Heterogeneous Domain Adaptation for IoT Intrusion Detection: A Geometric Graph Alignment Approach

Jiashu Wu, *Graduate Student Member, IEEE*, Hao Dai, *Graduate Student Member, IEEE*, Yang Wang, *Member, IEEE*, Kejiang Ye, *Member, IEEE*, and Chengzhong Xu, *Fellow, IEEE*

***Abstract*—Data scarcity hinders the usability of data-dependent algorithms when tackling IoT intrusion detection (IID). To address this, we utilise the data rich network intrusion detection (NID) domain to facilitate more accurate intrusion detection for IID domains. In this paper, a Geometric Graph Alignment (GGA) approach is leveraged to mask the geometric heterogeneities between domains for better intrusion knowledge transfer. Specifically, each intrusion domain is formulated as a graph where vertices and edges represent intrusion categories and category-wise interrelationships, respectively. The overall shape is preserved via a confused discriminator incapable to identify adjacency matrices between different intrusion domain graphs. A rotation avoidance mechanism and a centre point matching mechanism is used to avoid graph misalignment due to rotation and symmetry, respectively. Besides, category-wise semantic knowledge is transferred to act as vertex-level alignment. To exploit the target data, a pseudo-label election mechanism that jointly considers network prediction, geometric property and neighbourhood information is used to produce fine-grained pseudo-label assignment. Upon aligning the intrusion graphs geometrically from different granularities, the transferred intrusion knowledge can boost IID performance. Comprehensive experiments on several intrusion datasets demonstrate state-of-the-art performance of the GGA approach and validate the usefulness of GGA constituting components.**

***Index Terms*—Internet of Things (IoT), Intrusion Detection, Domain Adaptation, Geometric Graph Alignment, Pseudo Label Election**

I. Introduction

INternet of Things (IoT) devices become indispensable for various real world applications and innovatively transform several fields such as healthcare [1], [2], etc. However, limited power and computational capability of IoT devices hinder the applicability of powerful security mechanisms, together with relatively infrequent maintenance, making IoT vulnerable to malicious intrusions. Therefore, to keep the IoT infrastructure safe, an effective IoT intrusion detection (IID) system is vital. To advance the intrusion detection techniques, several research directions become popular. Dietz [3] proposed to automatically scan IoT devices for pre-defined vulnerability patterns, and isolated suspicious devices to block the botnet spreading. McDermott [4] tackled the problem via deep recurrent neural network (RNN) and achieved satisfying detection performance. However, these efforts required either a thorough intrusion pattern repository, or abundant labelled training data, which is expensive to collect and time-consuming to annotate, and is especially difficult for IoT due to factors such as data privacy concerns, the frequent emergence of new IoT things, etc. Therefore, the data-scarcity of IoT hinders the usability of these rule or data-dependent methods.

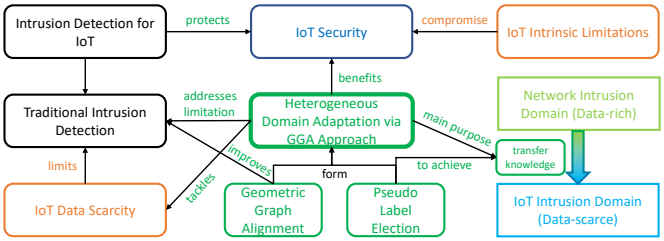


Fig. 1. The general motivation of the GGA approach

Considering that the Internet intrusion data is richer than IoT domains, and they share several similar intrusion categories, several domain adaptation-based (DA) methods were proposed to treat the network intrusion (NI) as the source domain, and transfer rich intrusion knowledge to facilitate the data-scarce target IoT intrusion (II) domains. By achieving domain-invariant alignment, the transferred intrusion knowledge can facilitate more accurate IID. For instance, Ning [5] presented a Laddernet-based DA solution to improve the intrusion classification accuracy and secure the industrial IoT infrastructures. Hu [6] studied a deep subdomain adaptation network with attention mechanism and focused on distribution alignment between domains via local maximum mean discrepancy. Methods such as [7], [8] proposed to achieve intrusion domain alignment by aligning the graph learning results. Efforts such as [9]–[11] attempted to explore unlabelled target domain via directly predicted, threshold selected or softly assigned pseudo-labels (PLs), respectively, to facilitate better intrusion knowledge transfer.

Despite their success, they left some deficiencies that need to be addressed. Previous DA-based ID methods didn't tackle the problem from a geometric graph perspective and failed to explore unlabelled target domain via geometric and neighbourhood-aware PLs. The ignorance of intrinsic geometric properties in domain graphs and the under-explored target

* Yang Wang is the corresponding author

Jiashu Wu, Hao Dai, Yang Wang and Kejiang Ye are with Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China. Email: {js.wu, hao.dai, yang.wang1, kj.ye}@siat.ac.cn

Jiashu Wu and Hao Dai are also with University of Chinese Academy of Sciences, Beijing 100049, China. Email: {wujiashu21, daihao19}@mails.ucas.ac.cn

Chengzhong Xu is with the State Key Laboratory of IoT for Smart City, Faculty of Science and Technology, University of Macau, Macau 999078, China. Email: czxu@um.edu.mo

Manuscript received January 00, 2022; revised January 00, 2023.

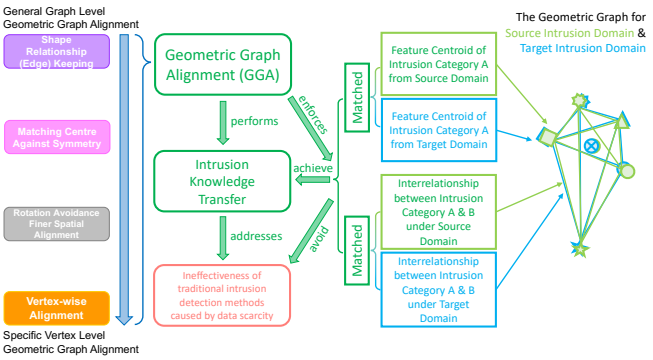


Fig. 2. An overview of the geometric graph alignment (GGA) mechanism

domain can result in coarse-grained alignment. Although some graph-based DA methods were proposed, they didn't attempt graph alignment from a pure geometric perspective, leaving the geometric properties under-explored. Despite some methods can partially convey the geometric properties through graph embedding, however, the embedding learning is highly data-dependent which is challenging for IoT scenarios. Besides, although there were some traditional PL-based DA methods, their isolated PL assignment strategy failed to leverage the geometric and neighbourhood information between labels, which may produce error-prone PLs and mislead the intrusion knowledge transfer.

To address these limitations and achieve more fine-grained intrusion knowledge transfer, following the motivation illustrated in Fig. 1 and 2, we propose a Geometric Graph Alignment (GGA) approach that works under the semi-supervised heterogeneous domain adaptation (HDA) setting, i.e., the target is scarcely-labelled and significant source-target heterogeneities exist, such as having diverse feature spaces, following different distributions, etc. To positively exploit the unlabelled target domain, we utilise a pseudo-label election (PLE) mechanism. To prevent the error-prone PL from misleading the model, the geometric property is considered to eliminate confident but incorrect PLs based on their geometric relationship with each category. Besides, the PLE consults the neighbourhood label information when assigning PLs to avoid near-boundary ambiguous PLs, which cannot be fulfilled by traditional isolated PL assignment strategies. By jointly considering the network prediction, the geometric property and the neighbourhood information, the PLE can boost pseudo label accuracy and hence lead to positive intrusion knowledge transfer.

The GGA then formulates each domain as a graph, where vertices and edges represent intrusion categories and their interrelationships. As illustrated in Fig. 2, enforcing a perfect geometric graph alignment can have each intrusion category and their interrelationships well aligned between domains. Firstly, with the help of the PLE, the GGA performs a graph-level shape keeping via a confused discriminator which is incapable to distinguish weighted adjacency matrices between intrusion domain graphs. Upon aligning the graph shapes, a centre point matching mechanism and a rotation avoidance mechanism avoid graph misalignment caused by symmetry and graph rotation, respectively. Finally, the GGA will perform a vertex-level matching by preserving categorical correlation knowledge between domains, which equivalently aligns the graph vertex of each intrusion category between domains. Holistically, they form a graph alignment framework from general to specific level from a geometric perspective. The GGA can robustly transfer the enriched intrusion knowledge from the NI domain to facilitate more accurate intrusion detection in the II domain and hence secure IoT infrastructures. GGA's motivation has been illustrated in Fig. 1-2.

In summary, the contributions of this paper are three-fold:

- We propose to transfer enriched intrusion knowledge from the NI domain to facilitate more accurate intrusion detection for data-scarce IoT domains and formulate it as a semi-supervised HDA problem.
- To our best knowledge, we are the pioneer to tackle this HDA problem from a pure geometric graph alignment perspective with the help of the PLE mechanism. Rather than using isolated coarse-grained PL strategies, the PLE makes fine-grained PL assignment by jointly considering geometric and neighbourhood information to filter confident but geometrically incorrect PLs and near-boundary ambiguous PLs. The GGA then aligns domain graphs geometrically through four mechanisms, holistically forming a graph alignment framework from graph to vertex granularity.
- We conduct comprehensive experiments of several tasks on 5 widely used intrusion detection datasets to verify the superior performance achieved by the GGA, and show the usefulness of its constituting components.

The rest of the paper is organised as follows: Section II presents related works by categories and demonstrates the research opportunities of the GGA method. Section III provides model preliminaries, graph formulations and the GGA model architecture. The detailed geometric graph alignment and pseudo label election mechanism are explained in Section IV. Section V presents experimental setups and result analyses. The last section concludes the paper. For better readability, an acronym table and a notation table have been presented in the Appendix section.

II. Related Work

IoT Intrusion Detection Methods The IoT intrusion detection (IID) has drawn wide research attention to secure IoT infrastructures. Rule-based IID methods were initially popular. Dietz [3] performed automatic scanning of neighbouring IoT devices for known vulnerability patterns and temporarily isolated detected compromised devices. Chen [12] proposed to filter security violations via complex event processing, which required a sophisticated rule repository. On the other hand, machine learning techniques were also widely used for IID. Anthi [13] presented a three layer intrusion detector that worked under a supervised fashion for smart home settings. Shukla [14] tackled the problem via a hybrid two-stage mechanism that combined Kmeans clustering and decision trees. On the deep learning perspective, models such as feedforward neural network, deep autoencoder and BiLSTM RNN were utilised to work on the IID problem by [15], [16] and [4], respectively. However, these methods either needed

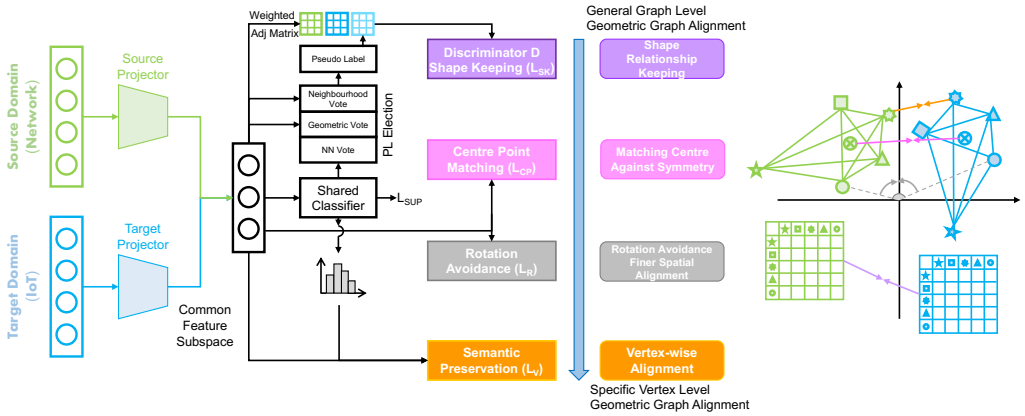


Fig. 3. The architecture of the GGA method

a sophisticated intrusion pattern repository, which requires substantial expertise to build and can hardly be complete and up-to-date, or required abundant amount of fully labelled data for training, which is labour-intensive to annotate. Hence, it naturally leads to domain adaptation-based methods which can comfortably work under the challenging data-scarce IID scenarios.

Domain Adaptation and Intrusion Detection Domain adaptation leverages source domain knowledge to facilitate better learning for data-scarce target domains, and hence is suitable to tackle the intrusion detection for the data-scarce IoT domain. Vu [17] utilised two autoencoders on source and target dataset and forced the alignment of the bottleneck layers. Later, Ning [5] leveraged the Laddernet to tackle IID under a semi-supervised setting. Hu [6] presented a deep subdomain adaptation network with attention mechanism that performed intrusion knowledge transfer by minimising local maximum mean discrepancy. However, these methods didn't use pseudo-label (PL) assignments to exploit unlabelled target domain, and failed to perform DA via a geometric graph-based approach, hence didn't preserve geometric properties during intrusion knowledge transfer. On the other hand, Chen [18] tackled the intrusion domain alignment problem via Transfer Neural Tree (TNT), a unified framework that combined feature mapping, adaptation and classification. A Generalised Joint Distribution Adaptation (G-JDA) approach was presented [9] to learn a pair of feature projectors to eliminate the marginal and conditional distribution divergence. Yao [19] proposed the DDA method that applied an adaptive classifier to reduce distribution divergence and enlarge inter-class divergence. The TNT, G-JDA and DDA applied direct prediction as PLs for unlabelled target instances and completely ignored the label neighbourhood information. Yao [11] put forward the STN, a conditional distribution alignment strategy with the help of a soft-label paradigm. Singh [20] presented the STAR framework, which emphasised unlabelled target instances based on the distance with closest class prototypes during intrusion domain alignment. Saito [21] achieved intrusion knowledge transfer by optimising the minimax loss on the domain conditional entropy (MME). It utilised unlabelled target data based on a threshold-based strategy. The APE [22] method chose to alleviate intra-domain discrepancy via three procedures, namely Attraction, Perturbation and Exploration. From a clustering-based perspective, Li [10] proposed Cross-Domain Adaptive Clustering (CDAC) to tackle the DA problem. In MME, APE and CDAC, unlabelled target instances will be pseudo-labelled based on a threshold strategy. However, when assigning PLs, these method either failed to jointly consider geometric properties, or assigned PLs in an isolated manner that ignored the relationships between pseudo-labelled instances and their neighbouring labelled instances, compromising accurate intrusion knowledge transfer. Some methods also required a manual threshold set based on prior experience and was not generalisable between tasks.

Tackling intrusion domain alignment from a graph perspective is also feasible. For example, the WCGN method matched domains via graph learning [7], [8] to benefit the domain alignment. Pilanci [23] proposed a graph base alignment method by transferring the graph spectrum information. However, although these embedding-related methods can partially convey the geometric information of domains, learning a good embedding is highly data-dependent, hindering their applicability. Besides, none of these graph-based methods solved the graph matching problem from a pure geometric graph alignment perspective, which left a void to be filled.

Our method tackles the HDA problem from a pure geometric graph perspective, which jointly considers several levels of geometric property matching. The GGA method does not require a huge amount of data for graph embedding learning and enjoys a relatively low complexity. Besides, we utilise a pseudo-label election mechanism which jointly accounts for the geometric properties and the neighbourhood information, so that the confident but wrong PL prediction that violates geometric properties and near-boundary ambiguous PLs can be avoided for positive transfer. Finally, we utilise the GGA method to facilitate more accurate intrusion detection for the data-scarce IoT domain.

III. Model Preliminary and Architecture

A. Model Preliminary

The geometric graph alignment (GGA) method works under a semi-supervised heterogeneous DA setting. It involves a

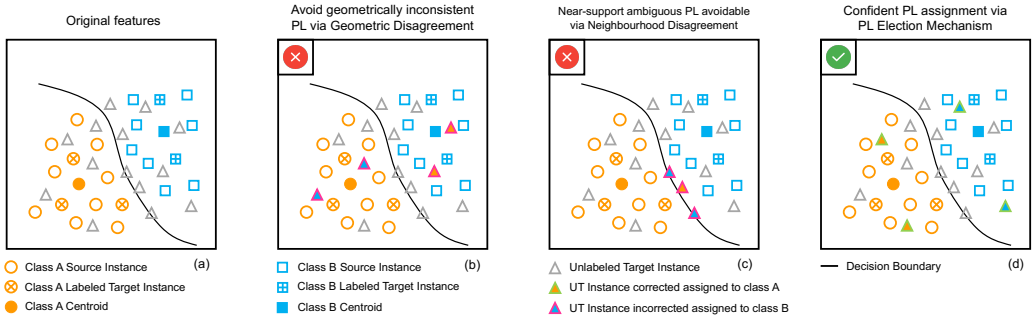


Fig. 4. Illustration of the PLE. (a) the original feature; (b) avoid geometrically inconsistent PL via geometric disagreement; (c) near-boundary ambiguous PL assignment avoidable via neighbourhood disagreement; (d) PLE's assignment.

source NI domain formulated as follows:

$$\begin{aligned} &\mathcal{D}_S = \{\mathcal{X}_S, \mathcal{Y}_N\} = \{(x_{Si}, y_{Si})\}, i \in [1, n_S], \\ &x_{Si} \in \mathbb{R}^{d_S}, y_{Si} \in [1, K], \end{aligned} \tag{1}$$

where the source NI domain contains n_S traffic records with their corresponding intrusion label. Each record is represented using d_S features, and there are K categories. Similarly, the target II domain are defined as follows:

$$\begin{aligned} &\mathcal{D}_{TL} = \{\mathcal{X}_{TL}, \mathcal{Y}_{TL}\} = \{(x_{TLi}, y_{TLi})\}, i \in [1, n_{TL}], \\ &\mathcal{D}_{TU} = \{\mathcal{X}_{TU}\} = \{(x_{TUj})\}, j \in [1, n_{TU}], \\ &\mathcal{D}_T = \mathcal{D}_{TL} \cup \mathcal{D}_{TU}, x_{TLi}, x_{TUj} \in \mathbb{R}^{d_T}, y_{TLi} \in [1, K], \\ &n_T = n_{TL} + n_{TU}, n_{TL} \ll n_{TU}. \end{aligned} \tag{2}$$

Under the semi-supervised setting, the target domain is scarcely-labelled, i.e., $n_{TL} \ll n_{TU}$. The source and target domain present heterogeneities such as belonging to different feature spaces, i.e., $d_S \neq d_T$.

B. Graph Formulation

To perform the geometric graph alignment, we formulate each intrusion domain as a graph, i.e., $G_X =< V_X, E_X >$, $X \in \{S, T\}$. Both domains share K intrusion categories, therefore, each graph has K vertices, the vertex V_i is the centroid of the category i, denoted as follows:

$$V_X^i = \frac{1}{n_X^i} \sum_{j=1}^{n_X^i} x_{X_j}, i \in [1, K], X \in \{S, T\}, \tag{3}$$

where n^i is the number of records under category i. Both graphs are formulated as a complete graph, the weight of edge $E < V_X^i, V_X^j >$ is set to be the Euclidean distance between vertex V_X^i and V_X^j. The corresponding weighted adjacency matrices (WAMs) are denoted as M_S and M_T.

C. Model Architecture and Overview

The architecture of the GGA method has been shown in Fig. 3. Each intrusion domain has a feature projector that maps features into a common feature subspace with dimension d_C. The feature projector is defined as follows:

$$\begin{aligned} &f(x_i) = \begin{cases} E_S(x_i) & \text{if } x_i \in \mathcal{X}_S \\ E_T(x_i) & \text{if } x_i \in \mathcal{X}_T = \mathcal{X}_{TL} \cup \mathcal{X}_{TU} \end{cases} \\ &f(x_i) \in \mathbb{R}^{d_C}. \end{aligned} \tag{4}$$

The GGA method will then utilise the pseudo-label election (PLE) mechanism to assign fine-grained PLs to unlabelled target data and avoid error-prone PLs to mitigate negative transfer. To perform geometric graph alignment between these heterogeneous domains, the WAM of the source data, the labelled target data, and the combination of labelled and pseudo-labelled target data will be generated to confuse the discriminator D. Highly similar WAMs indicate well-aligned intrusion categories and the fine preservation of category-wise interrelations, and is equivalent with a perfect geometric graph shape keeping. By fusing three WAMs, the geometric shape of domain graphs are aligned, meanwhile the labelled and pseudo-labelled target data will be better fused together. After keeping the shape, the domain graphs can still misalign due to rotation and symmetry, which are mitigated by the rotation avoidance mechanism and the centre point matching. Besides, categorical correlations yielded by the shared classifier C will be preserved between domains, which acts as a vertex-level alignment mechanism. Holistically, the GGA approach aligns the domain graphs from general shape level to specific vertex level. The motivation is to align the domain graphs in a fine-grained manner, such that the shared classifier C yields the best intrusion detection accuracy for unlabelled target domain.

IV. The GGA Algorithm

In this section, we will firstly introduce the pseudo-label election mechanism which facilitates better target participation during the geometric graph alignment process. Then, the geometric graph alignment process and its constituting components will be explained.

A. Pseudo-label Election Mechanism

Assigning pseudo-labels to unlabelled target data can excavate its potentials during intrusion knowledge transfer. However, erroneous PL assignment may mislead the model towards negative transfer. Traditional efforts mainly assigned PL to instances in an isolated manner, without considering the relationship between labels, and suffered from issues such as confident but geometrically-inconsistent PLs and near-support ambiguous PLs. Therefore, we utilise the Pseudo-Label Election (PLE) mechanism to mitigate these issues as much as possible. The PLE jointly considers the voting of NN prediction, the geometric properties and neighbourhood information. A PL assignment will only be made for an instance if these three votes reach a consensus. When producing the

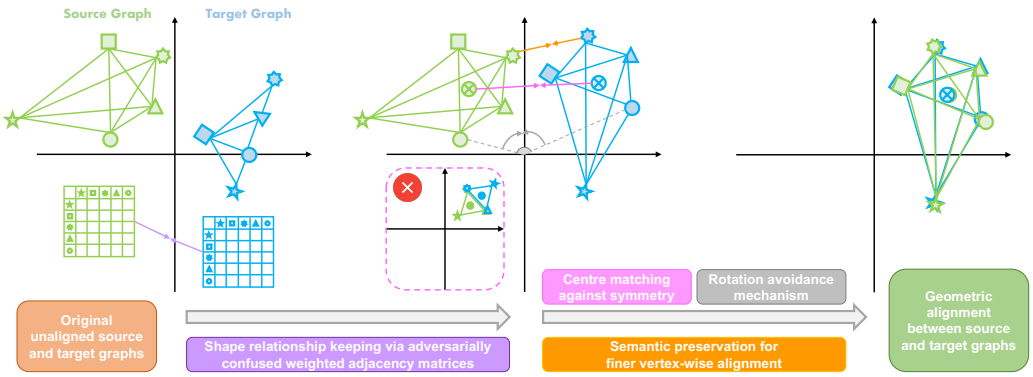


Fig. 5. Illustration of the Geometric Graph Alignment (GGA) method.

geometric property-based PL, the category of the most Cosine-similar labelled data centroid μ^i_{S+TL} will be utilised as PL_G for each unlabelled target instance and is defined as follows:

$$\begin{aligned} PL^i_G &= \underset{k}{\operatorname{argmax}}\, COS(\mu^{(k)}_{S+TL}, x^i_{TU}), \\ \mu^{(k)}_{S+TL} &= \frac{1}{n^{(k)}_S + n^{(k)}_{TL}} \sum_{x_i \in \mathcal{X}^{(k)}_S \cup \mathcal{X}^{(k)}_{TL}} x_i \,, \end{aligned} \tag{5}$$

where PL^i_G denotes the geometric-based PL for the x^i_{TU}, $\mathcal{X}^{(k)}_S$ denotes source instances from category k. If the NN-prediction yields a confident PL prediction but is inconsistent with the geometric similarity property, then such confident but contradictary PL will be rejected as illustrated in Fig. 4 (b). Besides, the PLE will also consult the neighbourhood information when assigning PLs. If the K-nearest neighbourhood around an unlabelled target instance cannot reach a majority agreement, or reach an agreement against the NN prediction or the geometric vote, then such assignment will also be rejected. This is useful especially when deciding the PL for near-boundary unlabelled instances, as illustrated in Fig. 4 (c). Since the neighbourhood can be harder to reach an agreement near the boundary due to ambiguity, the PLE can effectively get rid of near-boundary PLs which are more likely to be erroneous. Overall, the PLE will only assign confident PLs with probabilistic, geometric and neighbourhood soundness, which can significantly boost the PL accuracy and therefore lead to positive intrusion knowledge transfer.

B. Geometric Graph Alignment

The GGA method has been illustrated in Fig. 5. It will perform geometric graph alignment from the general graph granularity to the specific vertex granularity, i.e., the shape keeping via confused discriminator (*purple step*), rotation avoidance mechanism (*grey step*), centre point matching against symmetry (*pink step*) and the vertex-level alignment via semantic preservation (*orange step*).

Shape Keeping Firstly, the GGA with align the graph shape via a confused discriminator. We define the *Same Shape Rule* as follows:

Definition 1. ***Same Shape Rule****: Both G_S and G_T have their shape aligned with each other if and only if the weighted adjacency matrices (WAMs) M_S and M_T are the same.*

Specifically, with the help from the PLE, the GGA will construct three WAMs: the source WAM M_S, the labelled target WAM M_{TL} and a WAM based on both labelled and pseudo-labelled target data M_{TL+PL}. These WAMs will then be flattened and feed into the discriminator D, a single-layer neural network that tries to distinguish the origin of the input WAM. The source domain WAM M_S is assigned with domain label 1, while target domain WAMs are assigned with domain label 0. The shape keeping loss $\mathcal{L}_{SK}$ is defined as follows:

$$\begin{aligned} \mathcal{L}_{SK} &= \log(D(M_S)) \\ &\quad + \frac{1}{2} \sum_{M \in \{M_{TL}, M_{TL+PL}\}} (1 - \log(D(M))) \end{aligned} \tag{6}$$

The source and target projector will try to minimise the $\mathcal{L}_{SK}$ and let the discriminator D to be unable to distinguish the origin of the input WAMs, while the discriminator will try to stay unconfused. Upon this minimax game reaches an equilibrium, both G_S and G_T will have their shape aligned as indicated in Fig. 5 (Mid), and the labelled and pseudo-labelled target data will be better fused together.

Rotation Avoidance against Rotated Misalignment As indicated in Fig. 5 (Mid), graph rotation can still cause domain graph misalignment, even though the shape is aligned. Therefore, to further align the domain graphs geometrically, we define the *Same Angle Rule* as follows:

Definition 2. ***Same Angle Rule*** *For graph G_S and G_T, the Same Angle Rule holds if $\forall i \in [1, K], 1 - COS(V^i_S, V^i_T) = 0$.*

The GGA method will keep the Same Angle Rule holding by minimising the rotation loss, defined as follows:

$$\mathcal{L}_R = \sum_{i=1}^{K} (1 - COS(V^i_S, V^i_T)) \tag{7}$$

By minimising $\mathcal{L}_R$, graph misalignment caused by rotation will be prevented since the categorise-wise vertex angle is enforced to be $0°$.

Centre Point Matching against Symmetric Misalignment As shown in the pink-boxed example in Fig. 5 (Mid), upon fixing the graph shape and enforcing the Same Angle Rule, symmetry can still cause misalignment. Therefore, we further define the *Same Centre Rule* as follows:

Definition 3. ***Same Centre Rule*** *For graph G_S and G_T, the Same Centre Rule holds if $\mu_S = \mu_T$, $\mu_X = \frac{1}{n_X}\sum_{i=1}^{n_X}\mathcal{X}_{Xi}, X \in \{S,T\}$.*

We minimise the centre point matching loss as follows:

$$\mathcal{L}_{CP} = ||\mu_S - \mu_T||^2 \tag{8}$$

Enforcing the Same Centre Rule brings three-fold advantages: Firstly, the graph misalignment caused by symmetry can be prevented; Then, it boosts data participation during the intrusion knowledge transfer to fully excavate the potentials in all data, irrespective of whether a PL is assigned or not; Finally, despite its computational simplicity, it can boost the intrusion detection performance as indicated by the experiments.

Vertex-level Alignment via Semantic Preservation The above three steps focus on the overall graph level, in this step, we shift our focus to the vertex granularity. Each vertex in the domain graph represents an intrusion category centroid, during prediction, it presents a unique probabilistic category-wise correlation. Use object as an example, a laptop should be highly similar with other laptops, somewhat similar with TV screen, and very dissimilar with a bike, irrespective of its domain origin. By enforcing the corresponding vertices from both domain graphs to preserve the correlation semantic, it in turn forces vertex-level alignment between domain graphs. Specifically, for source category k, its correlation semantic is defined as follows:

$$q^{(k)} = \frac{1}{n_S^{(k)}} \sum_{x_i \in \mathcal{X}_S^{(k)}} \text{softmax}(\frac{C(f(x_i))}{T}), \tag{9}$$

where $\mathcal{X}_S^{(i)}$ denotes category i source instances, C and f denote the shared classifier and the feature projector, respectively, T is a temperature hyperparameter that controls the correlation semantic smoothness. Similarly, the correlation semantic of each labelled target instance is defined as follows:

$$p_i = \text{softmax}(C(f(x_i))), x_i \in \mathcal{X}_{TL} \tag{10}$$

To perform the semantic preservation, we minimise the vertex-level alignment loss as follows:

$$\mathcal{L}_{VS} = -\frac{1}{n_{TL}} \sum_{(x_i,y_i)\in(\mathcal{X}_{TL},\mathcal{Y}_{TL})} (q^{(y_i)})^\top \log(p_i) \tag{11}$$

Together with the supervision provided by the labelled target instances, the final vertex-level alignment loss is defined as follows:

$$\begin{aligned}\mathcal{L}_V = &\frac{1-\alpha}{n_{TL}} \sum_{(x_i,y_i)\in(\mathcal{X}_{TL},\mathcal{Y}_{TL})} \mathcal{L}_{ce}(C(f(x_i)), y_i) \\ &+ \alpha\mathcal{L}_{VS},\end{aligned} \tag{12}$$

where $\mathcal{L}_{ce}$ denotes cross entropy loss. By minimising the vertex-level alignment loss $\mathcal{L}_V$, it will enforce vertices in the same category from different graphs to align with each other. Otherwise, the correlation semantic will fail to be preserved.

Geometric Graph Alignment Theorem The GGA forms the above mechanism into a holistic framework, and can align two domain graphs with theoretical guarantee. We state the *GGA Theorem* as follows:

Theorem 1. ***GGA Theorem*** *Given graph G_S and G_T, if the Same Shape Rule, the Same Angle Rule and the Same Centre Rule hold simultaneously, then G_S and G_T must exactly align with each other.*

The proof of the GGA Theorem is as follows:

Proof. We prove the GGA Theorem by *induction with the help of contradiction*.

Case 1: Both G_S and G_T have 2 vertices. Given that the Same Angle Rule holds, both graphs must be parallel with each other. Given that the Same Shape Rule holds, it enforces the only edge in both graphs to have equal length. Therefore, it is trivial to conclude that these two graphs are the same.

Case k: Both G_S and G_T have k vertices. The aforementioned 3 rules hold and G_S and G_T are aligned. We add an additional vertex to G_S and G_T separately. Without breaking any of the aforementioned 3 rules, the new graph G'_S and G'_T also align with each other.

Proof Case k by contradiction:

Since under Case k, the prerequisite states that G_S and G_T align with each other, therefore, we simply denote both of them as G_X.

Case k1: If G_X is asymmetric regarding any line, then it is trivial that there is no possible strategy to add point differently to G_X to get G'_X and G''_X without violating the Same Centre Rule.

Case k2. If G_X is symmetric regarding a symmetric axis, to stick to the Same Shape Rule and the Same Angle Rule, the only possible strategy to add V_S^{k+1} and V_T^{k+1} is as follows: the line crossing V_S^{k+1} and V_T^{k+1} should also cross origin (Same Angle Rule) and the centre point of the symmetric axis (Same Shape Rule). However, if we add V_S^{k+1} and V_T^{k+1} differently, then they must be symmetric regarding the graph symmetric axis, which violates the Same Centre Rule for the graphs. □

Given that the GGA theorem holds, the constituting components of the GGA method can achieve a fine-grained geometric graph alignment with theoretical guarantee and benefit intrusion knowledge transfer.

C. Overall Optimisation Objective

Finally, the source labels provide supervision during the training process with supervision loss defined as follows:

$$\mathcal{L}_{SUP}(\mathcal{X}_S, \mathcal{Y}_S) = \frac{1}{n_S} \sum_{(x_i,y_i)\in(\mathcal{X}_S,\mathcal{Y}_S)} \mathcal{L}_{ce}(c(f(x_i)), y_i). \tag{13}$$

Overall, the optimisation objective of GGA is as follows:

$$\min_{C,E_S,E_T} \max_{D} (\mathcal{L}_{SUP} + \gamma\mathcal{L}_{SK} + \eta\mathcal{L}_R + \lambda\mathcal{L}_{CP} + \mathcal{L}_V), \tag{14}$$

where γ, η and λ are hyperparameters controlling the influence of loss components during optimisation. During initial training stages, both domain graphs may suffer from immature shape, therefore the γ is set to a relatively low value to emphasise other components such as vertex-wise semantic alignment, etc. As the training progresses, the γ will grow linearly to gradually emphasise the importance of shape keeping. To form the optimisation into an end-to-end procedure, we follow

TABLE I
INTRUSION DETECTION ACCURACY UNDER DEFAULT DATA SCARCITY RATIO $n_{TL} : n_{TU} = 1 : 50$

S → T	C → B	N → B	K → B	C → G	N → G	K → G	C → W	N → W	K → W	Avg
TNT	10.71	16.42	16.40	53.85	70.07	61.37	53.41	70.07	61.36	45.96
APE	33.99	51.46	52.42	54.18	70.35	61.67	47.24	63.35	55.81	54.50
MME	34.98	51.35	52.67	54.14	70.29	61.63	50.44	65.46	57.81	55.42
CDAC	34.04	50.29	50.87	54.22	70.35	61.70	53.89	70.34	61.65	56.37
STAR	33.69	50.87	51.02	54.17	70.32	61.65	53.72	70.23	61.83	56.39
DDAS	35.13	52.15	52.01	53.90	72.52	60.11	54.35	71.54	63.48	57.24
STN	35.93	51.94	53.48	57.44	69.85	64.84	54.28	71.28	63.24	58.03
DDAC	35.20	52.54	53.55	55.65	71.92	59.36	56.96	71.80	65.34	58.04
WCGN	42.79	60.10	63.80	58.26	76.62	66.83	56.82	72.39	64.41	62.45
GGA (Ours)	**48.59**	**68.99**	**77.26**	**59.43**	**79.18**	**68.25**	**58.71**	**72.75**	**66.22**	**66.60**

TABLE II
INTRUSION DETECTION ACCURACY UNDER VARIED $n_{TL} : n_{TU}$ RATIOS

S → T	N → B			K → G			C → W			Overall	1 : 100
$n_{TL} : n_{TU}$	1 : 10	1 : 50	1 : 100	1 : 10	1 : 50	1 : 100	1 : 10	1 : 50	1 : 100	Avg	Avg
TNT	18.86	16.42	16.38	61.40	61.37	61.30	53.44	53.41	53.31	43.99	43.66
APE	51.84	51.46	47.14	61.70	61.67	61.50	53.75	47.24	31.76	52.01	46.80
MME	52.84	51.35	47.01	61.65	61.63	45.23	53.36	50.44	50.07	52.62	47.44
CDAC	50.70	50.29	50.03	61.73	61.70	61.32	54.24	53.89	53.64	55.28	55.00
STAR	51.96	50.87	50.75	61.70	61.65	61.40	53.75	53.72	53.70	55.50	55.28
DDAS	52.50	52.15	51.64	60.37	60.11	59.70	54.56	54.35	53.71	55.45	55.02
STN	53.71	51.94	51.65	65.15	64.84	61.03	58.68	54.28	54.05	57.26	55.58
DDAC	53.16	52.54	52.25	59.93	59.36	58.61	58.17	56.96	56.60	56.40	55.82
WCGN	61.98	60.10	58.47	67.09	66.83	65.92	58.37	56.82	55.17	61.19	59.85
GGA (Ours)	**72.87**	**68.99**	**68.26**	**68.73**	**68.25**	**67.79**	**59.04**	**58.71**	**57.30**	**65.55**	**64.45**

[24] to apply the gradient reversal layer for the discriminator and optimise the model using Adam gradient descent. Upon the equilibrium of the above minimax game is reached, the network training concludes, and the domain graphs can be aligned in a fine-grained manner, which facilitates more accurate intrusion detection for the target IoT domain.

V. EXPERIMENT

A. Experimental Datasets

During experiments, we utilise 5 representative and comprehensive intrusion detection datasets, which include 3 network intrusion datasets: NSL-KDD, UNSW-NB15 and CICIDS2017, and 2 IoT intrusion datasets: UNSW-BOTIOT and UNSW-TONIOT.

Network Intrusion Dataset: NSL-KDD The NSL-KDD dataset [25] was released in 2009, which addressed issues of the prior dataset KDD CUP99 [26] such as having lots of redundant records. The NSL-KDD dataset contains benign traffic with 4 types of intrusions, such as probing attack, denial of service (DoS) attack, etc. Follow [13], we reasonably utilise 20% of the dataset. Each traffic record in the dataset is represented using 41 features. We follow Harb [27] to use the top 31 most informative features as the feature representation. The dataset is denoted as K.

Network Intrusion Dataset: UNSW-NB15 The UNSW-NB15 dataset [28] was created by UNSW in 2015 using the IXIA PerfectStorm tool, which aimed to address the data quality issue and out-of-date incomprehensive network flow issue observed in previous datasets. The dataset contains 10 traffic categories, including normal traffic, DoS attacks, reconnaissance attacks, etc. We utilise 2700 traffic records, which follows the dataset magnitude in [29]. The dataset is represented using 49 features, we perform preprocessing to remove 4 features having value 0 for nearly all records. The dataset is denoted as N.

Network Intrusion Dataset: CICIDS2017 The CICIDS2017 dataset [30] was released in 2017, which served as one of the most up-to-date network intrusion datasets with modern attack patterns. The dataset has 7 types of intrusions with benign traffic, represented in 77-dimensional features. We utilise the 20% portion of the dataset provided by the dataset creator to perform the model training and testing. During preprocessing, we perform data deduplication and categorical-numerical entries conversion. Following Stiawan [31], we use features with top 40 information gain as the feature space of the dataset and denote the dataset as C.

IoT Intrusion Dataset: UNSW-BOTIOT The UNSW-BOTIOT dataset [32] was released in 2017 by UNSW, which presented up-to-date modern attack scenarios captured based on a realistic testbed environment. The testbed environment deployed IoT devices such as weather station, smart fridge, etc., and utilised MQTT protocol, a lightweight IoT communication protocol commonly used in realistic IoT scenarios. The dataset quality has been carefully addressed, and the attack diversity has been improved. The dataset contains 4 categories including normal traffic, DoS attacks, information theft attacks, etc. Following [29], we utilise 10000 data records. The original dataset utilises a 46-dimensional feature

TABLE III
INTRUSION DETECTION PERFORMANCE USING VARIOUS EVALUATION METRICS UNDER DEFAULT DATA SCARCITY

S → T	N → G				K → B			
Metrics	P	R	F	A	P	R	F	A
APE	0.495	0.703	0.581	0.500	0.395	0.516	0.366	0.557
CDAC	0.493	0.702	0.579	0.500	0.251	0.501	0.335	0.553
STAR	0.495	0.703	0.581	0.491	0.250	0.500	0.333	0.481
DDAS	0.700	0.718	0.688	0.696	0.390	0.522	0.376	0.674
STN	0.495	0.703	0.581	0.693	0.525	0.534	0.405	0.768
DDAC	0.749	0.712	0.713	0.700	0.393	0.540	0.407	0.769
WCGN	0.763	0.747	0.740	0.884	0.608	0.617	0.583	0.743
GGA (Ours)	**0.828**	**0.800**	**0.790**	**0.927**	**0.778**	**0.773**	**0.757**	**0.884**

space. We follow the dataset creator's advice to use top 10 most informative features as the feature space. The dataset is denoted as B.

IoT Intrusion Dataset: UNSW-TONIOT The UNSW-TONIOT dataset [33] serves as one of the latest IoT intrusion datasets [34], released in 2021. It reflects modern IoT standards, protocols, and is captured on modern testbed consists of 7 types of IoT devices, such as smart fridge, modbus sensor, GPS tracker, etc. The dataset covers 9 types of intrusions, including scanning attacks, DoS attacks, etc. Heterogeneities present between IoT devices as the features captured by each type of IoT device have their own feature dimension. Following [29], [35], we utilise around 10% of the dataset, and select the weather meter and GPS tracker as the IoT devices used during experiments, which are denoted as W and G, respectively.

Comprehensiveness of Datasets The datasets we utilised are representative and comprehensive to verify the effectiveness of the proposed method. Firstly, these datasets are widely recognised and widely adopted by the research community to testify intrusion detection effectiveness with thousands of citations. Secondly, these datasets are developed and released in recent years, some of them are release in 2021, therefore, they can reflect current intrusion trends and methods. Finally, these datasets are captured on realistic testbeds with large-scale real IoT devices, and the sufficiency of the testbed is recognised by the research community. Hence, these datasets are representative with guaranteed comprehensiveness.

B. Shared Intrusion

The network intrusion datasets and the IoT intrusion datasets can have at most 8 shared categories that can be transferred as intrusion knowledge, such as DoS attack, password attack, backdoor attack, etc. These shared intrusion categories account for 100%, 54.9%, 100%, 100% and 98.3% amount of records in NSL-KDD, UNSW-NB15, CICIDS2017, UNSW-BOTIOT and UNSW-TONIOT dataset, respectively. Therefore, transferring intrusion knowledge with wide coverage can effectively detect most modern intrusions targeting the IoT domain.

C. Implementation Details

We implement the GGA method using the PyTorch deep learning framework. Following [11], [36], feature projectors are two-layer neural networks using LeakyReLu [37] as their activation function. Both the shared classifier and the discriminator are implemented as single-layer neural networks. The hyperparameter setting is fixed during all experiments as follows: $\alpha = 0.1$, $\gamma_{min} = 0.01$, $\gamma_{max} = 0.1$, $\eta = 0.01$, $\lambda = 0.01$, $d_C = 3$, $T = 5$ and $\#neighbour = 4$. Note that γ will increase linearly from γ_{min} to γ_{max} as the training progresses. To emphasise the target data scarcity, we set $n_{TL} : n_{TU} = 1 : 50$ as the default ratio. We also conduct the parameter sensitivity analyses to verify the stable and robust performance of the GGA method. Following [29], [38], we use unlabelled target prediction accuracy as our major evaluation metric, and also use the category-weighted precision (P), recall (R), F1-score (F) and Area under the ROC Curve (A) [39], [40] to evaluate the GGA performance. Specifically, we define true positive $TP^{(k)}$ as the number of category k intrusions being corrected identified, similar for true negative $TN^{(k)}$, false positive $FP^{(k)}$ and false negative $FN^{(k)}$. The mathematical definitions of evaluation metrics are as follows:

$$Accuracy = \frac{\sum_{k=1}^{K}(TP^{(k)} + TN^{(k)})}{n_{TU}}, \tag{15}$$

$$\begin{aligned} Precision &= \sum_{k=1}^{K} \frac{|\mathcal{X}_{TU}^{(k)}|}{n_{TU}} \cdot \frac{TP^{(k)}}{TP^{(k)} + FP^{(k)}} \\ &= \sum_{k=1}^{K} \frac{|\mathcal{X}_{TU}^{(k)}|}{n_{TU}} \cdot Precision^{(k)}, \end{aligned} \tag{16}$$

$$\begin{aligned} Recall &= \sum_{k=1}^{K} \frac{|\mathcal{X}_{TU}^{(k)}|}{n_{TU}} \cdot \frac{TP^{(k)}}{TP^{(k)} + FN^{(k)}} \\ &= \sum_{k=1}^{K} \frac{|\mathcal{X}_{TU}^{(k)}|}{n_{TU}} \cdot Recall^{(k)}, \end{aligned} \tag{17}$$

$$F1 = \sum_{k=1}^{K} \frac{|\mathcal{X}_{TU}^{(k)}|}{n_{TU}} \cdot \frac{2 \cdot Precision^{(k)} \cdot Recall^{(k)}}{Precision^{(k)} + Recall^{(k)}}. \tag{18}$$

Besides, metrics A (AUC) represents the area under the ROC curve, a curve plotting the TP rate and the FP rate.

D. State-of-the-art Baselines

We utilise 9 state-of-the-art comparing methods, including TNT [18], MME [21], STN [11], APE [22], DDAS, DDAC [19], WCGN [7], [8], CDAC [10] and STAR [20]. All of them are from top-tier conferences and journals, and 8 of them

are proposed between 2019 and 2021. We summarise their differences with GGA as follows:

- From the pseudo labelling perspective, the DDAC, DDAS, WCGN and TNT utilise predicted hard pseudo label for target instances and ignore both the geometric property and neighbourhood information. On the other hand, MME, CDAC and APE involve threshold-based pseudo label strategy. However, setting thresholds properly requires expertise experience and has compromised flexibility. Both APE and STAR apply pseudo label strategy using geometric property as reference, however, the neighbourhood information is still ignored. Besides, STN utilises a soft-label strategy, lacking emphasise on confident predictions. The GGA is the only method that jointly considers both the geometric property and neighbourhood information, while avoiding hard-to-set threshold.
- From the domain alignment perspective, these baselines apply diverse techniques such as CDAC's adversarial adaptive clustering, MME's alternated conditional entropy minimisation, STN's joint distribution matching, etc. However, these methods fail to explicitly conduct domain alignment from a geometric graph perspective. To our best knowledge, there lacks a similar pure geometric-based graph baseline method, the WCGN is a comparable state-of-the-art graph method based on graph learning framework. However, since a proper graph learning requires both sufficient data and a high complexity, it is challenging under the data-scarce and computationally-constrained IoT scenario. Conversely, the GGA performs knowledge transfer via a geometric graph alignment perspective. It fills the void of previous methods, avoids heavy data dependency, and has a relatively low complexity.

Therefore, these state-of-the-art baseline methods are representative and can be used to verify the superiority of the GGA method from different perspectives.

E. Performance Evaluation

Performance Analysis under Default Data Scarcity Ratio: We analyse the performance against state-of-the-art counterparts under the default target data scarcity ratio. As indicated in Table. I, the GGA clearly outperforms all comparing methods by at least 4.2%. The best-performed comparing method WCGN utilises graph learning framework, however, it does not perform well under the data-scarce IoT scenario, its PL assignment strategy also lacks consideration of geometric properties and assigns PL in an isolated manner. Hence, it is natural to observe a performance boost achieved by the GGA.

Performance Analysis under Diverse Data Scarcity Ratios: To verify the effectiveness and robustness of the GGA under varied target data scarcities, we vary the data scarcity ratio $n_{TL} : n_{TU}$ between $1:10$ and $1:100$. Following [5], [11], [36], the $1:100$ case is enough to represent an extreme data-scarce setting. We randomly pick three tasks and present their performance in Table. II. As we can observe, the GGA achieves the best intrusion detection performance among all three tasks under all data scarcity settings. It yields

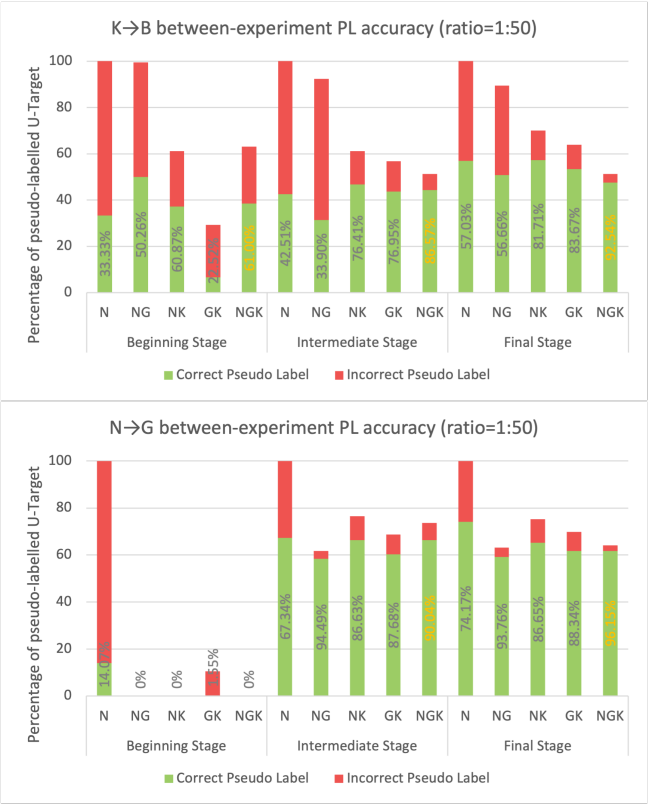


Fig. 6. Pseudo-label accuracy under default data scarcity ratio between ablated experiments

a 4.36% and 8.29% overall average performance increase compared with the best and second best-performed method WCGN and STN. Moreover, under the extreme data scarcity case, the performance boost achieved by GGA reaches 4.6% and 8.63% compared with the best and second best-performed counterpart WCGN and DDAC, and only presents a 0.87% drop compared with the performance under the $1:50$ case, which further verifies the superiority and robustness of the GGA when working under diverse data scarcity conditions.

Performance Analysis using Diverse Evaluation Metrics: To further verify the effectiveness of the GGA method using evaluation metrics other than accuracy, we randomly select 2 tasks and record the performance using another 4 evaluation metrics, and present the result in Table. III. We observe GGA achieves superior performance on all evaluation metrics. Specifically, the highest precision indicates that most of the flagged malicious decisions made by GGA are correct. The highest recall means the GGA can flag the most amount of intrusions among all malicious traffic. As a harmonic mean of precision and recall, the highest F1-score indicates that the GGA can balance properly between flagging malicious actions and avoiding triggering false alarms. Finally, the highest AUC shows the GGA can effectively distinguish malicious intrusions from normal traffic. Together, these evaluation metrics verify the effectiveness of the GGA method and its real-world usability in terms of false alarm avoidance.

Intrusion Detection Performance Summary: We verify the GGA has the best performance on all tasks when evaluated using all metrics. Therefore, it is sufficient to indicate that

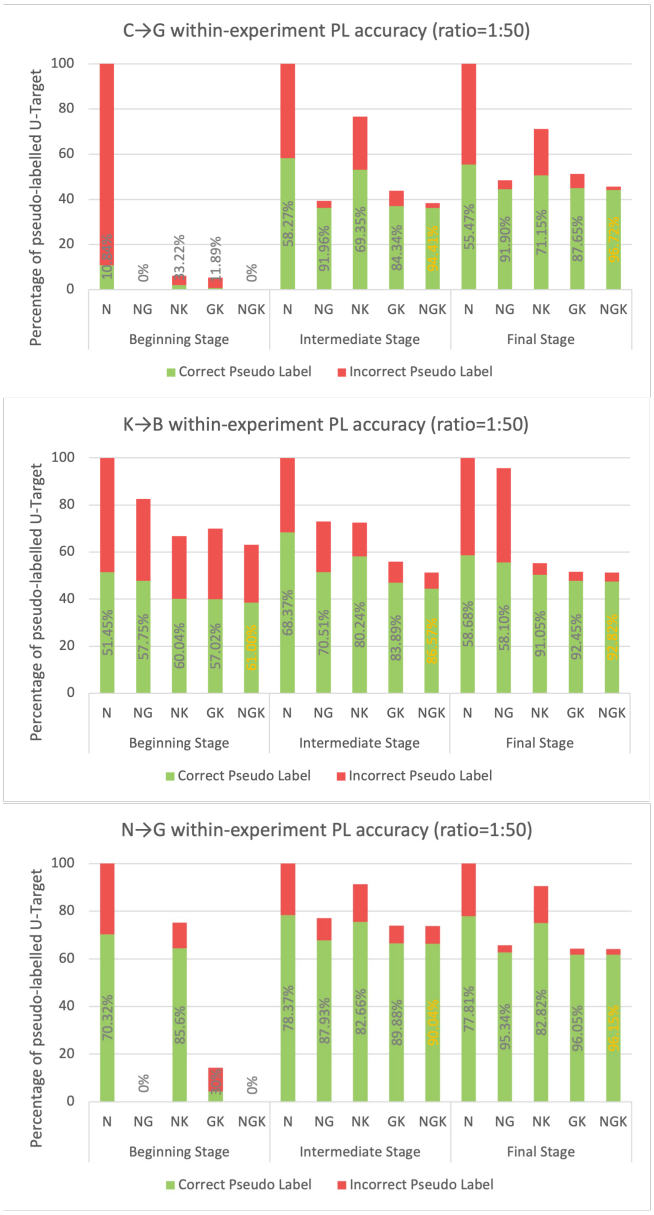


Fig. 7. Pseudo-label accuracy under default data scarcity ratio within a single experiment that utilises full PLE

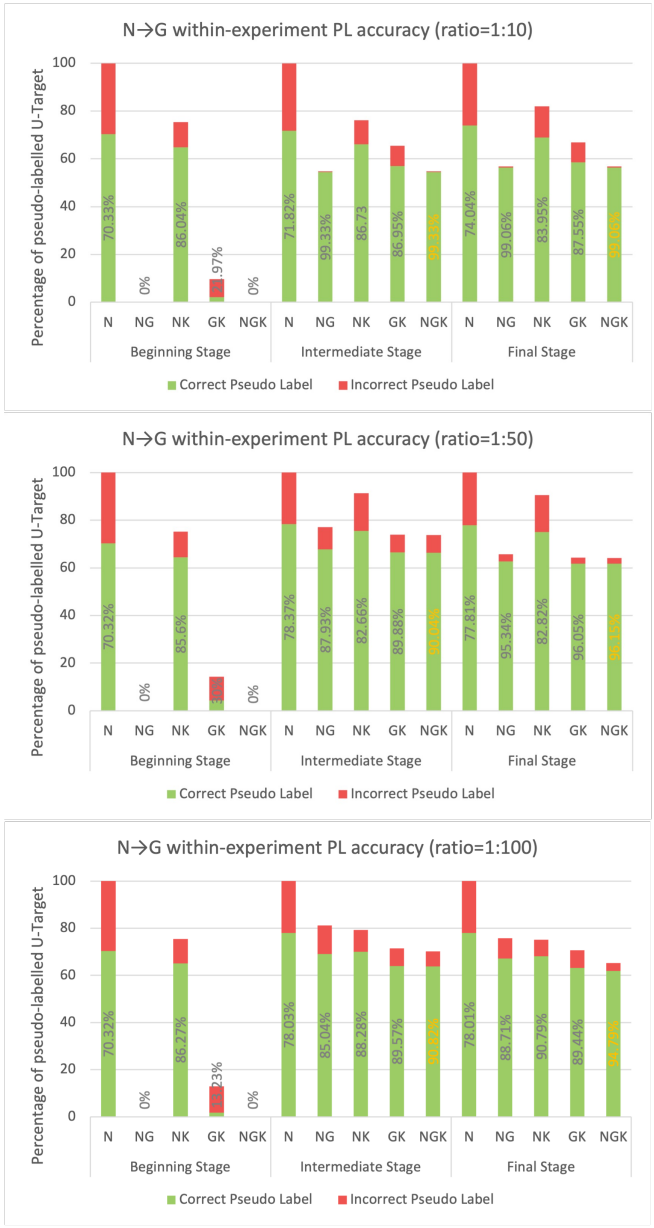


Fig. 8. Within-experiment Pseudo-label accuracy under varied data scarcity ratios

the GGA method can accurately flag malicious traffic while not causing severe false alarms. The best F1-score and AUC score performance also verify the GGA has the best ability to distinguish benign traffic and different intrusions. Having such capability promotes the real-world usefulness of GGA when performing effective intrusion detection.

F. Pseudo-label Accuracy Analysis

To justify the efficacy of the pseudo-label election (PLE) mechanism, we perform PL accuracy analysis in three ways: (a) analyse the PL accuracy between ablated experiments under default data scarcity ratio; (b) analyse the PL accuracy of different PLE configurations in a single full PLE setting under default data scarcity ratio; (c) perform (b) under varied data scarcity ratios to verify the robustness of the PLE.

The results on 2 randomly selected tasks for case (a) has been illustrated in Fig. 6. Note that N, G and K represents NN prediction vote, geometric property vote and neighbourhood vote, respectively. The height of each bar represents percentage of unlabelled target data being pseudo-labelled, and red-green colour and the value written on each bar indicates the accuracy of PL assignment. As we can observe, using the full PLE yields advantages in three-fold: (1) the full PLE achieves the highest PL accuracy during all training stages. During the beginning stage, to avoid blindly generating a vast amount of false PLs and mislead the alignment process, the full PLE can even temporarily generate no pseudo-labels, since generating PL blindly can only deteriorate the alignment process. (2) the full PLE can reach a relatively high PL accuracy around $86.6\% - 90\%$ even at the intermediate training stage, which guides the aligning process positively by fully exploiting the unlabelled target data. (3) the full PLE eventually achieves a PL accuracy around $92.5\% - 96.2\%$, which indicates the

TABLE IV
ABLATION STUDY GROUP A: INTRUSION DETECTION ACCURACY OF GGA WITH ABLATED GGA COMPONENTS

Experiment	GGA Components				Task			Avg
	Shape Keeping	Rotation Avoidance	Centre Pt Match	Vertex Semantic Match	K → B	N → G	C → W	
A_1	× ($\gamma_{min}, \gamma_{max} = 0$)	✓	✓	✓	71.83	74.2	56.64	67.56
A_2	✓	× ($\eta = 0$)	✓	✓	66.75	77.42	56.36	66.84
A_3	✓	✓	× ($\lambda = 0$)	✓	69.07	75.47	57.74	67.43
A_4	✓	✓	✓	× ($\alpha = 0$)	75.63	77.77	57.86	70.42
Full GGA	✓	✓	✓	✓	**77.26**	**79.18**	**58.71**	**71.72**

TABLE V
ABLATION STUDY GROUP B: INTRUSION DETECTION ACCURACY OF GGA WITH ABLATED PLE COMPONENTS

Experiment	Pseudo Label Election Mechanism Components			Task			Avg
	NN Label	Geometric Label	Neighbourhood Label	K → B	N → G	C → W	
B_1	✓	×	×	72.62	74.37	56.01	67.67
B_2	✓	✓	×	75.26	75.17	56.72	69.05
B_3	✓	×	✓	73.76	77.25	57.78	69.60
B_4	×	✓	✓	69.48	76.75	56.42	67.55
Full GGA	✓	✓	✓	**77.26**	**79.18**	**58.71**	**71.72**

superiority of PLE on accurate PL assignment. Although the full PLE may not generate the highest amount of PLs, however, the accuracy matters more than the amount, as indicated by the superior performance achieved by the full PLE during the ablation study.

Besides the pseudo-label accuracy analysis performed between ablated experiments, we also perform the PL accuracy analysis within a single full PLE experiment under 3 randomly picked tasks. As indicated in Fig. 7, the within-experiment performances also comply with the advantages summarised above. The full PLE can stably achieve the highest PL assignment accuracy during all training stages. During each stage, ablating any PLE constituting component will cause the PL accuracy to drop significantly. This result further verifies the usability of all components considered in the PLE.

To demonstrate the robustness of the PLE under varied data scarcities, the within-experiment PL accuracy is analysed under varied data scarcity ratios as indicated in Fig. 8. Under a relatively low data scarcity case, the PLE can achieve a 99.06% PL accuracy during the final training stage. Even under the extreme data scarcity case, the PL accuracy only drops by 4.27% compared with the $1:10$ case, which demonstrates that under varied data scarcities, the PLE can work robustly to generate accurate PL assignment and benefit positive intrusion knowledge transfer during the geometric graph alignment process.

G. Ablation Study

We further investigate the efficacy of GGA's constituting components. Ablation group A has the corresponding GGA components in Equation 14 being turned off. Ablation group B has different PLE voters being ablated. Ablation group C compares GGA with the method that uses direct vertex Euclidean distance alignment as an alternative, which is defined

TABLE VI
ABLATION STUDY GROUP C: INTRUSION DETECTION ACCURACY OF METHODS WITH DIFFERENT GRAPH ALIGNMENT MECHANISMS

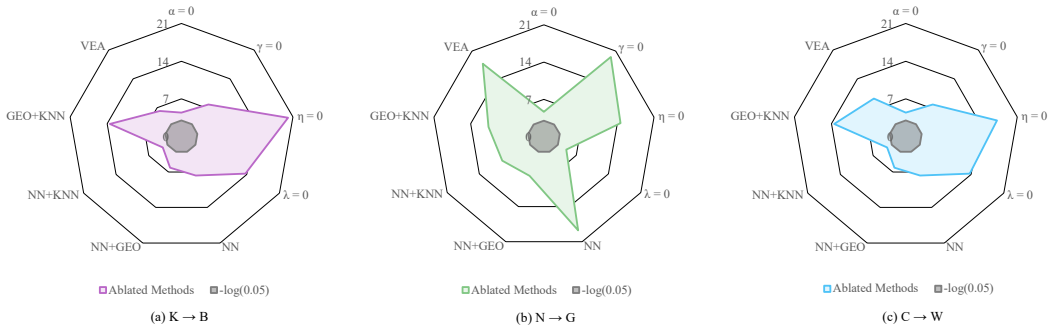
Experiment	Alignment Mechanism	Task			Avg
		K→B	N→G	C→W	
C	Vertex E-dist Alignment	74.38	74.79	57.65	68.94
Full GGA	**Geometric Graph Alignment**	**77.26**	**79.18**	**58.71**	**71.72**

as follows:

$$\begin{aligned} &\min \sum_{i}^{K} \sum_{(A,B)} ||V_A^i - V_B^i||^2, \\ &(A,B) \in \{(S, TL+PL), (S, S+TL+PL), \\ &(TL+PL, S+TL+PL)\}, \end{aligned} \tag{19}$$

where $S+TL+PL$ stands for combining instances from the source, labelled-target and pseudo-labelled target domains.

As indicated in Table. IV to Table. VI, the full GGA outperforms all its ablated counterparts by 3.4% on average, verifying positive contributions made by all constituting components towards a fine-grained geometric graph alignment. In ablation group A, the rotation avoidance mechanism is the best performance contributor with 4.9% of performance boost, followed by the centre matching, shape keeping mechanism and vertex-level semantic preservation. In ablation group B, the results verify that all three voting components are indispensable. The performance will drop by 3.3% on average without any one of them. Finally, in ablation group C, a 2.8% performance reduction is observed by the Euclidean distance-based pure vertex alignment procedure. It is natural to observe since there are huge heterogeneities between domains, therefore, pure vertex-level distance-based alignment may not be strong enough to enforce a fine-grained graph alignment, which results in degraded intrusion knowledge transfer. By



(a) K → B (b) N → G (c) C → W

Fig. 9. Significance T-tests on 3 randomly selected tasks have been performed to verify the statistical soundness of the contributions from different constituting components of the GGA. The grey shaded area denotes the significant threshold $-\log(0.05)$. Among each dimension, the wider the coverage is, the more significant the contribution is on that ablated component.

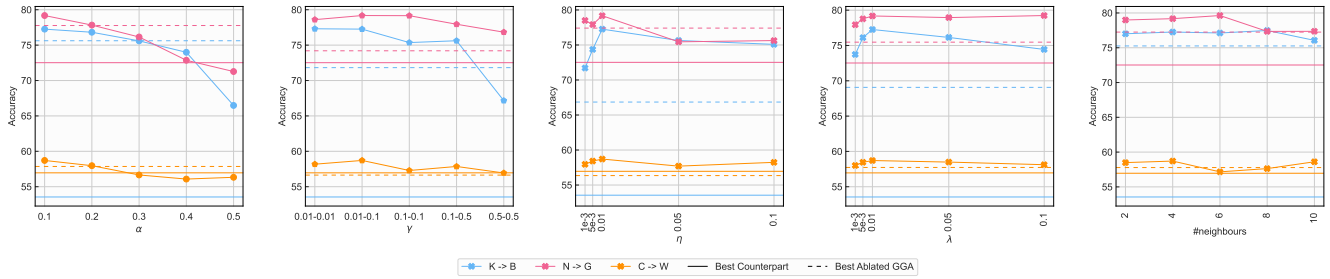


Fig. 10. The parameter sensitivity analysis of the GGA method for hyperparameters α, γ, η, λ, and $\#neighbour$ under their corresponding reasonable range. As shown in the legend, the solid and dashed horizontal lines indicate the best comparing method and the best-performed GGA ablated counterpart under each task, respectively.

jointly considering several granularities from shape keeping to vertex-level alignment, the GGA can facilitate a finer alignment and an enhanced intrusion detection performance.

H. Hypothesis Testing for Ablation Study

To verify the statistical significance of the contributions made by each constituting component, i.e., the performance boost is not observed randomly by chance, we perform significance T-test on 3 randomly selected tasks. As illustrated in Fig. 9, the grey shaded area denotes the significant threshold $-\log(0.05)$. Among each dimension, the wider the coverage is, the more significant the contribution is on that ablated component. As we can see from Fig. 9, the coloured area has wider coverage than the grey shaded area among all dimensions, which indicates that the contributions made by all constituting components have statistical soundness. Therefore, all proposed components are indispensable for GGA to achieve a fine-grained intrusion knowledge transfer via graph alignment.

I. Parameter Sensitivity

The parameter sensitivity of the GGA method has been illustrated in Fig. 10. The GGA shows relatively stable performance under these hyperparameter ranges without showing severe fluctuations. Besides, the GGA outperforms the best-performed comparing method under nearly all hyperparameter ranges. Additionally, the GGA constantly shows superior performance than its best ablated counterpart. Therefore, we verify that the GGA method is robust on varied hyperparameter settings.

Besides, during all experiments, only a fixed set of hyperparameters is used to tackle diverse data domains. The GGA can constantly show satisfying performance without the need to perform time-consuming hyperparameter resetting. Therefore, it further demonstrates the robustness of GGA on hyperparameters and its usefulness when tackling diverse intrusion data domains.

J. Computational Efficiency

TABLE VII
COMPARISON ON TRAINING TIME PER EPOCH (SECOND)

S → T	K → B	N → G	C → W	Avg
DDAC	7.54	3.86	1.96	4.45
WCGN	0.18	**0.11**	0.15	0.15
GGA (Ours)	**0.17**	**0.11**	**0.13**	**0.14**

TABLE VIII
COMPARISON ON INFERENCE TIME PER INSTANCE (MICROSECOND $= 10^{-6}$ SECOND)

S → T	K → B	N → G	C → W	Avg
DDAC	1.43	1.76	0.96	1.38
WCGN	0.20	0.17	0.19	0.19
GGA (Ours)	**0.17**	**0.14**	**0.17**	**0.16**

Finally, to verify the computational efficiency of the GGA method, we measure both the training time per epoch and inference time per instance, and make comparison between

two best-performed baseline counterparts. The results are presented in Table VII and VIII. As we can observe, the GGA achieves the best training and inference efficiency. Specifically, the GGA trains 31 times and 6.67% faster than DDAC and WCGN, respectively. Besides, the GGA also achieves the fastest inference speed, which outperforms the best-performed counterpart WCGN by 15.79%. Hence, it verifies the efficiency of the GGA, making it suitable to be used under computationally-constrained IoT scenarios.

VI. Conclusion

In this paper, we utilise the knowledge rich network intrusion domain to facilitate accurate intrusion detection for data-scarce IoT domain. We tackle this HDA problem through a geometric graph alignment approach. Firstly, a pseudo-label election mechanism is employed to exploit the unlabelled target instances, which jointly considers the network prediction, geometric property and neighbourhood information to boost the PL assignment accuracy. The PLE avoids geometrically diverged confident but wrong PLs and near-boundary ambiguous PLs. Then, both intrusion domains are formulated as graphs, with the GGA performed using four mechanisms, from general graph granularity to vertex-level alignment. Specifically, the graph shape is kept via a confused discriminator that is incapable to distinguish the origin of weighted adjacency matrices. Besides, the rotation avoidance mechanism and the centre point matching mechanism avoid graph misalignment caused by rotation and symmetry, respectively. Additionally, the vertex-level semantic is preserved to facilitate a more fine-grained graph alignment. By forming these mechanisms into a holistic whole, the GGA method can align intrusion graphs in a fine-grained manner, which benefits the intrusion knowledge transfer between domains. Comprehensive experiments demonstrate the state-of-the-art performance of the GGA method. Insight analyses also verify the usefulness of each constituting component of the GGA method.

Acknowledgment

This work is supported by the Third Xinjiang Scientific Expedition Program (Grant No.2021xjkk1300), and also in part by Science and Technology Development Fund of Macao SAR (FDCT) (1058No.0015/2019/AKP), Chinese Academy of Sciences President's International Fellowship Initiative (Grant No. 2023VTA0001).

References

[1] M. N. Bhuiyan, M. M. Rahman, M. M. Billah, and D. Saha, "Internet of things (iot): a review of its enabling technologies in healthcare applications, standards protocols, security, and market opportunities," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10 474–10 498, 2021.

[2] J. Wu, Y. Wang, B. Xie, S. Li, H. Dai, K. Ye, and C. Xu, "Joint semantic transfer network for iot intrusion detection," *IEEE Internet of Things Journal*, pp. 1–1, 2022.

[3] C. Dietz, R. L. Castro, J. Steinberger, C. Wilczak, M. Antzek, A. Sperotto, and A. Pras, "Iot-botnet detection and isolation by access routers," in *2018 9th International Conference on the Network of the Future (NOF)*. IEEE, 2018, pp. 88–95.

[4] C. D. McDermott, F. Majdani, and A. V. Petrovski, "Botnet detection in the internet of things using deep learning approaches," in *2018 international joint conference on neural networks (IJCNN)*. IEEE, 2018, pp. 1–8.

[5] J. Ning, G. Gui, Y. Wang, J. Yang, B. Adebisi, S. Ci, H. Gacanin, and F. Adachi, "Malware traffic classification using domain adaptation and ladder network for secure industrial internet of things," *IEEE Internet of Things Journal*, 2021.

[6] X. Hu, C. Zhu, G. Cheng, R. Li, H. Wu, and J. Gong, "A deep subdomain adaptation network with attention mechanism for malware variant traffic identification at an iot edge gateway," *IEEE Internet of Things Journal*, 2022.

[7] Z. Wang, Y. Luo, Z. Huang, and M. Baktashmotlagh, "Prototype-matching graph network for heterogeneous domain adaptation," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 2104–2112.

[8] L. Wang, C. Huang, W. Ma, X. Cao, and S. Vosoughi, "Graph embedding via diffusion-wavelets-based node feature distribution characterization," in *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*, ser. CIKM '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 3478–3482.

[9] Y.-T. Hsieh, S.-Y. Tao, Y.-H. H. Tsai, Y.-R. Yeh, and Y.-C. F. Wang, "Recognizing heterogeneous cross-domain data via generalized joint distribution adaptation," in *2016 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2016, pp. 1–6.

[10] J. Li, G. Li, Y. Shi, and Y. Yu, "Cross-domain adaptive clustering for semi-supervised domain adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2505–2514.

[11] Y. Yao, Y. Zhang, X. Li, and Y. Ye, "Heterogeneous domain adaptation via soft transfer network," in *Proceedings of the 27th ACM international conference on multimedia*, 2019, pp. 1578–1586.

[12] C. Jun and C. Chi, "Design of complex event-processing ids in internet of things," in *2014 sixth international conference on measuring technology and mechatronics automation*. IEEE, 2014, pp. 226–229.

[13] E. Anthi, L. Williams, M. Słowińska, G. Theodorakopoulos, and P. Burnap, "A supervised intrusion detection system for smart home iot devices," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 9042–9053, 2019.

[14] P. Shukla, "Ml-ids: A machine learning approach to detect wormhole attacks in internet of things," in *2017 Intelligent Systems Conference (IntelliSys)*. IEEE, 2017, pp. 234–240.

[15] M. Ge, X. Fu, N. Syed, Z. Baig, G. Teo, and A. Robles-Kelly, "Deep learning-based intrusion detection for iot networks," in *2019 IEEE 24th pacific rim international symposium on dependable computing (PRDC)*. IEEE, 2019, pp. 256–25 609.

[16] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, and Y. Elovici, "N-baiot—network-based detection of iot botnet attacks using deep autoencoders," *IEEE Pervasive Computing*, vol. 17, no. 3, pp. 12–22, 2018.

[17] L. Vu, Q. U. Nguyen, D. N. Nguyen, D. T. Hoang, and E. Dutkiewicz, "Deep transfer learning for iot attack detection," *IEEE Access*, vol. 8, pp. 107 335–107 344, 2020.

[18] W.-Y. Chen, T.-M. H. Hsu, Y.-H. H. Tsai, Y.-C. F. Wang, and M.-S. Chen, "Transfer neural trees for heterogeneous domain adaptation," in *European Conference on Computer Vision*. Springer, 2016, pp. 399–414.

[19] Y. Yao, Y. Zhang, X. Li, and Y. Ye, "Discriminative distribution alignment: A unified framework for heterogeneous domain adaptation," *Pattern Recognition*, vol. 101, p. 107165, 2020.

[20] A. Singh, N. Doraiswamy, S. Takamuku, M. Bhalerao, T. Dutta, S. Biswas, A. Chepuri, B. Vengatesan, and N. Natori, "Improving semi-supervised domain adaptation using effective target selection and semantics," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2709–2718.

[21] K. Saito, D. Kim, S. Sclaroff, T. Darrell, and K. Saenko, "Semi-supervised domain adaptation via minimax entropy," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 8050–8058.

[22] T. Kim and C. Kim, "Attract, perturb, and explore: Learning a feature alignment network for semi-supervised domain adaptation," in *European conference on computer vision*. Springer, 2020, pp. 591–607.

[23] M. Pilanci and E. Vural, "Domain adaptation on graphs by learning aligned graph bases," *IEEE Transactions on Knowledge and Data Engineering*, 2020.

[24] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by back-propagation," in *International conference on machine learning*. PMLR, 2015, pp. 1180–1189.

[25] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the kdd cup 99 data set," in *2009 IEEE symposium on computational intelligence for security and defense applications*. Ieee, 2009, pp. 1–6.
[26] S. Hettich, "Kdd cup 1999 data," *The UCI KDD Archive*, 1999.
[27] H. M. Harb, A. A. Zaghrot, M. A. Gomaa, and A. S. Desuky, "Selecting optimal subset of features for intrusion detection systems," 2011.
[28] N. Moustafa and J. Slay, "Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set)," in *2015 military communications and information systems conference (MilCIS)*. IEEE, 2015, pp. 1–6.
[29] O. Alkadi, N. Moustafa, B. Turnbull, and K.-K. R. Choo, "A deep blockchain framework-enabled collaborative intrusion detection for protecting iot and cloud networks," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9463–9472, 2020.
[30] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization." *ICISSp*, vol. 1, pp. 108–116, 2018.
[31] D. Stiawan, M. Y. B. Idris, A. M. Bamhdi, R. Budiarto *et al.*, "Cicids-2017 dataset feature analysis with information gain for anomaly detection," *IEEE Access*, vol. 8, pp. 132 911–132 921, 2020.
[32] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-iot dataset," *Future Generation Computer Systems*, vol. 100, pp. 779–796, 2019.
[33] T. M. Booij, I. Chiscop, E. Meeuwissen, N. Moustafa, and F. T. den Hartog, "Ton_iot: The role of heterogeneity and the need for standardization of features and attack types in iot network intrusion data sets," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 485–496, 2021.
[34] G. Abdelmoumin, D. B. Rawat, and A. Rahman, "On the performance of machine learning models for anomaly-based intelligent intrusion detection systems for the internet of things," *IEEE Internet of Things Journal*, vol. 9, no. 6, pp. 4280–4290, 2021.
[35] H. Qiu, T. Dong, T. Zhang, J. Lu, G. Memmi, and M. Qiu, "Adversarial attacks against network intrusion detection in iot systems," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10 327–10 335, 2020.
[36] S. Li, B. Xie, J. Wu, Y. Zhao, C. H. Liu, and Z. Ding, "Simultaneous semantic alignment network for heterogeneous domain adaptation," in *Proceedings of the 28th ACM international conference on multimedia*, 2020, pp. 3866–3874.
[37] A. L. Maas, A. Y. Hannun, A. Y. Ng *et al.*, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, vol. 30, no. 1. Citeseer, 2013, p. 3.
[38] J. Li, Z. Zhao, R. Li, and H. Zhang, "Ai-based two-stage intrusion detection for software defined iot networks," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2093–2102, 2018.
[39] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, "Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study," *Journal of Information Security and Applications*, vol. 50, p. 102419, 2020.
[40] S. Zavrak and M. İskefiyeli, "Anomaly-based intrusion detection from network flow features using variational autoencoder," *IEEE Access*, vol. 8, pp. 108 346–108 358, 2020.

Jiashu Wu received BSc. degree in Computer Science and Financial Mathematics & Statistics from the University of Sydney, Australia (2018), and M.IT degree in Artificial Intelligence from the University of Melbourne, Australia (2020). He is currently pursuing his Ph.D. at the University of Chinese Academy of Sciences (Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences). His research interests including big data and cloud computing.

Hao Dai received the BS and M.Sc degrees in Communication and Electronic Technology from the Wuhan University of Technology in 2015 and 2017, respectively. He is currently working toward the Ph.D. degree in the Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences. His research interests include mobile edge computing, federated learning and deep reinforcement learning.

Yang Wang received the BSc degree in applied mathematics from Ocean University of China (1989), and the MSc. and PhD. degrees in computer science from Carlton University (2001) and University of Alberta, Canada (2008), respectively. He is currently with Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, as a professor and with Xiamen University as an adjunct professor. His research interests include service and cloud computing, programming language implementation, and software engineering. He is an Alberta Industry R&D Associate (2009-2011), and a Canadian Fulbright Scholar (2014-2015).

Kejiang Ye received his BSc. and Ph.D. degree in Computer Science from Zhejiang University in 2008 and 2013 respectively. He was also a joint PhD student at University of Sydney from 2012 to 2013. After graduation, he worked as Post-Doc Researcher at Carnegie Mellon University from 2014 to 2015 and Wayne State University from 2015 to 2016. He is currently an Associate Professor at Shenzhen Institutes of Advanced Technology, Chinese Academy of Science. His research interests focus on the performance, energy, and reliability of cloud computing and network systems.

Chengzhong Xu received the Ph.D. degree from the University of Hong Kong in 1993. He is currently the Dean of Faculty of Science and Technology, University of Macau, China, and the Director of the Institute of Advanced Computing and Data Engineering, Shenzhen Institutes of Advanced Technology of Chinese Academy of Sciences.His research interest includes parallel and distributed systems, service and cloud computing, and software engineering. He has published more than 200 papers in journals and conferences. He serves on a number of journal editorial boards, including IEEE TC, IEEE TPDS, IEEE TCC, JPDC and China Science Information Sciences. He is a fellow of the IEEE.

VII. APPENDIX

Acronym Table: For better readability, we provide the following acronym table for reference. Note that all acronyms are defined in the main text as well as at the first time they are introduced.

TABLE IX
ACRONYM TABLE

Acronym	Interpretation
IID	IoT Intrusion Detection
NID	Network Intrusion Detection
GGA	Geometric Graph Alignment
DA	Domain Adaptation
NI	Network Intrusion
II	IoT Intrusion
PL(s)	Pseudo Label(s)
HDA	Heterogeneous Domain Adaptation
PLE	Pseudo Label Election
WAM(s)	Weighted Adjacency Matrix(Matrices)
NN	Neural Network

Notation Table: We provide a notation table for better readability.

TABLE X
NOTATION TABLE

Notation	Interpretation
$\mathcal{D}_*$	Domain, $* \in \{S, TL, TU\}$
x_{*_i}	The i^{th} instance of domain $*$
y_{*_i}	The intrusion class of the i^{th} instance of domain $*$
n_*	Number of instances in domain $*$
d_*	Dimension of domain $*$ instances
K	Number of intrusion categories
G_X	Domain graph of domain X
V_X	Vertices in domain graph G_X
E_X	Edges in domain graph G_X
V_X^i	Class i vertices in domain graph G_X
E_X	Feature projector for domain X
$f(x_i)$	Features projected by the feature projector
PL_G^i	The geometric label for the i^{th} unlabelled target instance
$\mathcal{X}_S^{(k)}$	Source domain instances belong to class k
$\mu_{S+TL}^{(k)}$	Centroid of the k^{th} class of source and labelled target domain instances
M_X	The weighted adjacency matrix of domain X
$\mathcal{L}_{SK}$	Shape keeping loss
$D()$	The discriminator
$\mathcal{L}_R$	Rotation avoidance loss
$\mathcal{L}_{CP}$	Centre point matching loss
$q^{(k)}$	The correlation semantic of class k for the source domain
$C()$	The shared classifier
T	Temperature hyperparameter controlling the semantic smoothness
p_i	The correlation semantic of the i^{th} labelled target instance
$\mathcal{L}_V$	Vertex-level alignment loss
α	Hyperparameter in $\mathcal{L}_V$
$\mathcal{L}_{ce}$	Cross entropy loss
$\mathcal{L}_{SUP}$	The supervision loss of source domain
γ	Hyperparameter balancing the weight of $\mathcal{L}_{SK}$
η	Hyperparameter balancing the weight of $\mathcal{L}_R$
λ	Hyperparameter balancing the weight of $\mathcal{L}_{CP}$
$TP^{(k)}$	True positive for intrusion category k