实验报告

本报告总结了在浙江大学软件学院数据治理与数据挖掘夏令营项目中，对三项前沿图异常检测（Graph Anomaly Detection, GAD）方法的复现与分析工作。项目聚焦于三篇代表性论文：TFGAD (WWW 2025)、GADAM (ICLR 2024) 和 UniGAD (NIPS 2024)，它们分别代表了 GAD 领域的无训练、自适应消息传递和多级统一检测三大新兴趋势。本报告首先说明各模型的核心理论与创新机制。后呈现并分析在多个基准数据集上的复现实验结果，并与原始论文报告的性能进行比对。

1 复现的论文说明

本项目挑选了三篇于 2024 至 2025 年发表在国际顶级会议上的论文。这三项工作不仅在性能上取得了突破，更重要的是，它们分别代表了三种截然不同且极具启发性的技术范式：

TFGAD (WWW 2025): 该方法挑战了深度学习在 GAD 领域的主导地位，提出一种基于奇异值分解（SVD）的无训练（training-free）方案。它追求极致的效率与简洁性，旨在证明通过经典的矩阵分解技术亦可高效解决复杂的 GAD 问题 。

GADAM (ICLR 2024): 该模型致力于解决 GNN 在 GAD 应用中的一个核心矛盾——消息传递机制的平滑效应与异常检测所需的不一致性信号之间的冲突。它通过解耦局部与全局分析，并设计自适应消息传递机制来增强检测性能，是对现有 GNN-based 方法的改良。

UniGAD (NIPS 2024): 该框架提出了一个统一的多级（节点、边、图）异常检测方案。它首次尝试同时处理不同粒度的异常并建模它们之间的内在关联。

1.1 TFGAD: 基于奇异值分解的无训练高效检测

TFGAD (Training-free Graph Anomaly Detection)核心思想在于，

许多图异常现象的本质是少数节点在属性或局部结构上偏离了大多数正常节点所共同构成的一个内在的、更简洁的模式。因此，通过经典的矩阵分解技术直接捕捉这种偏差，可能比复杂的端到端训练更为直接和高效。

TFGAD 没有采用 GNN 将节点属性与图结构耦合在一起进行信息传递，而是将两者视为独立的信源进行处理。它分别对属性矩阵和一个代表局部结构的矩阵（由邻接矩阵 A 派生）进行操作，这种解耦的处理方式简化了问题，允许为每种信息类型找到最优的线性变换 。

SVD 是 TFGAD 实现无训练的核心。对于一个给定的矩阵，SVD 能够找到其最优的低秩逼近。模型利用 SVD 来计算一个投影矩阵，该矩阵能将原始的高维特征（无论是节点属性还是结构信息）投影到一个低维的“正常子空间”中。由于正常节点的特征和结构遵循着共同的模式，它们在这个子空间中能够被很好地表示；而异常节点由于其偏离性，在投影和重构过程中会产生显著的信息损失。这个寻找最优投影矩阵的过程具有确定性的闭式解，因此无需任何迭代式训练 。

节点的最终异常分数由两部分加权组合而成，分别量化了其在属性和结构上的异常程度：

属性重构误差（Attribute Reconstruction Error）：将节点的原始属性向量投影到低维子空间后再重构回原始维度，计算重构向量与原始向量之间的欧氏距离。正常节点的属性符合低维流形假设，重构误差小；异常节点的属性则偏离此流形，导致重构误差大。

结构投影长度（Structural Projection Length）： 将节点的局部结构信息投影到其对应的低维子空间后，计算投影向量的 L2 范数。正常节点的结构模式在投影后会聚集在原点附近，而结构异常的节点则会被投影到离原点较远的位置，因此其投影向量的长度更大。

1.2 GADAM: 解耦与自适应消息传递的增强检测

GADAM (Graph Anomaly Detection with Adaptive Message Passing)解决当前基于 GNN 的异常检测方法中一个普遍存在的根本性矛盾。GNN 的核心机制——消息传递，其本质是通过聚合邻居信息来使节点表示变得更加平滑和相似。而许多经典的异常检测逻辑，其基本假设是异常节点在属性或结构上与其邻居存在显著差异。不加区分地在 GAD 任务中应用标准 GNN，其消息传递过程反而可能“稀释”或“淹没”用以识别异常的关键信号。

为了解决这一核心冲突，GADAM 设计了一个精巧的两阶段解耦框架，其策略可以概括为“先分后合，以局部指导全局” 。

第一阶段：局部不一致性挖掘（LIM）

此阶段的目标是在不受消息传递干扰的情况下，纯粹地挖掘节点的局部异常信号。为此，GADAM 完全弃用 GNN，转而采用一个简单的多层感知机（MLP）对每个节点进行独立编码。然后，通过对比学习的方式，比较一个节点自身的表示与其一阶邻居聚合后的表示之间的相似度。如果一个节点与其邻居“格格不入”，则其相似度低，局部异常分数高。由于 MLP 处理每个节点时是独立的，没有发生节点间的信息交换，因此最原始的局部不一致性信号得以完整保留。

第二阶段：全局一致性判别（GIM）

在获得可靠的局部异常分数后，第二阶段利用 GNN 强大的消息传递能力来捕捉更复杂的全局上下文信息和结构性异常模式。为了“驾驭”而不是“屈服于”消息传递，GADAM 将第一阶段产出的局部异常分数作为伪标签，选取分数最高和最低的节点分别构成高置信度的异常集和正常集。将无监督的异常检测问题转化为了一个有监督信号（伪标签）引导的二分类任务。

为了实现节点级别的自适应消息传递，GADAM 为每个节点设计了一个动态的注意力机制，来决定它应该从邻居那里聚合多少信

息。该注意力由两部分构成：

前置注意力（Pre-attention）：基于第一阶段得到的局部异常分数 的差异。如果一个节点和它的邻居在局部异常程度上相似，则它应该更多地听取邻居的意见。

后置注意力（Post-attention）：基于节点间原始特征的相似度。特征相似的邻居更有可能传递有用的信息。

这两种注意力通过一个动态变化的权重进行加权求和。在训练初期，模型更依赖于较为可靠的前置注意力；随着训练的进行，节点表示学得越来越好，模型更注意特征相似度（后置注意力）。

1.3 UniGAD: 面向多级任务的统一检测框架

在许多场景下，异常并非孤立存在于单一层面，而是多层次、相互关联的系统性问题。一个异常的银行账户（节点异常）可能会参与一系列可疑的洗钱交易（边异常），而这些账户和交易共同构成一个欺诈团伙（子图或图异常）。以往的 GAD 方法通常只关注其中一个层面，忽略了这些跨层级的关联信息。UniGAD 是第一个统一处理节点、边、图三个层面异常的框架。

UniGAD 引入了两项关键创新：

MRQSampler (最大瑞利商子图采样器)

实现“任务形式统一”的核心。MRQSampler 能将所有不同级别的异常检测任务（分析一个节点、一条边或一个完整的图）转化为单一的、标准化的“图级别”任务。具体而言，无论分析对象是什么，MRQSampler 都会为其采样一个最能体现其“异常性”的局部子图。

该采样器基于谱图理论。图中存在异常会引发谱能量分布的“右移”现象，即能量更多地集中在高频部分 。这种累积的谱能量可以通过瑞利商（Rayleigh Quotient）来量化。MRQSampler 的目标就是找到一个包含分析对象且瑞利商最大的子图，从而确保采样出的子图保留了最关键、最丰富的异常信号 。

GraphStitch Network (图拼接网络)

在通过 MRQSampler 将所有任务统一为图级别表示后，要如何共同训练这些来自不同源任务的子图表示，以实现跨任务的知识共享，同时又避免不同任务目标之间的冲突。

文章并非简单地将所有子图表示输入一个共享的网络，而是为每个任务级别（节点、边、图）保留了独立的网络分支。同时，在这些分支之间设计了可学习的“拼接单元”，通过学习到的参数来动态控制不同分支之间的信息流强度。

2 实验数据表格

实验环境，实验运行等在项目 README.MD 中体现，此处不多赘述。

对于实验数据集，本次复现实验采用了涵盖多种类型和领域的公开图数据集，以全面评估所复现模型的性能和泛化能力。具体数据集如下：

Cora, Citeseer, Pubmed, ACM,BlogCatalog, Reddit, Weibo, T-Social, Yelp,Amazon, Books, T-Finance, DGraph-Fin, Elliptic,Questions, Tolokers

需要说明的是，由于三篇原始论文并非在所有上述数据集上都进行了测试，本复现工作主要在原文涉及的数据集子集上进行评估。

2.1 TFGAD

如图 1，文章仅在项目要求的 10 个基准数据集上评估了 reddit 数据集，无法很好的衡量复现效果。

Dataset	Source	AUROC	AUPRC	Rec@K
amazon	Reproduced	0.2657	0.0428	0.0012
	Paper	*N/A*	*N/A*	*N/A*
dgraphfin	Reproduced	0.5498	0.0047	0.0043
	Paper	*N/A*	*N/A*	*N/A*
elliptic	Reproduced	0.2786	0.0141	0.0086
	Paper	*N/A*	*N/A*	*N/A*
questions	Reproduced	0.6330	0.0519	0.0719
	Paper	*N/A*	*N/A*	*N/A*
reddit	Reproduced	0.6016	0.0424	0.0410
	Paper	0.6021	0.0423	*N/A*
enron	Reproduced	0.9080	0.0645	0.2198
	Paper	*N/A*	*N/A*	*N/A*
yelp	Reproduced	0.4929	0.1461	0.1483
	Paper	*N/A*	*N/A*	*N/A*
higgs	Reproduced	0.8251	0.0018	0.0000
	Paper	*N/A*	*N/A*	*N/A*
tfinance	Reproduced	0.6361	0.0645	0.0155
	Paper	*N/A*	*N/A*	*N/A*
weibo	Reproduced	0.6430	0.3957	0.3756
	Paper	*N/A*	*N/A*	*N/A*

图 1　TFGAD 复现比较（10 个基准数据集）

故笔者在后面的复现中，使用文章评估过的数据集对复现效果进行评估，如图 2。

Dataset	Source	AUROC	AUPRC	Rec@K
ACM	Reproduced	0.9728	0.4298	0.4817
	Paper	0.9677	0.4337	*N/A*
BlogCatalog	Reproduced	0.7936	0.2718	0.3267
	Paper	0.8042	0.2750	*N/A*
Citeseer	Reproduced	0.9898	0.8396	0.7467
	Paper	0.9895	0.8364	*N/A*
Cora	Reproduced	0.9876	0.8279	0.7333
	Paper	0.9867	0.8197	*N/A*
Pubmed	Reproduced	0.9814	0.5819	0.6300
	Paper	0.9828	0.5830	*N/A*
books	Reproduced	0.5172	0.0239	0.0000
	Paper	0.7010	0.0571	*N/A*
reddit	Reproduced	0.5965	0.0422	0.0410
	Paper	0.6021	0.0423	*N/A*

图 2　TFGAD 复现比较（文章数据集）

此外，如图 3，文章未评估的数据集，笔者根据对数据集的大概了解设计了相关的超参数。出于时间等原因，并未进行网格搜索搜索等超参数调优。

Dataset	Source	k_a	k_s	η
ACM	Paper	1	60.0	10.00
BCatalog	Paper	1	220.0	0.05
Books	Paper	10	*N/A*	200.00
Citeseer	Paper	1	5.0	100.00
Cora	Paper	1	5.0	10.00
Pubmed	Paper	1	35.0	100.00
Reddit	Paper	10	5.0	500.00
Amazon	Custom	10	100.0	1.00
Questions	Custom	10	50.0	10.00
Tolokers	Custom	10	50.0	10.00
Yelp	Custom	10	100.0	1.00
dgraphfin	Custom	20	20.0	500.00
elliptic	Custom	20	20.0	500.00
tfinance	Custom	20	20.0	500.00
tsocial	Custom	10	200.0	1.00
weibo	Custom	10	50.0	10.00

图 3　TFGAD 复现超参数

2.2 GADAM

在文章的数据集上，评估结果如图 4 所示。

Dataset	Source	AUROC	AUPRC	Recall@K
ACM	Reproduced	0.9593	0.4469	0.4600
	Paper	0.9603	*N/A*	0.4590
BlogCatalog	Reproduced	0.8065	0.2966	0.3667
	Paper	0.8117	*N/A*	0.3667
Citeseer	Reproduced	0.9374	0.7029	0.6333
	Paper	0.9415	*N/A*	0.7120
Cora	Reproduced	0.9576	0.7480	0.7533
	Paper	0.9556	*N/A*	0.7299
Pubmed	Reproduced	0.9284	0.2761	0.3550
	Paper	0.9581	*N/A*	0.4620
Reddit	Reproduced	0.5792	0.0463	0.0792
	Paper	0.5809	*N/A*	0.0699
Books	Reproduced	0.5184	0.0214	0.0000
	Paper	0.5983	*N/A*	0.0143

图 4　GADAM 复现比较（文章数据集）

2.3 UniGAD

在文章的数据集上，评估结果如图 5 所示。T-Finance 数据集显存超限，未进行评估。

Dataset	Source	AUROC_Node	AUPRC_Node	Rec@K_Node	AUROC_Edge	AUPRC_Edge	Rec@K_Edge
Reddit	Reproduced	0.6664	0.0548	0.0612	0.9385	0.0529	0.068
	Paper	0.7165	0.0389	*N/A*	0.6546	0.0311	*N/A*
Weibo	Reproduced	0.8930	0.6666	0.6023	0.8890	0.5625	0.5147
	Paper	0.9902	0.9335	*N/A*	0.9913	0.8647	*N/A*
Amazon	Reproduced	0.8245	0.3207	0.4012	0.9537	0.0939	0.1482
	Paper	0.8292	0.1086	*N/A*	0.8004	0.2807	*N/A*
Tolokers	Reproduced	0.7583	0.4286	0.4275	0.7488	0.2688	0.2997
	Paper	0.7726	0.5150	*N/A*	0.7289	0.5426	*N/A*
Questions	Reproduced	0.6861	0.0843	0.1233	0.6082	0.0425	0.0838
	Paper	0.7392	0.1244	*N/A*	0.7472	0.1389	*N/A*
Yelp	Reproduced	0.6196	0.2709	0.2905	0.7112	0.1837	0.2780
	Paper	0.6322	0.2200	*N/A*	0.6174	0.3599	*N/A*

图 5　UniGAD 复现比较（文章数据集）

3 结果分析

3.1 TFGAD

如图 1 与图 2，除 books 数据集外，复现结果均与论文结果相近，TFGAD 的方法相对简单，由于模型不涉及随机参数初始化、随机数据增强或迭代优化过程，其计算过程是确定性的。

对于 books 数据集，笔者猜测所用数据集与文章所用数据集版本、内容等有区别，GADAM 的复现中，books 数据集表现也不佳。

3.2 GADAM

如图 4，绝大多数数据集的复现结果均与论文结果相近。

在 Pubmed 和 books 这两个数据集上出现了一定差距差距，可能是数据集不同的原因，也可能是文章中提到的超参数敏感性问题。

文章提到，在某些数据集上，作者使用了特定的伪标签选取比例（异常 1%，正常 50%），但笔者切换为该比例后，效果仍不佳。可能因为对超参数和初始化更敏感导致微小的参数变动或不同的随机种子都可能导致结果产生较大波动。

3.3 UniGAD

如图 5，在节点异常检测上的核心指标与论文报告的结果非常接近。但边检测任务的指标与论文差距较大。

原因可能在于边标签的生成策略不同。论文中明确指出，边的异常标签是基于其端点节点的“异常概率”通过平均值公式生成的。但，笔者使用的 GADBench 原始数据集仅提供非 0 即 1 的二进制标签，故笔者采用一种简化方法，只有当两个连接节点都为异常时，边才被标记为异常，来进行复现。这种简化实现虽然逻辑清晰，但产生了数量极少且信号单一的异常边样本，导致差距。