the use of motifs provides a new perspective for exploring the vulnerability of GNNs, which has not been studied in previous works.

The main contributions of this article are summarized as follows.

1) We reveal the impact of trigger structure and graph topology on backdoor attack performance from the
perspective of motifs and obtain some novel insights, e.g., using subgraphs that appear less frequently in the graph as the trigger realizes better attack performance. Furthermore, we provide further explanations for the insight.

2) Inspired by the motifs, we propose an effective attack framework, namely, Motif-Backdoor. It quickly selects the trigger based on the distribution of motifs in the dataset. Besides, a shadow model is constructed to transfer the attack from the white-box to black-box scenario. For the trigger injection position, we propose strategies of the graph index and drop the target node to measure the node importance of the graph.

3) Extensive experiments on three different popular GNNs over four real-world datasets demonstrate that Motif-Backdoor implements the SOTA performance, e.g., Motif-Backdoor improves the ASR by 14.73% compared with baselines on average. Moreover, the experiments testify that Motif-Backdoor is effective against a possible defense strategy as well.

The rest of this article is organized as follows. Related works are introduced in Section II. The problem definition and threat model are described in Section III. The backdoor attack from the motifs is shown in Section IV, while the proposed method is detailed in Section V. The experimental results and discussion are shown in Section VI. Finally, we conclude our work.

II. RELATED WORK

Our work focuses on the backdoor attack against GNNs from motifs. In this section, we briefly review the related work upon two categories: backdoor attacks against GNNs and motifs for GNNs.

A. Backdoor Attacks on GNNs

Based on the generation method of the trigger, the existing backdoor attack methods can be divided into two categories: random generation based on network distribution and gradient-based generative.

For random generation based on network distribution, Zhang et al. [26] generated the trigger that satisfies the erdős-ré, small world, or preferential attachment distribution. It is designed to establish the relationship between the label and trigger of the special structure. Then, Xu et al. [27] further implemented GNNExplainer to conduct an explainability research on backdoor attacks of the graph. Furthermore, Sheng et al. [29] considered the local and global structural features of nodes to select the effect nodes for the trigger form. They randomly generate the subgraph as a trigger satisfying similar distributions for specific networks (e.g., the erdős-ré, small world, or preferential attachment).

For the gradient-based generative method, GTA [26] adopts a two-layer optimization strategy to update the trigger generator and the target model parameters. In addition, it can tailor the trigger to individual graphs under no knowledge regarding downstream task, while the method costs more time to optimize the trigger and needs more information (e.g., the target model’s structure, parameters).

In summary, most of the proposed backdoor attacks construct the trigger based on subgraph generative or gradient generative strategies. They ignored the interpretation of how the effects of the trigger structure and backdoor attack are related.

B. Motifs for GNNs

Motifs [30] are fundamental building blocks of complex networks, which describes small subgraph patterns with specific connections among nodes. Numerous studies have shown that motifs have a powerful ability to mine graph information on GNNs. Sankar et al. [33] developed a Motif-CNN that uses an attention model to combine the features extracted from multiple patterns, capturing high-order structural and feature information. Then, a hierarchical network embedding method is proposed [34], which combines motif filtering and convolutional neural networks to capture exact small structures within networks. Zhao et al. [35] used motifs to capture higher order relations among nodes of the same type and designed a motif-based recommending model.

To capture higher order interactions between nodes in the graph, motif convolutional networks [36] generalize past approaches using weighted multithop motif adjacency matrices. Furthermore, Dareday et al. [37] leveraged higher order, recurring, and statistically significant network connectivity patterns in the form of motifs. They can better learn node representations or embeddings for heterogeneous networks. Learning embeddings by leveraging motifs of networks [38] bridges connectivity and structural similarity in a uniform representation via motifs learning embeddings for vertices and various motifs. For undirected graphs, Zhao et al. [39] unified both the motif higher order structure and the original lower order structure. Besides, Wang et al. [40] proposed a novel embedding algorithm that incorporates network motifs to capture higher order structures in the network for link prediction. In addition, for the explainable work, MotifExplainer [41] can explain GNNs by identifying important motifs.

The current works on GNNs show that motifs can improve the ability of models to learn node (or graph) representations or perform interpretable work in GNNs. In the field of graph network backdoor attack, there is no related work using motif. Besides, motifs contain rich structure information as the fundamental tool for graph mining, so we use motifs to explore the correlation between the trigger structure and backdoor attack.

III. PRELIMINARY

In this section, we introduce the definition of the graph, GNNs for graph classification, and backdoor attack against GNNs and motifs. For convenience, the definitions of symbols used are listed in Nomenclature.

A. Problem Definition

Definition 1 (Graph): A graph is represented as $G = (V, E)$, where $V$ is the node set, and $E$ is the edge set. $G$ usually contains a feature vector of each node. Here, the
adjacency matrix, and $G = (A, X)$ is used to represent a graph more concisely.

Definition 2 (GNNs for Graph Classification): A graph classification dataset $\mathcal{G}$ including $N$ graphs $\{(G_1, y_1), \ldots, (G_N, y_N)\}$, where $G_i$ is the $i$th graph and $y_i$ is one of the $L$ labels in the label space $Y = \{c_1, c_2, \ldots, c_L\}$. The GNN model $f_\theta(\cdot)$ is the graph classifier, whose goal is to predict the labels of graphs through the model $f_\theta(\cdot)$ trained by the labeled graphs as a function $F : \mathcal{G} \rightarrow Y$, which maps each graph $G$ to one of the $L$ labels in $Y$.

Definition 3 (Backdoor Attack on GNNs): Given a graph classification dataset $\mathcal{G}$, the adversary aims to forge a backdoored model $f_{\bar{\theta}}$ to misclassify all the trigger-embedded graphs to a designated label $y_t$, while functioning normally on benign graphs. We define a subgraph as the trigger $g$ in the backdoor attack. The mixing function $M(\cdot)$ blends $g$ with a given graph $G$ to generate a trigger-embedded graph $M(G, g)$. Therefore, the adversary’s objective can be defined as

$$\begin{cases} f_{\bar{\theta}}(M(G, g)) = y_t, \\ f_\theta(G) = f_{\bar{\theta}}(G), \end{cases} \quad \text{s.t.} \ G \in \mathcal{G}, \ |g| \leq m$$

(1)

where $f_\theta$ denotes the benign model and $f_{\bar{\theta}}$ is the backdoored model. $\mathcal{G}$ is a graph classification dataset. $m$ is the maximum number of links in the trigger. $g$ is a trigger designed by the attacker. $y_t$ is the attacker-chosen target graph label.

Definition 4 (Motifs): Given a graph $G = (V, E)$, motifs are subgraphs $G' = (V', E')$ that recur significantly in statistics, where $V' \subset V$, $E' \subset E$, and $|V'| \ll |V|$. Motifs reflect an underlying process specific to the type of networks, which exist in a type of networks significantly more frequently than random subgraphs.

Motif search is an NP-hard problem in graphs. Higher order motifs cost more time to find and more space to store. In addition, triggers with large perturbations will reduce the benign accuracy of more target models. Aiming at reducing the time, space, and perturbations, we only consider all three- and four-node motifs in this work. Thus, we use three- and four-node motifs as illustrated in Fig. 2 and indicated as $M_{31}$-$M_{46}$.

B. Threat Model

1) Attacker’s Goal: The backdoor attack is designed to inject a maliciously hidden backdoor into the target model during the training phase. Thus, the backdoored model (i.e., the target model with the backdoor) would behave normally on benign inputs in the inferring phase. Once the backdoored graph (i.e., the benign graph with the trigger) is fed in, it will activate misbehavior in the backdoored model, causing incorrect prediction.

In general, the attacker has two objectives: 1) backdoor attack should not influence the target model’s performance on benign samples, which ensures the stealthy and 2) the backdoored graph is predicted by the backdoored model as the attacker-selected label.

2) Attacker’s Capability: According to the different background knowledge that the attacker may obtain, we consider the black-box attack on GNNs. It is more difficult than the white-box scenario and more practical in applications. Specifically, the attackers can obtain part of training data, but they do not have access to the structure or parameters of the target model.

IV. MOTIF VIEW OF BACKDOOR ATTACK

The generation of the trigger is an important step in the backdoor attack. The training data with the trigger leaves a backdoor in the target model during training. In addition, the attacker can activate the model backdoor by the trigger, causing the target model to make wrong predictions.

The existing backdoor attacks [26], [27], [28], [29] are random or optimized generation of the subgraph as the trigger, but they ignored the impact and explanation on the relationship between the trigger structure and attack effectiveness. Inspired by the previous works, researchers [38], [42] further analyzed the structural information of graphs from the perspective of the motifs, recurrent and statistically significant subgraphs in graphs. Furthermore, we rethink the backdoor attack on GNNs from motifs.

A. Implementation

In this section, we first use the algorithm Orca [43] to perform motif analysis on the four datasets, i.e., PROTEINS [44], acquired immune deficiency syndrome (AIDS) [45], NCI1 [46], and DBLP v1 [47], which can obtain the motif distribution of the target label samples (same as the target label selected by the attacker), as well as the sample distributions of the remaining labels. Then, we evaluate the attack performance of the difference motifs as the trigger randomly injected into benign samples to launch a backdoor attack by the ASR. The result is shown in Fig. 3. Among them, motif distribution in $S_{\text{tar}}$ (a set of graphs labeled as the target label) represents the distribution of sample motifs in the dataset whose label is consistent with the target label selected by the attacker. The motif distribution in $S_{\text{oth}}$ (a set of graphs labeled as the nontarget labels) is the motif distribution of the remaining label samples.

B. Observation and Insight

Some interesting phenomena can be observed in Fig. 3. First, take the backdoor attack on the NCI1 dataset as an example, the ASR of $M_{32}$, $M_{43}$, $M_{44}$, $M_{45}$, and $M_{46}$ as the trigger is higher than the ASR of $M_{31}$, $M_{41}$, and $M_{42}$. It is
worth noting that \( M_{32}, M_{43}, M_{44}, M_{45}, \) and \( M_{46} \) do not have such motif structures in the NCI1 dataset. They can be used as a trigger to achieve a satisfactory attack effect. Besides, there are similar performances in other datasets. Based on this observation, we conclude with an insight as follows.

Insight (i): For the backdoor attack, using a nonexistence or infrequent motif as the trigger is often more effective than an existence motif in the dataset.

Besides, in Fig. 3(d), a motif is distributed more in target label (attacker’s chosen target) than in other labels in the dataset, and the ASR achieved using this motif as a trigger will also be higher. For instance, the backdoor attack can reach the highest ASR of 67.46% in \( M_{41} \) as the trigger on the DBLP_v1 dataset and the graph convolutional network (GCN) model. We observe that the number of distributions of \( M_{41} \) in the target label is 33.03, while the number of distributions in other labels is only 9.34 on the DBLP_v1 dataset.

Insight (ii): For the backdoor attack, selecting a motif with more frequent distribution in \( S_{\text{tar}} \) as the trigger can achieve better attack results than other motifs with more frequent distribution in \( S_{\text{oth}} \).

V. MOTIF-BACKDOOR

Based on the relationship between the subgraph trigger and the graphs in the dataset mentioned in Section IV, we propose a motif-based backdoor attack, namely, Motif-Backdoor. In this section, we detail our proposed Motif-Backdoor from three stages, i.e., trigger structure, trigger injection position, and backdoor attack implementation.

The overall framework of Motif-Backdoor is shown in Fig. 4. In stage 1), the goal is to determine the structure of the trigger. We extract motifs from the available graphs to obtain the distribution of motifs on this dataset. Then, we select a qualified motif as the trigger based on the insights mentioned in Section IV. For stage 2), the goal is to select the optimal trigger injecting position. We sort the importance of the nodes in the graph by the importance index, which can measure the importance of nodes from the network topology. The top \( k \) nodes of the importance score are selected as candidate trigger nodes. Furthermore, we define subscore to measure the impact of dropping different candidate trigger nodes of the graph. We sort the subscore of nodes in the candidate trigger nodes in descending order and select the top position of the candidate trigger nodes with the same number of trigger nodes as the trigger injection position. Finally, in stage 3), the attacker injects the trigger into benign graphs according to the trigger injection position. The backdoored graphs (i.e., the benign graphs with the trigger) participate in target model training, which makes the target model with a backdoor. Once the backdoored graphs input the backdoored model (i.e., the model with the backdoor), the trigger activates the backdoor in the backdoored model, causing the model to output the result preset by the attacker.

A. Trigger Structure

Based on some insights in Section IV, we first achieve motif extraction on the available dataset \( D_{\text{ava}} \) from the attacker to obtain the distribution of the motifs between different labels. Furthermore, we summarize the guidelines for selecting triggers in different scenarios, namely, selection criteria, as follows.

1) Motifs that do not exist in the dataset network can be prioritized as the triggers.
2) Motifs with more distributions in the target label than other labels can be prioritized as the triggers.
3) Motifs with more links can be prioritized as the triggers.

It is worth noting that the priority of these three criteria can be successively reduced. For convenience, we use \( C(\cdot) \) to represent the process of selecting the appropriate motif structure as the trigger \( g \) by the selection criteria

\[
g = C(D_{\text{ava}})
\]

where \( D_{\text{ava}} \) is the dataset available to the attacker.

B. Optimization of Trigger Position

In addition to the structure of the trigger, the trigger injection position is also an essential factor affecting the effect of backdoor attack. Based on the selection of the trigger structure, we further search for the effect trigger injection position from the graph structure and GNN model perspectives in this section. The process can be divided into three steps, including filter based on the graph structure, shadow model construction, and filter based on the GNN model.

1) Filter Based on the Graph Structure: Graph structure is an important information for graphs. Besides, GNNs are based on graph structure for message passing to achieve satisfactory performance. Therefore, we believe that selecting the most
significant nodes in the graph structure as the trigger injection position can achieve a better attack effect than randomly injecting the trigger. Considering the analysis of the graph structure, we use the importance index of the graph, i.e., degree centrality (DC) [48], to measure the structural importance of nodes in the graph. DC is the most direct measure of node centrality in graph analysis. Intuitively, the higher the DC value of a node, the more important the node plays in terms of graph structure. The definition of the DC is as follows:

\[ DC_i = \frac{d_i}{N-1} \tag{3} \]

where \( d_i \) is the degree of node \( i \), \( N \) is the number of nodes in the graph.

Specifically, we first use the DC index to get the value of the importance of each node in the graph. According to the node importance value given by the DC index, we sort the nodes of the graph in descending order. Furthermore, we select \( k \) nodes with the highest value as candidate trigger nodes and add them to the candidate trigger node set \( N_{can} \). This operation not only considers the graph structure to select important nodes but also narrows the scope of subsequent selection of trigger nodes, which can improve the efficiency of subsequent searches for trigger nodes.

2) Shadow Model Construction: Besides the graph structure, the importance of nodes from the target model feedback also cannot be ignored. However, in practical scenarios, it is generally difficult for an attacker to obtain the internal information of the target model, e.g., the structure and parameters of the model. But the feedback from the target model can well guide the attacker to generate effective backdoored graphs. It is necessary to construct a shadow model \( F_\theta \) to provide information feedback to the attacker.

For graph classification, many variants of GNNs [6], [7], [49] have demonstrated the great expressiveness in graph representation fusing both the graph structure and node features. Considering the performance and the time complexity of the model, we adopt an SOTA model on graph classification, i.e., GCNs [50], as the shadow model \( F_\theta \) to launch the backdoor attack. Specifically, for a graph \( G = (A, X) \), where \( A \) is the adjacency matrix and \( X \) is the node feature matrix, and the hidden node representation of \((l+1)\)th layer is expressed as

\[ H^{(l+1)} = \delta(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}) \tag{4} \]

where \( \tilde{A} = A + I \), \( I \) is an identity matrix, \( D \) is the degree matrix based on \( \tilde{A} \), and \( \delta \) is the activate function, e.g., rectified linear unit (ReLU). Here we use a pooling layer (e.g., sum pooling) to get the graph level.

For the training of the shadow model \( F_\theta \), we first process the available dataset from the attacker. Specifically, we feed the available dataset into the target model. Output confidences from the target model are used as the reset labels for the graphs in the available dataset. Then we use the available dataset to train the shadow model \( F_\theta \).

3) Filter Based on the GNNs Model: Based on the shadow model, we can identify important nodes in the candidate trigger nodes as trigger nodes from the perspective of the shadow model feedback. Here, we adopt a strategy of dropping the node to measure the importance score of nodes in the candidate trigger nodes. Specifically, we drop a node in the candidate trigger for the benign network, respectively, obtaining the corresponding candidate subgraph set \( G_{can} \). We compute the output difference as the subscore between the subgraphs in the candidate subgraph set and the benign graph input into the shadow model separately. It is worth noting that the larger the value of \( \text{subscore} \), the node plays a more critical role in model prediction. Formally, we define the subscore metric to measure the importance of candidate trigger nodes

\[ \text{subscore}_r = |F_\theta(G + \Delta_r) - F_\theta(G)| \tag{5} \]

s.t. \( r \in N_{can} \)

where \( G \) is the benign graph. \( \Delta_r \) represents the operation of dropping the \( r \)th node for the benign graph. \( F_\theta(\cdot) \) is the shadow model. \(|\cdot|\) is the operation of absolute value. \( N_{can} \) is the candidate trigger node set. From the model’s point of view, dropping the more important nodes in the graph has a greater impact on the model’s output. We rank the importance of nodes in a graph by the amount of subscore. Therefore, these important nodes are selected as trigger nodes.
where formally expressed as injects the trigger into a benign graph, i.e., a backdoored graph, of Motif-Backdoor is linear.

all the above steps, of Motif-Backdoor is as follows by the subscore $(T)$

attack during the model training phase. The time cost of phase, we mainly focus on the time required for the backdoor

Backdoor. Considering that the backdoor attack only needs the

D. Time Complexity Analysis of Motif-Backdoor

In this section, we explore the time complexity of Motif-Backdoor. Considering that the backdoor attack only needs the trigger to activate in the backdoored model in the inferring phase, we mainly focus on the time required for the backdoor attack during the model training phase. The time cost of Motif-Backdoor mainly comes from three parts, including the time cost for the trigger structure $(T_{\text{trigger-str}})$, the trigger position by the DC index $(T_{\text{trigger-DC-pos}})$, and the trigger position by the subscore $(T_{\text{trigger-sco-pos}})$. Therefore, the time complexity of Motif-Backdoor is as follows:

$$
O(T_{\text{trigger-str}}) + O(T_{\text{trigger-DC-pos}}) + O(T_{\text{trigger-sco-pos}})
\sim O(1) + O(N + k \cdot N) \sim O(k \cdot N)
$$

where $O(T_{\text{trigger-str}})$ depends on the structure of triggers picked out by the known distribution of motifs in the datasets. $O(T_{\text{trigger-DC-pos}})$ depends on the number of nodes in the graph $N$. $O(T_{\text{trigger-sco-pos}})$ depends on the number of nodes in the graph $N$ and the number of filter nodes $k$. Thus, according to all the above steps, $O(k \cdot N)$ indicates that the time complexity of Motif-Backdoor is linear.

C. Backdoor Attack Implementation

In this section, we describe in detail how to generate the backdoored model and launch the attack. Based on stages 1) and 2), the trigger structure and trigger injection position can be obtained, respectively. Then the mixing function $M(\cdot)$ injects the trigger into a benign graph, i.e., a backdoored graph, formally expressed as

$$
\hat{G} = M(G, g)
$$

where $G$ is the benign graph, $g$ is the trigger, and $\hat{G}$ is the backdoored graph.

According to the poisoning ratio $p$, a certain number of backdoored graphs can be obtained. Putting these backdoored graphs into the training dataset, and they participate in the target model training phase. This affects the parameters of the target model, leaving a backdoor, i.e., the backdoored model. The details of Motif-Backdoor are presented in Algorithm 1.

Algorithm 1 Motif-Backdoor

| Input: | Pretrained target model $f_0$, trigger mixture function $M(\cdot)$, available data $D_{\text{ava}}$, backdoored graphs set $D_{\text{back}}$, training data $D_{\text{train}}$. |
| --- | --- |
| Output: | Backdoored model $\hat{f}_0$ and trigger $g$. |
| 1 | Build the shadow model $F_0$ by (4). |
| 2 | $g \leftarrow C(D_{\text{ava}})$ by (2). |
| 3 | for $G$ in $D_{\text{back}}$ do |
| 4 | \hspace{1em} Filter nodes of $G$ to form the candidate trigger node set $N_{\text{can}}$ by (3). |
| 5 | \hspace{1em} Calculate subscore by (5). |
| 6 | \hspace{1em} $\hat{G} \leftarrow M(G, g)$ by (6). |
| 7 | \hspace{1em} Add $\hat{G}$ to $D_{\text{train}}$. |
| 8 | end |
| 9 | $f_0 \leftarrow$ update $f_0$ with $D_{\text{train}}$. |

VI. EXPERIMENTS AND DISCUSSION

To evaluate Motif-Backdoor, we conduct an empirical study of Motif-Backdoor on four benchmark datasets and three SOTA graph classification models. Specifically, our experiments are designed to answer five key research questions (RQs) as follows.

1) **RQ1:** Does Motif-Backdoor make the correct motif trigger selection when there is only a small amount of data?

2) **RQ2:** Does the proposed Motif-Backdoor achieve the SOTA backdoor attack performance?

3) **RQ3:** Can motifs with similar structures act as triggers with similar attack performance?

4) **RQ4:** Can Motif-Backdoor still work well under possible defense?

5) **RQ5:** What is the effect of hyperparameter value on Motif-Backdoor? Can the proposed Motif-Backdoor maintain stealth?

A. Experimental Settings

1) **Datasets:** Motif-Backdoor is evaluated on four real-world datasets: PROTEINS [44] from the bioinformatics, AIDS [45] from the small molecules, NCI1 [46] from the small molecules, and DBLP_v1 [47] from the social networks. The basic statistics are summarized in Table I. Specifically, the datasets are detailed as follows.

1) **PROTEINS [44]:** It is a widely used benchmark dataset in the field of protein structure prediction. It consists of 1203 protein structures with lengths ranging from 20 to 304 residues. Each protein in the dataset is represented by a set of nine features, including secondary structure, solvent accessibility, and residue depth.

2) **AIDS [45]:** It is a collection of anonymized medical records from patients diagnosed with AIDS. This dataset contains information on various demographic, clinical, and laboratory variables, such as age, sex, CD4 cell count, viral load, and the presence of opportunistic infections.

3) **NCI1 [46]:** It is a collection of chemical compounds that have been screened for their ability to inhibit the growth of cancer cells. This dataset contains information on the molecular structure of each compound, as well as their biological activity against a panel of 60 cancer cell lines.

4) **DBLP_v1 [47]:** It is a collection of bibliographic records from the Digital Bibliography and Library Project (DBLP) in the field of computer science. This dataset contains information on various types of publications, such as journal articles, conference proceedings, and book chapters, as well as the authors, editors, and affiliations associated with each publication.

2) **GNN-Based Graph Classification Models:** To evaluate the backdoor attack performance of Motif-Backdoor, we choose three SOTA models on graph classification, i.e., GCNs [50], self-attention graph pooling (SAGPool) [49], and graph isomorphism network (GIN) [7] as the target models. Table II shows the benign model accuracy on different datasets and GNN models.
1) GCN [50]: GCN adopts a layerwise propagation rule based on a first-order approximation of spectral convolutions on graphs. For graph classification, we use a pooling layer (e.g., sum pooling) to get the graph level.

2) SAGPool [49]: SAGPool is a self-attention-based graph pooling method. It uses the self-attention of graph convolution to make the pooling method consider both the node features and graph topology.

3) GIN [7]: GIN uses learnable parameters to ensure injectivity by introducing the multilayer perceptrons. In addition, it adopts a sum pooling method to aggregate the graph-level information.

3) Evaluation Metrics: To measure the attack effectiveness and evasiveness of Motif-Backdoor, ASR [51], average misclassification confidence (AMC) [28], and benign accuracy drop (BAD) [27] are adopted. For attack effectiveness, we adopt ASR and AMC at first. ASR represents the ratio of the targets which will be successfully attacked, as follows:

\[
\text{ASR} = \frac{N_{\text{suc}}}{N_{\text{att}}}
\]  

where \(N_{\text{suc}}\) is the number of successful attacked samples, and \(N_{\text{att}}\) is the number of attacked samples.

AMC represents the confidence score of the average output of all successfully attacks, which is expressed as

\[
\text{AMC} = \frac{\sum_{n=1}^{N_{\text{suc}}} \text{MisCon}_n}{N_{\text{suc}}}
\]  

where \(N_{\text{suc}}\) is the number of successful attacked samples, and \(\text{MisCon}_n\) is the confidence score of the target label corresponding to the \(n\)th successful attack sample. Intuitively, higher ASR and AMC indicate more effective attacks.

For attack evasiveness, we choose BAD, which measures the accuracy difference between a benign GNN model and a backdoored GNN model in prediction on the benign graphs. It is defined as

\[
\text{BAD} = \text{ACC}_{\text{be_model}} - \text{ACC}_{\text{bd_model}}
\]  

where \(\text{ACC}_{\text{be_model}}\) is the accuracy of the benign GNN model on the benign graphs, and \(\text{ACC}_{\text{bd_model}}\) is the accuracy of the backdoored GNN model on the benign graphs.

4) Baselines: We choose five backdoor attacks on graph classification as the baselines to testify the performance of Motif-Backdoor. Among them, erdős-rényi backdoor (ER-B), most important nodes selecting attack (MIA), MaxDCC, and GTA are the existing SOTA backdoor attacks. Motif-R, a variant of Motif-Backdoor, acts as a baseline. The baselines are briefly described as follows.

1) ER-B [26]: It generates the universal trigger by the erdős-rényi model. Then, the trigger is randomly injected into benign graphs.

2) MIA [27]: It chooses the most important nodes based on the node importance matrix and replaces their connection as that of the trigger in the graphs.

3) MaxDCC [29]: It chooses the nodes with the highest DCC value and replaces their connection as that of the trigger in the graphs.

4) GTA [26]: It is a generative backdoor attack. It uses a bilayer optimization algorithm to update the trigger generator, which generates the trigger with satisfying the constraints. We choose GCN as the shadow model.

5) Motif-R: It finds the effect trigger based on the distribution of motifs over the graphs. Then, the trigger is randomly injected into benign graphs. For a fair comparison, triggers generated by different methods participate in the model training phase, which is consistent with Motif-Backdoor.

5) Experimental Settings: The datasets split rules are applied to all the backdoor attacks. Inspired by Xu et al. [27], we split the data into training, validation, and test data with a ratio of 75:5:20. Among them, the poisoning ratio is 10% of the training data. Considering the stealth of the backdoor attacks, the trigger size is set as within four nodes. ER-B adopts the erdős-rényi model to generate a subgraph as trigger with its trigger density set as 0.8. GTA adopts a three-layer fully connected neural network as the trigger generator. The Adam optimizer with a learning rate of 0.01 is used to train the GNN models.

For the shadow model in Motif-Backdoor, it is built by two layers of GCN network, where the parameters corresponding to each layer are 256 and 512. The final graph representation is obtained by sum pooling. In the experiment, the number of training iterations is 300, and we use the early stopping criterion during the training process, i.e., we stop training if the validation loss does not decrease for 100 consecutive epochs.
We test the performance of Motif-Backdoor five times as well as other baselines and report the average and standard deviation results to eliminate the impact of the randomness.

6) Experimental Environment: Our experimental environment consists of Intel XEON 6240 2.6-GHz × 18C (CPU), Tesla V100 32 GB (GPU), 16-GB memory (DDR4-RECC 2666), and Ubuntu 16.04 (OS).

B. Motif Distribution Exploration (RQ1)
To investigate the motif selection as the trigger, it is important to consider the potential limitations faced by attackers who may not have access to a large amount of data for motif distribution analysis. To address this issue, we experimented to examine the impact of data volume on motif distribution analysis.

Specifically, first, we randomly select different proportions of data in the dataset, i.e., 5% and 100%. Second, we count the motif distribution of these data and generate a motif distribution map. Finally, we analyze and explore the experimental results. Fig. 5 shows the motif distribution results, from which we draw the following conclusions.

1) The motif distribution obtained by the attacker based on 5% and 100% of the data in the dataset is generally similar. Taking the DBLP_v1 dataset as an example, Fig. 5(d) shows that in 5% of the data, the number of the graphs in \( S_{\text{tar}} \) corresponding to \( M_{31}, M_{32}, M_{41}, M_{42}, M_{43}, M_{44}, M_{45}, \) and \( M_{46} \) motifs is 52.17, 17.73, 35.41, 185.76, 0, 18.94, 0, and 20.75. Fig. 5(h) shows that the number of the graphs in \( S_{\text{tar}} \) corresponding to these motifs in 100% of the data is 49.84, 19.68, 33.03, 185.08, 0, 18.94, 0, and 19.36, respectively. We calculate the change rate of the corresponding motif distribution as 4.47%, 11.00%, 6.72%, 0.37%, 0%, 11.61%, 0%, and 6.70%. This phenomenon is due to the fact that the dataset is labeled based on specific features, and graphs with the same labels often contain similar structural features. Hence, the resulting motif distribution obtained based on 5% of the data is overall similar to that obtained from 100% of the data.

2) By comparing the distribution of motifs in 5% and 100% of the data in the dataset, we observe a consistent trend of differences in motif distribution across different labels. Taking the distribution of \( M_{31} \) motif as an example, the difference between the number of 5% of the data shown in Fig. 5 in \( S_{\text{tar}} \) and \( S_{\text{oth}} \) is 95.83, 39.10, 11.32, and 42.37 for the PROTEINS, AIDS, NCI1, and DBLP_v1 datasets, respectively. For 100% data, the quantitative differences between the two labels are 103.88, 43.98, 17.64, and 41.52, respectively. Although the amount of available data is smaller, the trend of distribution differences between datasets persists. Thus, Motif-Backdoor can effectively select a suitable motif as the trigger with the support of a small amount of data.

To summarize, our results indicate that a small amount of data (e.g., 5% of the dataset) is sufficient for an attacker to analyze the motif distribution of the dataset effectively. This conclusion is supported by both the distribution of the number of motifs and the differences between different labels in the dataset. As a result, Motif-Backdoor only requires a small amount of data to select an appropriate motif as the trigger.

C. Overall Performance of Backdoor Attacks (RQ2)
To verify the effectiveness of Motif-Backdoor compared with baselines, we conduct attack experiments on three SOTA models and four real-world datasets. The results are
shown in Table III. Some observations are concluded in this experiment.

First, Motif-Backdoor achieves the SOTA performance compared with baselines. Motif-Backdoor has the best attack performance among six attack methods in terms of ASR and AMC. Take the backdoor attack on the PROTEINS dataset and the GCN model as an example. Motif-Backdoor shows the ASR of 88.59%, while GTA (second in backdoor attacks) and Motif-R (third in backdoor attacks) can reach the ASR of 73.16% and 71.92%, respectively.

In fact, across the four datasets and three models, Motif-Backdoor achieves an average BAD of 0.0265, which is lower than MaxDCC (second in backdoor attacks) and MIA (third in backdoor attacks), with an average BAD of 0.0275 and 0.0292, respectively.

Motif-Backdoor’s superiority over other baselines can be attributed to several reasons. First, Motif-Backdoor leverages the knowledge of motif distribution to select an effective trigger for backdoor attacks. This approach allows the attacker to find a trigger that can activate the backdoor with minimal perturbation to the benign samples, ensuring that the performance of the backdoored model remains comparable to the benign model. Second, Motif-Backdoor uses a novel trigger injection position optimization method that combines both the graph importance index and subgraph importance from the shadow model. This approach ensures that the trigger is inserted into a structurally important region of the graph, thereby minimizing the impact on the overall graph structure while maximizing the backdoor’s effectiveness.

Second, Motif-Backdoor is a powerful technique that not only successfully backdoors the model but also ensures normal performance of the backdoored model. The backdoor attack on the AIDS dataset and the GIN model, for instance, achieved a BAD of 0.0051, indicating that the performance of the backdoored model is similar to that of the benign model. In fact, across the four datasets and three models, Motif-Backdoor achieves an average BAD of 0.0265, which is lower than MaxDCC (second in backdoor attacks) and MIA (third in backdoor attacks), with an average BAD of 0.0275 and 0.0292, respectively.
This impressive performance can be attributed to two factors. First, in optimizing the position of the trigger injection, we construct a shadow model that takes into account the impact on the model, ensuring that the backdoor samples with the trigger are similar to the distribution of the target label samples selected by the attacker. Second, we use the graph importance index to screen the trigger nodes, selecting structurally important nodes that do not cause significant damage to the graph structure when used as triggers. These two factors together ensure that Motif-Backdoor is an effective and reliable technique for backdooring graph-based machine learning models.

Third, from the model and the dataset perspective, the GIN model and the NIC1 dataset are more vulnerable to backdoor attacks in the three models and four datasets. There are differences in the attack effects of backdoor attacks on various GNNs. For instance, in the AIDS dataset, Motif-Backdoor reaches the ASR of 88.26%, 71.35%, and 89.08% on GCN, SAGPool, and GIN, respectively. We believe that the difference in attack effect is caused by the difference in the way the model extracts network features. Specifically, SAGPool adopts a self-attention mask strategy for feature aggregation, which drops the information of some nodes. This indicates that the trigger nodes may be dropped and affect the backdoor attack.

Furthermore, from the dataset point view, we observe that the difference in attack performance caused by the difference in the dataset is obvious. It is worth noting that Motif-Backdoor reaches the ASR of 99.43% at the Gin model on the NCI1 dataset, but only 72.78% on the DBLP_v1 dataset. The reason is that the DBLP_v1 dataset has more graphs than other datasets, i.e., it has tens of thousands of graphs. In the training phase of the target model, the number of benign graphs is much more than the number of backdoored graphs, which makes it more difficult for the target model to leave a backdoor.

**D. Series Triggers on the Backdoor Attack (RQ3)**

In this section, we explore the impact of the trigger choice at different phases (i.e., the training phase and the inference phase) of the backdoor attack on the attack effect. Specifically, we distinguish the triggers used in the training phase and the inference phase. According to the various motifs studied in this work, the backdoor attack is sequentially matched to each other on four datasets. To more intuitively observe the change in the backdoor attack performance caused by the change in the time motif, we directly select the motif as the trigger to randomly inject it into the benign sample to initiate the backdoor attack, namely, Series-Backdoor. The results are shown in Fig. 6. Some observations are concluded in this experiment.

First, Motifs with similar structures as the trigger have similar backdoor attack effects. Take the backdoor attack on the AIDS dataset as an example, we observed the similar attack effects from $M_{32}$ and $M_{43}$ as the trigger in the training phase or the inference phase. In the case of using the trigger as $M_{32}$ in the training phase, $M_{32}$ and $M_{43}$ reach the ASR of 84.13% and 84.63% in the inferring phase, respectively. Intuitively, we can find that the color depth of the heat map brought by $M_{32}$ and $M_{43}$ is similar, and we group such motifs into the same series.

The reason is that the similarity between the structures is high, so that the model obtains similar features when aggregating the trigger features. For instance, both $M_{32}$ and $M_{43}$ are closed-loop structures. Intuitively, the model uses the mean strategy to aggregate node features. The extracted features based on the two motifs are the same, which also indicates the similarity of the two motifs. Therefore, we believe that the same series of motifs, i.e., with the similar structures, can replace each other in the attack. Considering the attack effect and structure realized by the motif as a trigger, we group $M_{31}$, $M_{41}$, and $M_{42}$ into the same series, as well as $M_{44}$, $M_{45}$, and $M_{46}$.

Second, the same motif as the trigger during the training and the inference does not show the best attack effect at the whole time. It is worth noting that in most cases, we would consider using the same trigger for the training phase and inference phase to be the best. However, in the case of using the trigger as $M_{44}$ in the training phase on the PROTEINS dataset, $M_{44}$ and $M_{46}$ reach the ASR of 63.33% and 71.75% in the inferring phase, respectively. Besides, there are several other examples with similar situations, e.g., $M_{42}$ in the training phase on the NCI1 dataset.

We speculate the reason is that there is a mutually inclusive relationship between the motif structures. Specifically, $M_{46}$ contains the structures of the remaining motifs, and they can all find corresponding subgraphs in $M_{46}$. This means that $M_{46}$ can activate the backdoor left by the remaining motifs as triggers in the training. But it does not mean that the more complex the trigger, the better the attack. For instance, $M_{41}$ and $M_{42}$ can accomplish better attack results than $M_{46}$ in DBLP_v1 dataset. In addition to considering the complexity of the trigger structure, it is also necessary to consider the distribution of the motifs in the dataset to reach an effective backdoor attack.

**E. Defense Against Motif-Backdoor (RQ4)**

We evaluate the attack performance of Motif-Backdoor under possible defenses. Due to the current lack of research on the backdoor attack defense in GNNs, inspired by Wu et al. [52], we transfer the defense method against the adversarial attack to evaluate the Motif-Backdoor. They used the Jaccard index to calculate the similarity between nodes. Based on the node similarity, they filtered and removed abnormal connections, realizing defense against adversarial attacks. For the backdoor attack, the attacker needs to use the trigger to activate the backdoor in the model, causing the model to predict incorrectly. This also means that the defender can destroy the triggers in the input graphs, thus achieving the purpose of defending against the backdoor attack. Specifically, in the inference, we use the Jaccard index to calculate the similarity of nodes in the graph, dropping the links whose similarity ranks in the bottom 10%. We conduct defense experiments on three models and four datasets, where Motif-Backdoor-Def represents Motif-Backdoor with the Jaccard defense mechanism. The results are shown in Fig. 7.
We can observe that Motif-Backdoor and Motif-Backdoor-Def perform similarly on ASR and AMC. Take the backdoor attack on the AIDS dataset on the GIN model as an example, Motif-Backdoor achieves the ASR of 99.72% and the AMC of 0.9943, while Motif-Backdoor-Def reaches the ASR of 97.68% and the AMC of 0.9795. According to the attack results of Motif-Backdoor-Def, the defense mechanism based on the Jaccard index cannot play a satisfactory defense effect. The reason is that the defense method of dropping the part links may destroy the integrity of the trigger, but it is difficult to completely remove the trigger from the input data. In Section VI-D, it has been verified that the same series of subgraphs can also activate the backdoor in the model. We conclude that the defense method of deleting the part links leads to the fact that the trigger subgraph structure only changes in the same series, which can still be used for the backdoor attack.

F. Parameter Sensitivity Analysis and Visualization (RQ5)

In this section, we first analyze the influence of several key hyperparameters in different ranges on the effect of Motif-Backdoor. Then, we visualize the benign graphs and the backdoored graphs to explore the stealthiness of Motif-Backdoor intuitively.

1) Parameter Sensitivity Analysis: We study the sensitivity of several key hyperparameters, i.e., number of filter node \( k \) and poisoning rate \( p \), at different scales. Motif-Backdoor performs hyperparameter analysis on four datasets, as shown in Fig. 8. For the number of filter nodes \( k \), we observe that the performance of Motif-Backdoor is stable under the change in \( k \). For instance, on the NCI1 dataset and the GCN model in Fig. 8(a), Motif-Backdoor reaches the ASR of 98.34%, 98.55%, 98.26%, 98.06%, and 97.72% in the range of the number of filtered nodes, respectively. The possible reason is that the nodes selected by the graph importance index take into account the graph structure. They need further feedback from the shadow model to determine the trigger injection position. This also indicates that the trigger injection position is more dependent on the subscore. The graph index plays the role of filtering unimportant nodes for the first time, which reduces the number of nodes that are subsequently considered by the model.
In Fig. 8(b), for the poison rate $p$, the ASR achieved by Motif-Backdoor increases as the poison rate increases. Take the backdoor attack on the DBLP_v1 dataset on the GIN model as an example, Motif-Backdoor achieves the ASR of 64.01%, 67.22%, 72.78%, 73.10%, and 75.88% in the range of the number of filtered nodes, respectively. We speculate that with the increase in the poison rate $p$, the number of backdoored graphs has increased. This makes the backdoored graphs participate in the training of the target model to a greater extent, increasing the probability of leaving a backdoor in the target model. In addition, the increase in the number of backdoored graphs also increases the risk of the backdoor attack being detected. Therefore, it is critical to select an appropriate poisoning ratio. We observe that Motif-Backdoor can achieve a satisfactory attack effect when the poisoning ratio is equal to 0.1. In most cases, backdoor attacks can choose to set the poisoning ratio to 0.1, which takes into account the effectiveness and stealthiness of the attack.

2) Visualization: To explore the stealthiness of Motif-Backdoor, we visualize the backdoored graphs by the Motif-Backdoor on three models and four datasets, as shown in Fig. 9. Intuitively, Motif-Backdoor does not do much damage to the benign graphs, and only a few links are needed to modify the graph to conduct effective attack. The visualization of the benign graphs and the backdoored graphs is similar. We speculate that the trigger injection position plays a key role in the stealthiness of the graph. Motif-Backdoor considers the graph structure and model feedback to select an appropriate trigger injection position.

VII. CONCLUSION

In this work, based on the differences in the distribution of network motifs, we obtain two interesting insights on
the backdoor attack. Then, we propose a base motif backdoor attack framework against GNNs, named Motif-Backdoor which uses the distribution difference in the motifs in the dataset to search the trigger structure. Besides, it adopts the graph importance index and the defined subgraph score to find the trigger injection position. Extensive experiments show that Motif-Backdoor can achieve the SOTA performance. Even with the deployment of the defense mechanism, Motif-Backdoor still realizes impressive attack results.

However, Motif-Backdoor is still challenged in several aspects. It requires the confidence score of the input sample by accessing the target model extensively to construct the shadow model, which increases the risk of an attack being detected. Moreover, it is necessary to further pay attention to effective defense strategies on GNNs in different downstream tasks, e.g., node classification, edge classification, and structural role classification. The interpretability of backdoor attacks is also interesting.

REFERENCES

[1] X.-J. Zhang, J. Wang, X.-J. Ma, C. Ma, J.-Q. Kan, and H.-F. Zhang, “Influence maximization in social networks with privacy protection,” Phys. A, Stat. Mech. Appl., vol. 607, Dec. 2022, Art. no. 128179.
[2] F. Zola, L. Segurola-Gil, J. L. Bruse, M. Galar, and R. Orduna-Urrutia, “Network traffic analysis through node behaviour classification: A graph-based approach with temporal dissection and data-level pre-processing,” Comput. Secur., vol. 115, Apr. 2022, Art. no. 102632.
[3] J. Chen et al., “Software visualization and deep transfer learning for process fault diagnosis,” IEEE Trans. Ind. Informat., vol. 16, no. 3, pp. 1949–1958, 2020.
[4] Y. Ma, S. Wang, T. Derr, L. Wu, and J. Tang, “Graph adversarial attack via rewiring,” in Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining, Singapore, Aug. 2021, pp. 1161–1169.
[5] J. Li, T. Xie, L. Chen, F. Xie, X. He, and Z. Zheng, “Adversarial attack on large scale graph,” IEEE Trans. Knowl. Data Eng., vol. 35, no. 1, pp. 82–95, May 2021.
[6] J. Chen, J. Zhang, Z. Chen, M. Du, and Q. Xuan, “Time-aware gradient attack on dynamic network link prediction,” IEEE Trans. Knowl. Data Eng., vol. 35, no. 2, pp. 2091–2102, 2023.
[7] Z. Zhang, J. Jia, B. Wang, and N. Z. Gong, “Backdoor attacks to graph neural networks,” in Proc. 26th ACM Symp. Access Control Model Technol. (SACMAT), Shenzhen, China, Oct. 2021, 15:1–15:11.
[8] J. Xu, M. Xue, and S. Peck, “Explainability-based backdoor attacks against graph neural networks,” in Proc. 3rd ACM Workshop Wireless Secur. Mach. Learn. (WiseML@WiSec), Abu Dhabi, United Arab Emirates, Jul. 2021, pp. 31–36.
[9] Z. Xi, R. Pang, S. Ji, and T. Wang, “Graph backdoor,” in Proc. 30th USENIX Secur. Symp., M. Bailey and R. Greenstadt, Eds., Berkeley, CA, USA: USENIX Association, Aug. 2021, pp. 1523–1540.
[10] Y. Sheng, R. Chen, G. Cai, and L. Kuang, “Backdoor attack to graph neural networks based on subgraph trigger,” in Collaborative Computing: Networking, Applications and Worksharing (Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering), vol. 407, Cham, Switzerland: Springer, Oct. 2021, pp. 276–286.
[11] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Keshet, D. Chklovskii, and U. Alon, “Network motifs: Simple building blocks of complex networks,” Science, vol. 298, no. 5594, pp. 824–827, 2002.
[12] U. Alon, “Network motifs: Theory and experimental approaches,” Nature Rev. Genet., vol. 8, no. 6, pp. 450–461, 2007.
[13] Y. Zhuang, Z. Liu, P. Qian, Q. Liu, X. Wang, and Q. He, “Graph vulnerability detection using graph neural network,” in Proc. IJCAI, Long Beach, CA, USA, vol. 97, Jun. 2019, pp. 3734–3743.
[14] Y. Zhuang, Z. Liu, P. Qian, Q. Liu, X. Wang, and Q. He, “Smart contract vulnerability detection using graph neural network,” in Proc. IJCAI, 2020, pp. 3283–3290.
[15] J. Chen et al., “EGC2: Enhanced graph classification with easy graph compression,” Inf. Sci., vol. 629, pp. 376–397, Jun. 2023.
[16] J. Lee, I. Lee, and J. Kang, “Self-attention graph pooling,” in Proc. 36th Int. Conf. Mach. Learn. (ICML), in Proceedings of Machine Learning Research, Long Beach, CA, USA, vol. 97, Jun. 2019, pp. 3734–3743.
[17] M. Jia, D. Xu, T. Yang, Y. Liu, and Y. Yao, “Graph convolutional network soft sensor for process quality prediction,” J. Process Control, vol. 123, pp. 12–25, Mar. 2023.
[18] B. Perozzi, R. Al-Rfou, and S. Skiena, “DeepWalk: Online learning of social representations,” in Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD), New York, NY, USA, Aug. 2014, pp. 701–710.
[19] A. Grover and J. Leskovec, “node2vec: Scalable feature learning for networks,” in Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, San Francisco, CA, USA, Aug. 2016, pp. 855–864.
[20] D. Zügner, A. Akbarnejad, and S. Günnemann, “Adversarial attacks on neural networks for graph data,” in Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD), London, U.K., Y. Guo and F. Faraoun, Eds., Aug. 2018, pp. 2847–2856.
[21] M. Xu, Y. Ma, and Z. Chen, “Graph adversarial attack via rewiring,” in Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining, Singapore, Aug. 2021, pp. 1161–1169.
[22] H. Zheng et al., “Network motifs: Simple building blocks of complex networks,” Proc. Natl. Acad. Sci. U. S. A., vol. 104, no. 27, pp. 11237–11242, 2007.
[23] Y. Zhuang, Z. Liu, P. Qian, Q. Liu, X. Wang, and Q. He, “Smart contract vulnerability detection using graph neural network,” in Proc. IJCAI, Long Beach, CA, USA, vol. 97, Jun. 2019, pp. 3734–3743.
M. Zhao, Y. Zhang, X. Xia, and X. Xu, “Motif-aware adversarial graph representation learning,” IEEE Access, vol. 10, pp. 8617–8626, 2022.

L. Wang, J. Ren, B. Xu, J. Li, W. Luo, and F. Xia, “MODEL: Motif-based deep feature learning for link prediction,” IEEE Trans. Comput. Social Syst., vol. 7, no. 2, pp. 503–516, Apr. 2020.

Z. Yu and H. Gao, “MotifExplainer: A motif-based graph neural network explainer,” 2022, arXiv:2202.00519.

J. Piao, G. Zhang, F. Xu, Z. Chen, and Y. Li, “Predicting customer value with social relationships via motif-based graph attention networks,” in Proc. Web Conf. (WWW), Ljubljana, Slovenia, Apr. 2021, pp. 3146–3157.

T. Hočevar and J. Demšar, “A combinatorial approach to graphlet counting,” Bioinformatics, vol. 30, no. 4, pp. 559–565, Feb. 2014.

K. M. Borgwardt, C. S. Ong, S. Schönauer, S. V. N. Vishwanathan, A. J. Smola, and H. Kriegel, “Protein function prediction via graph kernels,” in Proc. 13th Int. Conf. Intell. Syst. Mol. Biol., Detroit, MI, USA, pp. 47–56.

R. Rossi and N. Ahmed, “The network data repository with interactive graph analytics and visualization,” in Proc. AAAI Conf. Artif. Intell., vol. 29, no. 1, 2015, pp. 1–2.

N. Shervashidze, P. Schweitzer, E. J. van Leeuwen, K. Mehlhorn, and K. M. Borgwardt, “Weisfeiler–Lehman graph kernels,” J. Mach. Learn. Res., vol. 12, pp. 2539–2561, Sep. 2011.

S. Pan, X. Zhu, C. Zhang, and P. S. Yu, “Graph stream classification using labeled and unlabeled graphs,” in Proc. IEEE 29th Int. Conf. Data Eng. (ICDE), Apr. 2013, pp. 398–409.

L. C. Freeman, “Centrality in social networks conceptual clarification,” Soc. Netw., vol. 1, no. 3, pp. 215–239, 1979.

J. Lee, I. Lee, and J. Kang, “Self-attention graph pooling,” in Proc. Int. Conf. Mach. Learn., 2019, pp. 3734–3743.

T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in Proc. 5th Int. Conf. Learn. Represent. (ICLR), Toulon, France, Apr. 2017, pp. 1–14.

J. Chen, D. Zhang, Z. Ming, K. Huang, W. Jiang, and C. Cui, “GraphAttacker: A general multi-task graph attack framework,” IEEE Trans. Netw. Sci. Eng., vol. 9, no. 2, pp. 577–595, Mar. 2022.

H. Wu, C. Wang, Y. Tsyshetskii, A. Docherty, K. Lu, and L. Zhu, “Adversarial examples on graph data: Deep insights into attack and defense,” 2019, arXiv:1903.01610.

Haibin Zheng received the B.S. and Ph.D. degrees from the Zhejiang University of Technology, Hangzhou, China, in 2017 and 2022, respectively. He is currently a University Lecturer with the Institute of Cyberspace Security, Zhejiang University of Technology. His research interests include deep learning and artificial intelligence security.

Haiyang Xiong is currently pursuing the master’s degree with the College of Information Engineering, Zhejiang University of Technology, Hangzhou, China. His research interests include graph data mining and applications, and artificial intelligence.

Jinyin Chen (Member, IEEE) received the B.S. and Ph.D. degrees from the Zhejiang University of Technology, Hangzhou, China, in 2004 and 2009, respectively. She studied evolutionary computing at the Ashikaga Institute of Technology, Ashikaga, Japan, in 2005 and 2006. She is currently a Professor with the Zhejiang University of Technology. Her research interests include artificial intelligence security, graph data mining, and evolutionary computing.

Huonan Ma is currently pursuing the master’s degree with the College of Information Engineering, Zhejiang University of Technology, Hangzhou, China. His research interests include graph data mining and applications, and artificial intelligence.

Guohan Huang is currently pursuing the master’s degree with the College of Information Engineering, Zhejiang University of Technology, Hangzhou, China. His research interests include graph data mining and applications, and artificial intelligence.