Semi-supervised Malicious Domain Detection Based on Meta Pseudo Labeling

Yi Gao[1,2], Fangfang Yuan[1](✉), Jinglin Yang[3], Dakui Wang[1], Cong Cao[1], and Yanbing Liu[1](✉)

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
{gaoyi,yuanfangfang,wangdakui,caocong,liuyanbing}@iie.ac.cn
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China
[3] National Computer network Emergency Response technical Team/Coordination Center of China (CNCERTCC), Beijing, China
yangjinglin@cert.org.cn

Abstract. The Domain Name System (DNS) is a crucial infrastructure of the Internet, yet it is also a primary medium for disseminating illicit content. Researchers have proposed numerous methods to detect malicious domains, with association-based approaches achieving relatively good performance. However, these methods encounter limitations in detecting malicious domains within isolated nodes and heavily relying on labeled data to improve performance. In this paper, we propose a semi-supervised malicious domain detection model named SemiDom, which is based on meta pseudo labeling. Firstly, we use associations among DNS entities to construct a semantically enriched domain association graph. In particular, we retain isolated nodes within the dataset that lack relationships with other entities. Secondly, a teacher network computes pseudo labels on the unlabeled nodes, which effectively augments the scarce labeled data. A student network utilizes these pseudo labels to transform both the structure and attribute features to domain labels. Finally, the teacher network is constantly optimized based on the student's performance feedback on the labeled nodes, enabling the generation of more precise pseudo labels. Extensive experiments on the real-world DNS dataset demonstrate that our proposed method outperforms the state-of-the-art methods.

Keywords: Malicious domain detection · Semi-supervised Learning · Meta Pseudo Labels.

1 Introduction

Domain Name System (DNS) is a vital component of the Internet infrastructure, providing a crucial service of associating domains with IP addresses. Due to its critical role in the Internet, DNS has emerged as an attack vector for cybercriminals, who leverage it for malicious activities such as spamming, phishing, and

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63751-3_21

malware dissemination. Consequently, malicious domain detection is essential to maintaining cyberspace security. Early rule-based detection methods [12, 19] rely on list filtering. With malicious domains multiplied, the size of the rule base grew quickly, making it increasingly difficult to maintain and causing a decline in detection performance. To overcome these limitations, feature-based methods were developed [7, 8, 10, 20]. These methods extract domain features to train a detection model, but the effectiveness of detection is heavily dependent on feature selection, which requires expertise and is vulnerable to feature tampering by attackers. Recently, researchers [16, 22–24] have proposed association-based methods that model the correlation between domains and utilize known domains to infer unknown ones. These methods are able to detect a greater number of concealed malicious domains and deliver outstanding detection outcomes. Nevertheless, association-based methods encounter the subsequent two challenges:

- As shown in Fig. 1, there are actually many isolated domain nodes in the real DNS traffic. The association-based methods ignore these nodes to prevent affecting knowledge propagation on the domain association graph (DAG). This operation not only causes a significant loss of domain data, but also loses the ability to detect malicious domains hidden within isolated nodes.
- Since the domain blacklist covers a limited portion of the domains, the labels of most domain nodes are unknown. Fig. 1 depicts the training data used by association-based methods, which consists only of labeled domain data, ignoring the large amount of information in the unlabeled data.

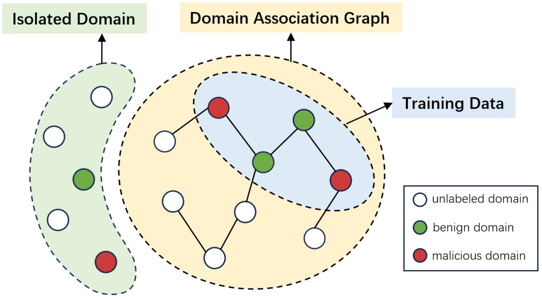


Fig. 1. A domain dataset constructed from real DNS traffic

To address the above challenges, we propose **SemiDom**, a semi-supervised malicious domain detection model based on meta pseudo labeling. In specific, we model the DNS scenario as a semantically rich DAG and preserves a large number of isolated domain nodes in the dataset. First, the *pseudo label generator* as a teacher network employs an adaptive label propagation algorithm to infer pseudo labels on unlabeled nodes in the DAG. Then, the *domain classifier*, which

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63751-3_21

is a student network, evaluates the efficacy of the pseudo label generation and provides feedback to the *pseudo label generator*. Next, the *pseudo-label generator* adjusts the label propagation strategy by the feedback to infer more accurate pseudo labels. Finally, the pseudo-labeled data effectively augments the existing labeled data and helps the *domain classifier* to transform structural features as well as the node attribute features into domain labels.

In summary, the main contributions of this paper are as follows:

1. We utilize the association among domain, IP addresses, and clients to build a DAG, and we additionally preserve a large number of isolated domain nodes in the dataset that are not associated to other entities.
2. We propose SemiDom, a semi-supervised meta pseudo labeling framework that mines the rich information implicit in the unlabeled domain data for malicious domain detection.
3. We conduct extensive experiments on a dataset constructed from real DNS traffic, and the experimental results demonstrate the effectiveness of our proposed method.

2 Related Work

2.1 Malicious Domain Detection

As DNS Flexible technology advances, the size of malicious domains is growing, making rule-based malicious domain detection methods less effective. Researchers [6, 9, 10, 21] have proposed feature-based detection methods, which train classifiers by extracting features from domain characters and DNS traffic. For example, Chin et al. [10] build a machine learning classifier to detect malicious domains using 27 features, including DNS records, average TTL, etc. However, attackers can modify the features of domains to evade the detection system. Recently, researchers [14, 16, 24, 25] have proposed to utilize hard-to-fake associations to detect malicious domains. These association-based methods construct the DNS scenario as a graph and utilize graph embedding algorithms, graph neural networks, etc. to accomplish domain classification. For example, Peng et al. [16] construct a bipartite graph between domains and IP addresses, using RF and XGBoost to classify domains. Wang et al. [24] model the DNS scene as a heterogeneous graph consisting of domains, clients, and IP addresses, and use the HAN model to detect malicious domains. Association-based methods have achieved well detection results, but these methods are unable to detect malicious domains hidden in isolated domain nodes, and also heavily rely on labeled data to train the model. Different from existing works, we propose a semi-supervised malicious domain detection model based on meta pseudo labeling, which utilizes unlabeled data to augment labeled data and is capable of detecting isolated malicious domain nodes.

2.2 Meta Pseudo Labels

Meta Pseudo Labels (MPL) [18] is one of the state-of-the-art semi-supervised learning methods that adopts a Teacher-Student architecture. The teacher network generates pseudo labels for unlabeled data, and the student network learns knowledge on labeled data. In particular, the teacher in Meta Pseudo Labels is not fixed. It is constantly optimized according to the student's performance on labeled data to generate better pseudo labels for teaching students. Peng et al. [17] proposed a federated meta pseudo labeling framework SynFMPL, to address the challenges of limited labeled data and data heterogeneity in federated learning. Meta Pseudo Labels has also been widely used in the field of anomaly detection, Zhao et al. [26] proposed a meta pseudo labeling based anomaly detection framework, MPAD. This framework seeks to obtain valid pseudo anomalies from unlabeled samples to complement the observed anomaly set. In addition, Zhou et al. [27] improved the Meta Pseudo labels for recommendation attack detection by using an experienced teacher network to generate a set of student networks instead of only one student in the original Meta Pseudo Labels. To the best of our knowledge, there is no prior work that applies the Meta Pseudo Labels to malicious domain detection.

3 Preliminary

Definition 1 ***Domain Association Graph.*** *We define a Domain Association Graph (DAG) as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{X}, \mathcal{Y})$, where $\mathcal{V}$ represents the set of domain nodes, and $\mathcal{E}$ represents the set of undirected edges. The feature vector of each domain is represented by $\mathcal{X} = (x_1, \ldots, x_n)$, and the corresponding label matrix is represented by $\mathcal{Y} = (y_1, \ldots, y_n)$.*

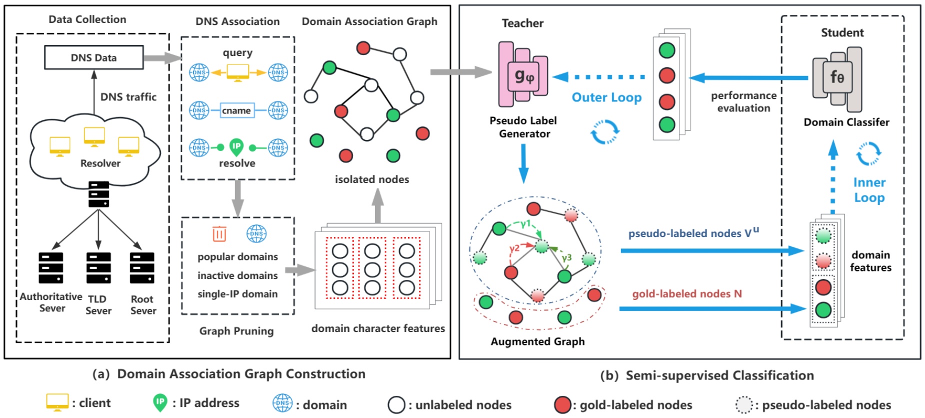
Definition 2 ***Pseudo Label and Gold Label.*** *In semi-supervised learning, human experts typically annotate a limited amount of unlabeled data. These manually annotated labels are **gold label**. In contrast, the model generates labels for the remaining unlabeled data. These labels generated through the model's predictions are **pseudo labels**.*

Definition 3 ***Semi-supervised Malicious Domain Detection.*** *The given domain dataset $\mathcal{D} = (\mathcal{G}, \mathcal{N})$ contains a DAG $\mathcal{G}$ and a set of isolated domain nodes $\mathcal{N}$ with gold labels. The node set in $\mathcal{G}$ is divided into an unlabeled node set $\mathcal{V}^u$ and a gold-labeled node set $\mathcal{V}^g$. Our goal is to learn a mapping function $\mathcal{U} : \mathcal{X} \to \mathcal{Y}$ that detects malicious domains using unlabeled data $\mathcal{V}^u$ and labeled data ($\mathcal{V}^g$, $\mathcal{N}$).*

4 Methodology

In this section, we describe SemiDom in detail. It consists of two parts: DAG construction and semi-supervised classification. The overall framework of SemiDom is shown in Fig. 2

DOI: 10.1007/978-3-031-63751-3_21



(a) Domain Association Graph Construction

(b) Semi-supervised Classification

Fig. 2. The overall framework of SemiDom

4.1 Domain Association Graph Construction

Data Collection. The DNS traffic offers comprehensive information about communication exchanges that take place between clients, resolvers, and upper-level DNS servers. We extract the query association between clients and domains, the resolution association between domains and IP addresses, and the CNAME of domains from the DNS traffic. Based on these three types of relationships, we construct a DAG with rich edges as shown in Fig. 2(a). The rules for adding edges in the domain graph are specified as follows:

- **Query:** An edge between two domain nodes is constructed if they have both been requested by the same client. This is because attacked clients are more likely to query malicious domains, while normal clients typically query benign ones.
- **Resolve:** Construct edges between domain nodes that resolve to the same IP address. This is due to the fact that such domains are often registered by the same entity and belong to the same category.
- **CNAME:** Connects a domain to its corresponding domains in the CNAME record. This is because domains in CNAME records usually belong to the same category.

It is important to note that in actual DNS traffic, not all domains are linked based on the three associations mentioned above. In fact, most domains are isolated. Unlike other association-based malicious domain detection methods, we reserve many isolated domain nodes within the dataset to empower the model to detect malicious domains among isolated nodes.

Graph Pruning. Since there are many noisy nodes in the real DNS traffic, which are not beneficial for information propagation and increase the computational pressure, we use the following three strategies to prune the association graph:

- **Popular domains:** These domains, which are requested by more than $T\%$ of clients, tend to be benign domains. This is because these popular domains can be quickly detected by the security management system if they are maliciously attacked.
- **Inactive domains:** The times of visits to these domains are less than Q, producing very few edges in the DAG and lacking valuable information
- **Single-IP Domain:** These domains can only resolve to a single IP address, usually belong to less important, temporary, or testing domains.

Domain Features. We refer to FANCI [20] to extract a total of 21 features (each with a dimension of 41) as the initialization vector of domains. These features consist of three categories: structural, semantic and statistical features, such as domain character length, digit ratio and n-gram frequency distribution.

4.2 Semi-supervised Classification

Semi-supervised classification of malicious domains relies on a meta pseudo labeling framework which comprises a teacher network *pseudo label generator* and a student network *domain classifier*. The *pseudo label generator* infers pseudo labels on unlabeled nodes to teach the *domain classifier*, while it is constantly adapted by the feedback of the student's performance on the labeled nodes.

Pseudo Label Generator (Teacher). To adaptively balance the label information of each node from different neighborhoods, the *pseudo label generator* g_Φ employs Adaptive Label Propagation (ALP) [11] algorithm to infer pseudo labels of unlabeled nodes. The goal of ALP is to get a an evenly smooth prediction matrix $\hat{\mathbf{Y}}$ through the label matrix $\mathbf{Y}$. Particularly, the propagation strategy of ALP can be described as the following equation:

$$\hat{\mathbf{Y}}_{i,:} = \sum_{k=0}^{K} \gamma_{ik} \mathbf{Y}_{i,:}^{(k)}, \quad \mathbf{Y}^{(k+1)} = \mathbf{T}\mathbf{Y}^{(k)}, \tag{1}$$

where $\mathbf{Y}^{(0)} = \mathbf{Y}$, $\mathbf{T}$ is the transition matrix, and $\mathbf{K}$ is the propagation step. γ_{ik} represents the influence of the k-hop neighborhood of node v_i, which is calculated using the attention mechanism described below:

$$\gamma_{ik} = \frac{\exp\left(\mathbf{a}^{\mathrm{T}} \operatorname{ReLU}\left(\mathbf{W}\mathbf{Y}_{i,:}^{(k)}\right)\right)}{\sum_{k'=0}^{K} \exp\left(\mathbf{a}^{\mathrm{T}} \operatorname{ReLU}\left(\mathbf{W}\mathbf{Y}_{i,:}^{(k')}\right)\right)}, \tag{2}$$

where $\mathbf{a}$ and $\mathbf{W}$ are the learnable attention vector and weight matrix, respectively. After $\mathbf{K}$ iterations, ALP learns a smooth predictive label matrix $\hat{\mathbf{Y}}$ that captures the label distribution of the k-hop neighborhood of node v_i. This matrix adjusts the impact of label propagation at each node while also capturing rich structural information on the graph.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63751-3_21

Domain Classifier (Student). After encoding the structural knowledge of the DGA into pseudo labels, we constructed a *domain classifier* $f_{\boldsymbol{\theta}}$ to transform the domain node features into node labels. The process of predicting labels can be formulated as follows:

$$\mathbf{P}_{i,:} = f_{\boldsymbol{\theta}}\left(\mathbf{X}_{i,:}\right), \tag{3}$$

where $f_{\boldsymbol{\theta}}$ is a multilayer perceptron with the addition of a softmax function. $\mathbf{X}_i$ and $\mathbf{P}_i$ are the feature and prediction label of the domain node v_i, respectively. The training data for the *domain classifier* consists of two parts: the set of unlabeled nodes $\mathcal{V}^u$ in the DAG and the set of gold-labeled isolated nodes $\mathcal{N}$.

Bi-level Optimization of Parameters. Ideally, generated pseudo labels should have the same contribution as the gold labels if they are accurate enough. Therefore, the optimization objective of SemiDom can be described as: *the generated pseudo labels should maximize the performance of the domain classifier at the gold-labeled nodes.* This objective implies a bi-level optimization problem with ϕ as the outer-loop parameters and θ as the inner-loop parameters.

Student (Inner-loop) update: For gold-labeled nodes sampled from the isolated node set $\mathcal{N}$, we use their real labels as the ground truth. However, for the unlabeled nodes sampled from $\mathcal{V}^u$, we use the generated pseudo labels as the ground truth. The *domain classifier* updates θ according to the following equation:

$$\boldsymbol{\theta}' = \boldsymbol{\theta} - \eta_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}} J_{\text{pseudo}}\left(\boldsymbol{\theta}, \phi\right), \tag{4}$$

where $\eta_{\boldsymbol{\theta}}$ denotes the inner learning rate. $J_{\text{pseudo}}\left(\boldsymbol{\theta}, \phi\right)$ denotes the loss of the *domain classifier* which is calculated on a batch of pseudo-labeled nodes and gold-labeled nodes.

Teacher (Outer-loop) update: The parameters of the *pseudo label generator* are updated with the learning rate η_ϕ as follows:

$$\phi' = \phi - \eta_\phi \nabla_\phi J_{\text{gold}}\left(\theta'(\phi)\right), \tag{5}$$

where $J_{\text{gold}}\left(\theta'(\phi)\right)$ is the outer loop loss computed on gold-labeled nodes which is back-propagated to calculated the gradient for the *domain classifier.*

To compute the gradient of ϕ, we utilize chain rule to differentiate $J_{\text{gold}}\left(\theta'(\phi)\right)$ with respect to ϕ through intermediate function $\theta'(\phi)$. This is expressed in the following equation:

$$\nabla_\phi J_{\text{gold}}\ \left(\boldsymbol{\theta}'(\phi)\right) \approx -\frac{\eta_\phi}{2\epsilon}\left[\nabla_\phi J_{\text{pseudo}}\ \left(\boldsymbol{\theta}^+, \boldsymbol{\phi}\right) - \nabla_\phi J_{\text{pseudo}}\ \left(\boldsymbol{\theta}^-, \boldsymbol{\phi}\right)\right], \tag{6}$$

where $\boldsymbol{\theta}'(\boldsymbol{\phi}) = \boldsymbol{\theta} - \eta_{\boldsymbol{\theta}} \nabla_{\boldsymbol{\theta}} J_{\text{pseudo}}\left(\boldsymbol{\theta}, \boldsymbol{\phi}\right)$, $\boldsymbol{\theta}^{\pm} = \boldsymbol{\theta} \pm \epsilon \nabla_{\boldsymbol{\theta}'} J_{\text{gold}}\ \left(\boldsymbol{\theta}'(\boldsymbol{\phi})\right)$, and ϵ is a small scalar used for finite difference approximation in the computation of gradients.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63751-3_21

In the above meta pseudo labeling framework, the *pseudo label generator* adjusts its label propagation strategy based on the *domain classifier*'s feedback to generate higher-quality pseudo labels. Using these improved pseudo labels, we can train a more precise and reliable *domain classifier*. The parameters of the *domain classifier* in the inner loop and the parameters of the *pseudo label generator* in the outer loop are updated alternatively. Algorithm 1 illustrates the complete detection algorithm.

Algorithm 1 The learning algorithm of SemiDom

```
Input: a DAG $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{X}, \mathcal{Y})$ with unlabeled node set $\mathcal{V}^u$ and
    gold-labeled node set $\mathcal{V}^g$, isolated node set $\mathcal{N}$, training epochs $E$,
    inner-loop learning rate $\eta_\theta$ and outer-loop learning rate $\eta_\phi$.
Output: The well-trained domain classifier
Initialize the parameters $\boldsymbol{\phi}$ and $\boldsymbol{\theta}$
while $e < E$ do
    Randomly sample a batch of nodes from $\mathcal{V}^u$ and $\mathcal{N}$.
    ▷ Pseudo Label Generation
    Compute the pseudo labels for sampled unlabeled nodes using
    the pseudo label generator $g_{\boldsymbol{\phi}}$.
    ▷ Inner-loop update for $\theta$.
    Compute $J_{\text{pseudo}}(\boldsymbol{\theta}, \phi)$ using the generated pseudo-labeled
    nodes and gold-labeled nodes.
    Update parameters $\theta$ of the domain classifier $f_{\boldsymbol{\theta}}$ via Eq.(4).
    ▷ Outer-loop update for $\phi$.
    Randomly sample a batch of nodes from $\mathcal{V}^g$ and $\mathcal{N}$
    Compute $J_{\text{gold}}(\theta'(\phi))$ on the labeled nodes using the updated
    domain classifier.
    Update parameters $\phi$ of the pseudo label generator $g_{\boldsymbol{\phi}}$ via
    Eq.(5) and Eq.(6).
end while
return The well-trained domain classifier
```

5 Experiments

In this section, we evaluate the performance of SemiDom by conducting experiments on a dataset constructed from real DNS traffic. Further, we analyze the impact of the *pseudo label generator* and the sensitivity of hyper-parameters.

5.1 Dataset

To evaluate the performance of the model, we collected actual DNS traffic data from August 31, 2020 to September 13, 2020 for a total of two weeks. In order to build a highly connected DAG, we construct connecting edges between two domain nodes as long as any one of the three associations mentioned in Section 4.1 exists between these two nodes. It is worth emphasizing that domains collected

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63751-3_21

from real traffic may not have any of the above associations, so we keep these isolated nodes in the dataset in order to detect malicious domains latent in them. In addition, we utilize the whitelist Alexa top 1M [1] to label benign domains, and the blacklists PhishTank [5], CoinBlockerLists [3], Malwaredomains [4] and AnudeepND [2] to label malicious domains. Finally, the dataset we constructed contains 101,023 isolated nodes, 104,583 associated nodes and 117,990 edges.

5.2 Baselines

In order to verify the effectiveness of our proposed SemiDom, we compared it with the following five baselines:

- **LP [28]:** LP (Label Propagation), a classical semi-supervised learning method, assumes that samples closely in the sample space are more similar. It ignores sample attribute features and uses only structural information.
- **GCN [15]:** GCN is a classical homogeneous graph neural network that uses the adjacency matrix and the node feature matrix to learn node representations through aggregation and convolution operations.
- **FANCI:** [20] FANCI is a feature-based malicious domain detection method. It analyzes the feature patterns of domains and classifies them using machine learning methods such as Support Vector Machine and Random Forest.
- **IpDom [13]:** This is a homogeneous graph-based method for malicious domain detection, which we refer to as IpDom for short. It utilizes the resolving relationship between domains and IP addresses to establish associations, and learns domain representations through an improved DeepWalk.
- **DeepDom [23]:** DeepDom is one of the state-of-the-art heterogeneous graph-based methods for malicious domain detection. It represents the DNS scenario as a heterogeneous graph, and uses short random walks based on meta-paths to guide the convolution operation.

5.3 Evaluation Metrics and Parameter Settings

To evaluate the models' performance, we use three standard metrics: precision (P), recall (R) and F1. Additionally, we draw ROC curves for each method, and the area under the curve (AUC) comprehensively evaluates the performance of the binary classification model. SemiDom is built on the PyTorch of version 1.9 and is executed for 200 iterations. The label propagation step K of the *pseudo label generator* is 10 and the learning rate η_ϕ is 0.0005. The *domain classifier* is configured with a 2-layer DNN and the learning rate η_θ is 0.001. The graph pruning strategy uses $T = 100$ and $Q = 5$. To ensure a fair comparison, the same number of isolated nodes are kept in the input data for all methods.

5.4 Performance Evaluation

Overall performance. For each method, we conduct three separate sets of experiments using 10%, 30%, and 50% of the training set. Table 1 shows the experimental result with the best one marked in bold. Fig. 3 illustrates the ROC curves for different models under the label ratio of 10%. Based on this information, we can draw the following conclusions:

1. SemiDom significantly outperforms the structure-based LP and feature-based FANCI. This is because SemiDom more comprehensively considers both the attribute features and structural knowledge of domains.
2. SemiDom exhibits superior performance compared to the IpDom. The reason is that SemiDom retains isolated domain nodes in real DNS traffic and can detect malicious domains hidden in isolated nodes.
3. SemiDom outperforms GCN and DeepDom due to the fact that these two approaches rely on a large amount of labeled data to improve model performance, ignoring the rich information hidden in unlabeled data.

Table 1. Performance comparison of all methods under different label ratios

Label Ratio	10 %			30 %			50 %		
Model	P	R	F1	P	R	F1	P	R	F1
LP	0.7327	0.7408	0.7366	0.7372	0.7402	0.7380	0.7401	0.7459	0.7423
GCN	0.8398	0.8427	0.8405	0.8440	0.8562	0.8499	0.8536	0.8619	0.8589
FANCI	0.8799	0.8650	0.8721	0.8802	0.8703	0.8780	0.8825	0.8754	0.8810
IpDom	0.7352	0.7388	0.7364	0.7423	0.7480	0.7454	0.7560	0.7594	0.7566
DeepDom	0.8982	0.9083	0.9010	0.9035	0.9165	0.9098	0.9156	0.9233	0.9189
SemiDom	**0.9519**	**0.9638**	**0.9553**	**0.9585**	**0.9780**	**0.9606**	**0.9610**	**0.9789**	**0.9676**

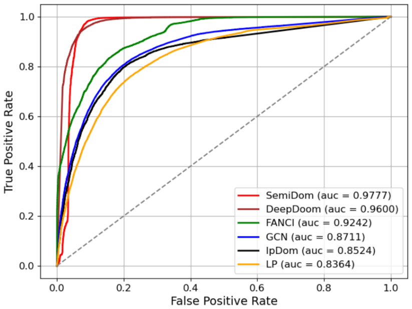


Fig. 3. ROC for each method

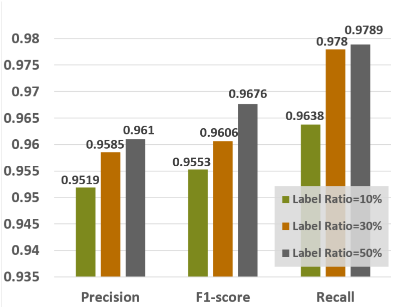


Fig. 4. Experimental results of SemiDom

SemiDom's performance with different labeling ratios. Fig. 4 visualizes the experimental results of SemiDom at different label ratios. It can be observed that when the ratio of labeled data grows, SemiDom exhibits improved classification performance. The results indicate that SemiDom can achieve superior detection performance on datasets with an increased proportion of labeled data.

SemiDom at 50% labeling vs. Baselines at 100% labeling. We train all supervised learning methods in the baselines on the full training set, while SemiDom is trained on only 50% of the training set. As we can see in Table. 2, SemiDom outperforms the other methods using only half of the labeled data, which fully validates the effectiveness of SemiDom.

Table 2. Performance: SemiDom at 50% Labeling vs. Baselines at 100% Labeling

Model	P	R	F1
GCN	0.8832	0.8901	0.8872
FANCI	0.8927	0.8990	0.8966
IpDom	0.7742	0.7787	0.7781
DeepDom	0.9339	0.9398	0.9361
SemiDom	**0.9610**	**0.9789**	**0.9676**

5.5 Model Analysis

Impact of the *pseudo label generator.* In order to verify the enhancement of the *pseudo label generator* to the *domain classifier*, we designed **w/o ALP**, in which we removed SemiDom's *pseudo label generator* and used labeled domain nodes to train the *domain classifier*. In this experiment, the label ratio of SemiDom is set to 10%. As shown in Fig. 5, SemiDom has obvious advantages over **w/o ALP** when the number of labeled data is limited. This is because the *pseudo label generator* generates pseudo labels for unlabeled nodes, effectively augmenting the labeling data, whereas **w/o ALP** learns only limited knowledge from the labeled data. Moreover, the pseudo labels generated based on the labels of neighboring nodes encode the structural information of the DAG, while **w/o ALP** focuses only on attribute features.

Hyper-parameter sensitivity study of *K*. The label propagation step K determines the number of times the label information is updated in the DAG. We study the hyper-parameter sensitivity of K in the *pseudo label generator*. As shown in Fig. 6, the performance of the model improves significantly as the number of iterations increases from $K = 2$ to $K = 10$. However, after $K = 10$ is reached, the performance gain from continuing to increase K diminishes and is accompanied by slower model convergence and more computational resource consumption. Considering the detection effect and training cost of the model, the number of steps for label propagation is set to $K = 10$.

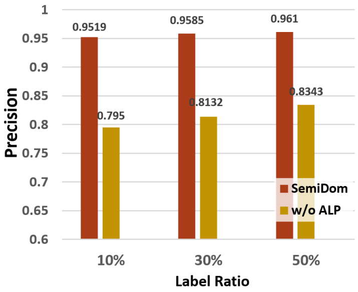


Fig. 5. Impact of pseudo label generator

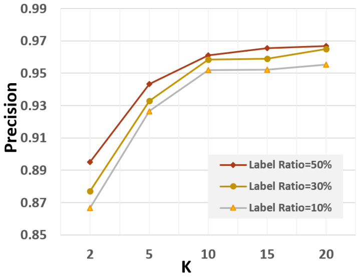


Fig. 6. Hyper-parameter sensitivity study

6 Conclusion

In this paper, we propose SemiDom, a semi-supervised malicious domain detection model based on meta pseudo labeling. We first model the DNS scenario as a domain association graph and retain isolated nodes in the dataset. We then employ a meta pseudo labeling framework which contains a teacher network *pseudo label generator* and a student network *domain classifier*. The *pseudo label generator* infers pseudo labels on unlabeled nodes to teach the *domain classifier*. Meanwhile, it constantly optimizes the label propagation strategy by the feedback from the *domain classifier*'s performance on the labeled nodes. Extensive experiments show that SemiDom outperforms other state-of-the-art methods even with limited labeled data.

7 Acknowledgment

This work is supported by Xinjiang Uygur Autonomous Region key research and development program (NO. 2022B03010).

References

1. alexa-top-sites (Aug 2022), https://aws.amazon.com/cn/alexa-top-sites/
2. Anudeepnd (Aug 2022), https://github.com/anudeepND/blacklist
3. Coinblockerlists (Aug 2022), https://gitlab.com/ZeroDot1/CoinBlockerLists
4. Malware domain block list (Aug 2022), http://www.malwaredomains.com/

5. Phishtank (Aug 2022), http://www.phishtank.com/
6. Anderson, H.S., Woodbridge, J., Filar, B.: Deepdga: Adversarially-tuned domain generation and detection. In: Freeman, D.M., Mitrokotsa, A., Sinha, A. (eds.) Proceedings of the 2016 ACM Workshop on Artificial Intelligence and Security. pp. 13–21. ACM (2016)
7. Antonakakis, M., Perdisci, R., Dagon, D., Lee, W., Feamster, N.: Building a dynamic reputation system for dns. In: 19th USENIX Security Symposium (USENIX Security 10) (2010)
8. Bilge, L., Sen, S., Balzarotti, D., Kirda, E., Kruegel, C.: Exposure: A passive dns analysis service to detect and report malicious domains. ACM Transactions on Information and System Security (TISSEC) **16**(4), 1–28 (2014)
9. Bilge, L., Sen, S., Balzarotti, D., Kirda, E., Kruegel, C.: Exposure: A passive dns analysis service to detect and report malicious domains. Acm Transactions on Information and System Security **16**(4) (2014)
10. Chin, T., Xiong, K., Hu, C., Li, Y.: A machine learning framework for studying domain generation algorithm (dga)-based malware. In: International Conference on Security and Privacy in Communication Systems (2018)
11. Ding, K., Wang, J., Caverlee, J., Liu, H.: Meta propagation networks for graph few-shot semi-supervised learning (2021)
12. Grill, M., Nikolaev, I., Valeros, V., Rehak, M.: Detecting dga malware using netflow. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM). pp. 1304–1309. IEEE (2015)
13. He, W., Gou, G., Kang, C., Liu, C., Xiong, G.: Malicious domain detection via domain relationship and graph models. IEEE (2019)
14. Khalil, I., Yu, T., Guan, B.: Discovering malicious domains through passive dns data graph analysis. In: Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security. pp. 663–674 (2016)
15. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)
16. Peng, C., Yun, X., Zhang, Y., Li, S.: Malshoot: Shooting malicious domains through graph embedding on passive dns data. In: Collaborative Computing (2018)
17. Peng, T., Chiu, T., Pang, A., Tail, W.: Synfmpl: A federated meta pseudo labeling framework with synergetic strategy. In: IEEE International Conference on Communications, ICC 2023, Rome, Italy, May 28 - June 1, 2023 (2023)
18. Pham, H., Dai, Z., Xie, Q., Le, Q.V.: Meta pseudo labels. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021 (2021)
19. Sato, K., Ishibashi, K., Toyono, T., Hasegawa, H., Yoshino, H.: Extending black domain name list by using co-occurrence relation between dns queries. IEICE transactions on communications **95**(3), 794–802 (2012)
20. Schüppen, S., Teubert, D., Herrmann, P., Meyer, U.: {FANCI}: Feature-based automated {NXDomain} classification and intelligence. In: 27th USENIX Security Symposium (USENIX Security 18). pp. 1165–1181 (2018)
21. Shi, Y., Chen, G., Li, J.: Malicious domain name detection based on extreme machine learning. Neural Process. Lett. **48**(3), 1347–1357 (2018)
22. Sun, X., Tong, M., Yang, J., Xinran, L., Heng, L.: {HinDom}: A robust malicious domain detection system based on heterogeneous information network with transductive classification. In: 22nd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2019). pp. 399–412 (2019)

DOI: 10.1007/978-3-031-63751-3_21

23. Sun, X., Wang, Z., Yang, J., Liu, X.: Deepdom: Malicious domain detection with scalable and heterogeneous graph convolutional networks. Computers & Security **99**, 102057 (2020)
24. Wang, Q., Dong, C., Jian, S., Du, D., Lu, Z., Qi, Y., Han, D., Ma, X., Wang, F., Liu, Y.: Handom: Heterogeneous attention network model for malicious domain detection. Computers & Security (2023)
25. Zhang, S., Zhou, Z., Li, D., Zhong, Y., Liu, Q., Yang, W., Li, S.: Attributed heterogeneous graph neural network for malicious domain detection. In: 2021 IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD). pp. 397–403. IEEE (2021)
26. Zhao, S., Yu, Z., Wang, X., Marbach, T.G., Wang, G., Liu, X.: Meta pseudo labels for anomaly detection via partially observed anomalies. In: Database Systems for Advanced Applications - 28th International Conference, DASFAA 2023, Tianjin, China, April 17-20, 2023, Proceedings, Part IV (2023)
27. Zhou, Q., Li, K., Duan, L.: Recommendation attack detection based on improved meta pseudo labels. Knowl. Based Syst. (2023)
28. Zhu, X.: Learning from labeled and unlabeled data with label propagation. Tech Report (2002)

DOI: 10.1007/978-3-031-63751-3_21