The Appendices of Paper: SFGA_Similarity-Constrained Fusion Learning for Unsupervised Anomaly Detection in Multiplex Graphs

Anonymous submission

A. Related Work in Detail

We first review the recent development of unsupervised multiplex graph learning. After that, we summarize recent advances in different anomaly detection methods for graphs.

Unsupervised Multiplex Graph Learning (UMGL). Unlike supervised approaches (Yun et al. 2019; Zhang et al. 2019; Hu et al. 2020; Wang et al. 2019b; Fu et al. 2020; Sadikaj et al. 2023), unsupervised multiplex graph learning (UMGL) addresses unsupervised tasks in multiplex graphs by leveraging both node features and graph structures (Pan and Kang 2023). In exploratory approaches within the UMGL direction, representative works such as **MvAGC** (Lin et al. 2021) and **MCGC** (Pan and Kang 2021) primarily attempt to uncover latent structural patterns in complex networks by integrating graph filtering with unsupervised learning techniques, including spectral clustering and subspace clustering.

With the advancement of deep representation learning (Khoshraftar and An 2024), researchers have integrated graph neural networks (GNNs) (Kipf and Welling 2016; Velickovic et al. 2017; Hamilton, Ying, and Leskovec 2017) with self-supervised techniques (Liu et al. 2021a) to develop UMGL methods for learning node representations, and applied the obtained representations to downstream tasks such as node classification. Additionally, **O2MAC** (Fan et al. 2020) first attempted to apply GNNs in UMGL model, this method pioneers a deep one-to-multi graph autoencoder framework for attributed multi-view graph clustering, resolving dual challenges of multi-view fusion and clustering-oriented embedding through unified optimization. Some researchers also fuse the representations of different relations by maximizing the mutual information between local and global contexts. **DMGI** (Park et al. 2020)innovatively implements an unsupervised consensus regularization framework for attributed multiplex networks, resolving representation conflicts across relation types through universal discriminators and attention-based fusion. **HDMI** (Jing, Park, and Tong 2021) proposes a high-order mutual information framework that jointly optimizes the interdependencies between node embeddings and global summaries (extrinsic signals) as well as node embeddings and attributes (intrinsic signals), while designing an attention fusion module to integrate multi-relational layer embeddings, achieving unsupervised multi-view network representation learning.

Among these efforts, some works achieve coordinated optimization and alignment of relational representations by employing multiple contrastive loss functions, while effectively preventing dimensional collapse. **MGCCN** (Liu et al. 2022b) proposes an unsupervised multi-layer graph contrastive clustering framework that captures node-neighbor relationships through an attention-based GCN encoder (self-reconstruction), aligns heterogeneous information via cross-layer contrastive learning, and iteratively refines embeddings and cluster centers with a dynamic centroid alignment strategy, achieving end-to-end joint optimization of three key objectives: topological reconstruction, cross-layer consistency, and discriminative cluster structure. **MGDCR** (Mo et al. 2023a) proposes a multi-layer graph dual-correlation reduction framework that jointly optimizes intra-layer denoising and inter-layer alignment, achieving self-supervised multi-layer graph representation learning without negative samples. **BTGF** (Qian, Li, and Kang 2024) proposes a Barlow Twins-bounded graph filter that theoretically proves the positive semi-definite inner product of input features can provide a loss upper bound, and designs a feature-structure doubly-constrained filter to achieve multi-relational graph clustering. Another line of work adopts the approach of jointly learning inter-graph consistency and complementary features to achieve comprehensive representation learning of multi-graph information. **CoCoMG** (Peng, Wang, and Zhu 2023) proposes a dual-constrained multi-graph learning framework that extracts complementary information through an MLP encoder and captures consistent information via canonical correlation analysis, addressing out-of-sample inference and noise robustness challenges in multi-graph learning. **DMG** (Mo et al. 2023b) proposes a multi-graph decoupled representation learning framework that achieves graph information decoupling and fusion through a cross-graph commonality alignment loss and an intra-graph private information purification loss.

However, despite significant progress, the application of current UMGL methods to anomaly detection in multiplex graphs remains an overlooked research area, and our method aims to develop an end-to-end approach for identifying anomalous nodes in multiplex graphs.

Anomaly Detection on Graph. In graph anomaly detection (GAD), some researchers have achieved promising results by employing GNNs in semi-supervised anomaly detection approaches (Wang et al. 2019a; Zhang et al. 2024b). Specially, **CapsGI** (Zheng et al. 2024) proposes a capsule graph information maximization framework that decouples node capsules to extract intrinsic attributes while reinforcing normal node associations through capsule graph contrastive learning, addressing the behavior-label inconsistency problem in anomaly detection. **PMP** (Zhuo et al. 2024) proposes a partitioning message passing framework that addresses label imbalance and the homophily-heterophily mixture problem in fraud detection through label-aware neighbor partitioning aggregation and node-adaptive weight generation.

Due to the frequent lack of data labels in real-world applications, GAD research has predominantly focused on unsupervised methods, gradually advancing toward medium- and large-scale datasets. The generative learning method typically utilizes reconstruction error for anomaly detection. **Dominant** (Ding et al. 2019) proposes a graph convolutional autoencoder framework that achieves unsupervised graph anomaly detection through a dual-channel reconstruction mechanism for both structure and attributes, where the anomaly score is calculated as the reconstruction error. **AnomalyDAE** (Fan, Zhang, and Li 2020) proposes a dual-channel autoencoder framework that jointly captures complex relationships between network structure and node attributes through dual-path reconstruction and cross-modal interaction mechanisms, achieving high-precision anomaly detection. **AEGIS** (Ding et al. 2021) proposes an inductive graph anomaly detection framework that captures anomaly-aware representations through graph meta-layers and enhances model generalization via adversarial generative training, enabling incremental anomaly detection without retraining. **GUIDE** (Yuan et al. 2021) proposes a higher-order structure-aware dual autoencoder framework that jointly captures anomalous behaviors in node attributes and topological patterns through network motif-based modeling of higher-order interactions and structural difference attention mechanisms. **VGOD** (Huang et al. 2023) proposes a variance-based dual-model framework that jointly optimizes a structural anomaly detection model and an attribute reconstruction model to address data leakage and detection imbalance issues in graph anomaly detection.

Contrastive learning-based methods detect graph anomalies based on the loss obtained from contrastive learning. **SL-GAD** (Zheng et al. 2021) proposes a context-subgraph-based self-supervised framework that achieves unsupervised graph anomaly detection through collaborative synergy between generative attribute reconstruction and multi-view contrastive learning modules. **AD-GCL** (Xu et al. 2025) proposes a structure-imbalance-aware graph contrastive learning framework that addresses the detection bottleneck for low-degree tail nodes through differentiated head/tail node augmentation strategies and dual-view consistency constraints, achieving robust anomaly detection. **CoLA** (Liu et al. 2021b) proposes a target node-local subgraph contrastive learning framework that achieves unsupervised attributed graph anomaly detection through multi-round neighbor subgraph sampling and a dual-channel discrimination mechanism. **Anemone** (Jin et al. 2021) proposes a multi-scale contrastive learning framework that achieves attributed graph anomaly detection through dual-granularity contrast at node-level and neighborhood-level, combined with a statistical anomaly estimator. **ADA-GAD** (He et al. 2024) proposes a two-stage anomaly-denoising autoencoder framework that addresses the anomaly overfitting and homophily trap issues in graph anomaly detection through multi-level graph augmentation pretraining and node anomaly distribution regularization.

Additionally, researchers have approached anomaly detection from a spatial distance perspective, distinguishing anomalous nodes based on differences in spatial distances. **AAGNN** (Zhou et al. 2021) proposes a neighbor-deviation-based anomaly detection framework that achieves unsupervised graph anomaly detection through subtractive aggregation characterizing node abnormality and hypersphere learning objectives. **OCGNN** (Wang et al. 2021) proposes a one-class graph neural network framework that drives GNNs to learn compact representations of normal nodes through hypersphere learning objectives, achieving topology-aware graph anomaly detection. **PREM** (Pan et al. 2023) proposes a preprocess-match dual-module framework that achieves efficient graph anomaly detection without training-time message passing through anonymized message passing for neighbor feature precomputation and a lightweight self-neighbor matching network.

In recent years, researchers have integrated diffusion models into GAD tasks, proposing numerous promising methods. **Diga** (Li et al. 2023) proposes a guided diffusion model-based subgraph recovery framework that achieves high-precision graph anomaly detection in anti-money laundering scenarios through a three-stage process of biased subgraph sampling, conditional graph diffusion and weight-sharing GNN. **GODM** (Liu et al. 2023) introduces a latent diffusion model-powered graph anomaly data augmentation framework, which synthesizes high-fidelity anomalous nodes via sequential heterogeneous information encoding, latent space diffusion, and conditional graph generation operations, effectively mitigating class imbalance challenges. **DiffGAD** (Li et al. 2024) introduces a latent diffusion-powered discriminative content distillation framework for unsupervised graph anomaly detection, employing a cascaded architecture that sequentially performs generic content preservation, shared feature construction and discriminative content separation.

However, when existing methods encounter multiplex graphs with heterogeneous edges in real-world scenarios, their performance often falls short of expectations, which once again poses challenges for research in this field.

B. Summary of SFGA

The computational process of SFGA is summarized in Algorithm 1. As we can see, during each iteration, we initially train the multiplex graph autoencoder using both the reconstruction loss $\mathcal{L}^{rec}$ and constraint loss $\mathcal{L}^{con}$ for e_i steps. During this stage, the multiplex graph autoencoder first reconstructs node original features using their fused representations across different layers, while $\mathcal{L}^{rec}$ enforces the reconstructed node features to be closed to original input. Additionally, $\mathcal{L}^{con}$ optimizes the node representations in

Algorithm 1: Our proposed SFGA

Input: Node original attributes $\mathbf{X}$ and the graph structures $\mathbf{A}^{(r)}$ of each layer in multiplex graph $\mathcal{G}$ for $\forall r \in [1, R]$.
Parameter: Non-negative parameters α and β, number of outer iterations e_o, number of inner iterations e_i, learning rate of relation separated autoencoder l_r.
Output: The anomaly scores of each node in $\mathcal{G}$.

```
Randomly initialize the parameters of all modules of this method.
for outer_iter = 1 to e_o do
  for inner_iter = 1 to e_i do
    Obtain view-specific nodes representations H^(r) by Eq. (1).
    Obtain the fused representations H^(f) by Eq. (6).
    Reconstruct X̂^(r) by decoding H^(r) by Eq. (2).
    Calculate losses:
      L^rec between X̂^(r) and X by Eq. (3).
      sim^(r)_{i|nei} for (v_i, v_j) pairs by Eq. (7).
      sim^(r)_{i|global} for (v_i, v_k) pairs by Eq. (8).
      constraint loss L^con by Eq. (9).
      objective function L by Eq. (10).
    Backpropagate L to update parameters.
  end for
end for
Calculate anomaly scores f(v_i)^con of constraint loss by Eq. (11).
Calculate anomaly scores f(v_i)^rec based on reconstruction errors by Eq. (12).
Calculate the final anomaly scores f(v_i) of nodes by Eq. (13).
```

latent space by maximizing similarity between connected node pairs while minimizing similarity between unconnected node pairs. Through this approach, the reconstruction loss $\mathcal{L}^{rec}$ and similarity constraint loss $\mathcal{L}^{con}$ mutually enhance each other's effectiveness. Then, $f(v_i)^{rec}$ identifies a subset of anomalous nodes through feature reconstruction errors, while $f(v_i)^{con}$ detects anomalies based on deviations in the constrained similarity. Finally, SFGA computes each node's anomaly score by fusing $f(v_i)^{rec}$ and $f(v_i)^{con}$.

C. Complexity Analysis

In this section, we briefly analyze the time complexity of each component in SFGA. The time complexity of the relation separated autoencoder is $O(r \cdot (N^2 \cdot h + N \cdot h \cdot c + L \cdot N \cdot h^2))$, where N is the number of nodes, d is the original node feature dimension, h is the hidden layer dimension, c is the encoder representation dimension, and r is the number of layers in the multiplex graph, L is the number of decoder layers. For the attentive inter-relation fusion mechanism, the time complexity is $O(r \cdot N^2 \cdot c)$. Since h and c are of the same order, the time complexity of the multiplex graph autoencoder can be simplified to $O(r \cdot N^2 \cdot h)$. For the feature reconstruction loss $\mathcal{L}_{rec}$, the time complexity is $O(r \cdot N \cdot d)$, while both the local and global terms of the similarity constraint loss have a time complexity of $O(r \cdot N^2 \cdot c)$. Since d and c are of the same order, the overall computational complexity of the model's loss function is $O(r \cdot N^2 \cdot c)$. Correspondingly, The complexity of feature reconstruction anomaly scoring $f(v_i)^{con}$ is $O(r \cdot N \cdot d)$, while the anomaly scoring of similarity constraint $f(v_i)^{rec}$ is $O(r \cdot N^2 \cdot c)$. To sum up, the time complexity of our proposed method can be expressed as $O(r \cdot N^2 \cdot h + L \cdot N \cdot h^2 + r \cdot N^2 \cdot c + r \cdot N \cdot d)$. Since $L \ll N$, d, c, and h are of the same order, the overall time complexity of our proposed method can be simplified to $O(r \cdot N^2 \cdot h)$.

D. Datasets Description

We perform anomaly detection experiments on four public multiplex graph datasets of varying scales across different domains, including two movie-domain multiplex graphs (Mo et al. 2023b) IMDB and Freebase with injected anomalies, along with two real-world e-commerce fraud datasets (Dou et al. 2020a; Zhang et al. 2024a) Amazon-fraud and YelpChi-fraud containing genuine anomalies.

Details of Synthetic Anomaly Injection. Specifically, the anomalies injected into IMDB and Freebase include both structural anomalies and attribute anomalies, where attribute anomalies are introduced following the prior work (Liu et al. 2021b). We use a different approach to inject structural anomalies: place n fully-connected cliques (each containing m nodes) within randomly selected subgraphs. That is to say, when placing each fully-connected clique, we first randomly select a subgraph from the multiplex graph, then choose m nodes within it to form complete connections, and exclude already marked anomalous nodes in subsequent operations. This process repeats n times, ultimately generating $m \times n$ structurally anomalous nodes in the multiplex graph.

Details of Collapsing Multiplex Graph into Homogeneous Graph. After obtaining the multiplex graph containing anomalies, we first perform element-wise summation of the adjacency matrices from all subgraphs to generate a fused matrix. Subsequently, we binarize the fused matrix by setting connections with values ≥ 1 to 1 while preserving others as 0, ultimately obtaining a multi-layer fused adjacency matrix. Through the aforementioned steps, we collapse the anomalous multiplex graph into an anomalous homogeneous graph.

Table 3 shows the data statistics including the number of nodes, edges, the dimension of the features, and the anomalies rate of the datasets, other details are discribed as follows:

- **IMDB** A relational movie network where nodes represent individual films categorized into three genres (action, comedy, drama), containing 4,780 movies in total. Its subgraphs correspond to two meta-paths: movie-actor-movie (MAM) and movie-director-movie (MDM). Features of each movie are generated from plot summaries using a bag-of-words (BoW) model, resulting in 1,232-dimensional vectors.
- **Freebase** A movie relationship dataset comprises three relation types: movie-actor-movie (MAM), movie-director-movie (MDM), and movie-writer-movie (MWM). It contains 3,492 films categorized

Datasets	**Nodes**	**Edges**	**Scale**	**Anomaly**(I/R)	**Ratio**(%)	**Relationships**	**Features**
IMDB	4,780	98,110 21,018	Small	400(I)	8.37	Movie-Actor-Movie (MAM) Movie-Director-Movie (MDM)	1,232 (BoW)
Freebase	3,492	254,702 8,404 10,706	Small	200(I)	5.73	Movie-Actor-Movie (MAM) Movie-Director-Movie (MDM) Movie-Writer-Movie (MWM)	3492 (one-hot)
Amazon-fraud	11,944	175,608 3,566,479 1,036,737	Medium	821(R)	6.87	User-Product-User (UPU) User-Star rate-User (USU) User-Text-User (UVU)	25 (handcraft)
YelpChi-fraud	45,954	49,315 573,616 3,402,743	Large	6674(R)	14.52	Review-User-Review (RUR) Review-Time-Review (RTR) Review-Star rate-Review (RSR)	32 (handcraft)

Table 3: Statistics of all datasets.

Settings	**IMDB**	**Freebase**	**Amazon-fraud**	**YelpChi-fraud**
e_o	10	10	10	2
e_i	10	10	10	10
Hidden units	128	128	128	512
D_E	128	8	8	64
l_r	1e-4	1e-3	1e-2	1e-4
L	2	3	3	4
α	0.9	0.4	0.4	0.9
β	0.5	0.3	0.7	0.2
Weight decay	1e-4	1e-4	1e-4	1e-4
w	1	1	1	1
Dropout	0.1	0.1	0.1	0.1

Table 4: Settings for the proposed SFGA.

into four genres (action, comedy, drama, documentary). Since the original dataset lacks raw features, we follow prior work (Mo et al. 2023b) to implement one-hot encoding for all node (movie) features.

- **Amazon-fraud** A dataset originates from the Amazon e-commerce platform, where nodes represent platform users containing activity data of 11,944 users. It encodes user-user interactions through three distinct adjacency matrices: user-product-user (UPU), user-star-user (USU), and user-text-user (UVU). Specifically, UPU connects users who have reviewed at least one common product, USU links users who gave identical star ratings within a one-week window, and UVU associates users with linguistically similar review styles.
- **YelpChi-fraud** A dataset derived from lodging and restaurant review data on Yelp, where all reviews are categorized as either spam or legitimate. This dataset comprises 45,954 reviews and constructs three subgraphs based on distinct inter-review relationships: review-user-review (RUR), review-time-review (RTR), and review-StarRate-review (RSR). Here, RUR connects reviews posted by the same user, RTR links reviews published within identical time windows, and RSR associates reviews sharing the same star ratings.

E. Baselines Details

We adopt the following anomaly detection methods as baselines:

- **TAM** (Qiao and Pang 2023): This approach detects anomalies by enhancing local node proximity through affinity-driven representation learning and selectively pruning heterophilous connections to eliminate structural noise.
- **CoLA** (Liu et al. 2021b): This approach detects anomalies by contrasting node-subgraph compatibility through self-supervised representation learning, leveraging multi-round sampling to quantify local structural deviations.
- **PREM** (Pan et al. 2023): This approach detects anomalies by decoupling feature preprocessing from ego-neighbor matching, which not only removes message-passing operations during training but also measures structural compatibility using anonymized similarity metrics.
- **DOMINANT** (Ding et al. 2019): This approach detects anomalies by jointly reconstructing graph topology and nodes attributes through graph convolutional autoencoders, then quantify the abnormality of nodes via structural and attribute reconstruction errors.
- **ADA-GAD** (He et al. 2024): This approach detects anomalies by first pretraining on graphs with denoised anomaly signals, then refining the anomaly score distributions. Such dual-phase approach prevents overfitting to spurious anomaly patterns through spectral-based data augmentation and entropy-driven regularization.
- **GADAM** (Chen et al. 2024): This approach introduces a neighborhood heterogeneity quantification framework that decouples interference from GNN signal propagation and amplifies anomaly discernment through context-aware message modulation.
- **AD-GCL** (Xu et al. 2025): This approach introduces a structural imbalance-aware framework that resolves performance disparity in tail anomaly detection and enhances robustness through dual adaptive neighborhood modulation.

F. Implementation Details

Hyperparameters. We performed a focused grid search on key hyperparameters that significantly impact model performance, while keeping less sensitive hyperparameters at fixed values (e.g., weight decay, self-connection weight w and dropout). The optimal hyperparameters obtained for each experimental dataset are presented in Table 4, with the grid search conducted within the following parameter space:

- Number of outer training iterations e_o: $\{2, 3, 5, 10, 15\}$
- Number of inner training iterations e_i: $\{5, 10, 15\}$
- Hidden units number of encoders: $\{16, 64, 128, 256, 512\}$
- Dimensions of node embeddings D_E: $\{8, 16, 64, 128\}$
- Learning rate of encoders l_r: $\{$1e-4, 5e-4, 1e-3, 5e-3, 1e-2$\}$
- Layer number of decoders L: $\{2, 3, 4, 5\}$
- Controlling parameter of objective function α: $\alpha \in [0, 1]$ with stride = 0.1
- Controlling parameter of anomaly scoring function β: $\beta \in [0, 1]$ with stride = 0.1

Computing environment. We implement the proposed SFGA with the following libraries: Python 3.9.21, PyTorch 1.21.1, CUDA_version 11.3, PyTorch_geometric 2.3.1 and Networkx 3.1.

Hardware configuration. For small and medium-scale datasets (e.g., IMDB, Freebase and Amazon-fraud), our experiments were conducted on a server equipped with one NVIDIA RTX 3090 GPU and one Intel i7-10700K CPU. For the large-scale dataset (e.g., YelpChi-fraud), we used a server with one H20-NVLink GPU (96GB memory) and one Intel Xeon Platinum 8457C CPU.