

Copyright Undertaking

This thesis is protected by copyright, with all rights reserved.

By reading and using the thesis, the reader understands and agrees to the following terms:

1. The reader will abide by the rules and legal ordinances governing copyright regarding the use of the thesis.
2. The reader will use the thesis for the purpose of research or private study only and not for distribution or further reproduction or any other purpose.
3. The reader agrees to indemnify and hold the University harmless from and against any loss, damage, cost, liability or expenses arising from copyright infringement or unauthorized usage.

IMPORTANT

If you have reasons to believe that any materials in this thesis are deemed not suitable to be distributed in this form, or a copyright owner having difficulty with the material being included in our database, please contact lbsys@polyu.edu.hk providing details. The Library will look into your claim and consider taking remedial action upon receipt of the written requests.

ENHANCING STABILITY OF NODE LABELING IN DIRECTED GRAPHS VIA STRONG-TIES

YU BU

PhD

The Hong Kong Polytechnic University

2025

The Hong Kong Polytechnic University
Department of Computing

Enhancing Stability of Node Labeling in Directed Graphs via
Strong-Ties

Yu Bu

A thesis submitted in partial fulfillment of the requirements for
the degree of Doctor of Philosophy
May 2025

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgment has been made in the text.

Signature: _____

Name of Student: Yu Bu

Abstract

Ensuring node labeling stability is critical for graph-based learning systems, directly impacting trust evaluation, node classification, and structure-aware inference in complex relational domains. Traditional manual annotation is costly and inconsistent, necessitating automated alternatives such as Large Language Models (LLMs). However, while LLM-assisted annotation improves efficiency, the reliability of node labeling remains a concern due to structural biases, particularly in unsupervised and semi-supervised learning settings. This dissertation systematically investigates the role of strong-tie structures in stabilizing label predictions across different node labeling paradigms, examining their influence on unsupervised trust prediction, defense against poisoning attacks in graph neural networks (GNNs), and LLM-based annotation.

First, we analyze how strong-tie structures influence unsupervised trust prediction in decentralized systems and financial networks. Our study reveals that trust annotation propagates preferentially along strong ties, making it susceptible to targeted adversarial manipulations. By strategically modifying a minimal number of edges, an attacker can significantly alter trust/untrust label assignments, exposing vulnerabilities in existing trust prediction frameworks. Our analysis reveals that strong-tie structures are preferentially exploited by adversarial agents. Understanding these patterns provides deeper insight into the structural vulnerabilities of trust prediction systems and offers a basis for evaluating the robustness of different algorithms.

Next, we extend our investigation to structural poisoning attacks in semi-supervised learning. Our findings show that poisoning behaviors exhibit clear structural preferences, targeting specific strong-tie patterns to maximize their impact on label propagation. This motivates our proposed Graph Adaptive Neural Network (GANN) framework, which dynamically adjusts propagation mechanisms based on fuzzy-theoretic strong-tie graphs (STiG). By integrating adaptive trust and risk zones, GANN mitigates the spread of adversarial noise while preserving high-confidence label prediction. Through structural decomposition and adaptive validation, our approach significantly strengthens defense mechanisms in poisoned graph environments.

Finally, we propose CSA-LLM (Crowd-sourced homophily-ties-based graph annotation via large language models), which utilizes strong-tie graph structures to design LLM prompts that enhance annotation quality. By embedding structural priors in prompt engineering, CSA-LLM improves consistency in automated label generation, offering a scalable alternative to traditional manual annotation. This structured approach not only enhances annotation robustness, but also mitigates the inconsistencies introduced by structure-agnostic token generation, where LLMs generate labels based solely on textual prompts without considering graph topology.

This dissertation provides a unified perspective on the impact of strong-tie structures across node labeling paradigms, bridging trust prediction, adversarial resilience, and LLM-assisted annotation. Our findings contribute to the development of attack-aware, structure-informed annotation frameworks, with implications for applications in social network security, financial fraud detection, recommendation systems, and decentralized finance (DeFi).

Publications Arising from the Thesis

1. Yu Bu, Yulin Zhu, longling Geng, Kai Zhou, “Uncovering Strong Ties: A Study of Indirect Sybil Attack on Signed Social Network”, in *2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.
2. Yu Bu, Yulin Zhu, longling Geng, Kai Zhou, “Unleashing the power of indirect attacks against trust prediction via preferential path”, in *Knowledge and Information Systems (KAIS):1-28*.
3. Yu Bu, Yulin Zhu, Kai Zhou, “Crowdsourced Homophily Ties Based Graph Annotation Via Large Language Model”, in *2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.
4. Yu Bu, Yulin Zhu, Yuni Lai, “A Distributed Adaptive System with Strong Tie Graphs for Trust-Aware Reasoning in Adversarial Graphs”, submitted to *The 2025 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*

Acknowledgments

I would like to express my deepest gratitude to my supervisor, Kai Zhou, for his invaluable guidance, encouragement, and unwavering support throughout my research journey. I am also thankful to my co-supervisors and colleagues at The Hong Kong Polytechnic University, whose constructive feedback and stimulating discussions have enriched my understanding of the subject. Their insightful advice and expertise have been instrumental in shaping this work.

This research was partly supported by the National Science Foundation of China (No. 62106210) and the Hong Kong Research Grant Council (No. PolyU25210821). I sincerely appreciate their financial support, which enabled me to pursue this research.

I would also like to acknowledge my fellow researchers and lab members for their collaboration, technical discussions, and assistance throughout this project. Their intellectual camaraderie has made this journey more fulfilling. Additionally, I extend my appreciation to my family and friends for their unwavering encouragement and patience, which kept me motivated through challenging times.

Lastly, I am grateful to the broader academic community whose research has inspired and contributed to this work. This thesis represents a step forward in the field of graph-based machine learning and security, and I hope it serves as a valuable contribution to future studies.

Table of Contents

Abstract	i
Publications Arising from the Thesis	iii
Acknowledgments	iv
List of Figures	xi
List of Tables	xiv
1 Introduction	1
1.1 Research Background	1
1.1.1 The Importance of Stable Node Labeling in Graph-Based Machine Learning	1
1.1.2 Exploiting Strong-Tie Structures to Mislead Node Labeling in Trust Prediction	2
1.1.3 Adversarial Robustness in Node Labeling via Strong-Tie Adaptive Propagation	4

1.1.4	Structurally-Informed Prompting: Leveraging Strong-Ties for LLM-Guided Labeling	5
1.2	Research Objectives	5
1.3	Thesis Organization	8
2	Literature Review	10
2.1	Trust Prediction in Signed Graphs: Structural Risks and Attack Strategies	10
2.1.1	Unsupervised Trust Prediction and Structural Vulnerabilities in Financial Networks	10
2.1.2	Adversarial Vulnerabilities in Trust Modeling with Signed Directed Graphs	11
2.1.3	Computational Trust Metrics and the Sensitivity to Structural Bias	12
2.1.4	Exploiting Structural Dependencies in Adversarial Attacks on Signed Graphs	13
2.2	Semi-Supervised Node Labeling in Graphs: Vulnerabilities and Defense Strategies	14
2.3	LLM-Assisted Node Annotation in Graphs: Challenges and Structural Solutions	17
2.4	Research Challenges and Opportunities	19
3	Strong-Tie Exploitation in Node Labeling: A Sybil Attack Perspective on Trust Prediction	22
3.1	Motivation	22

3.2	Preliminaries	26
3.2.1	The FGA Trust Model: Fairness-Goodness Propagation in Signed Networks	27
3.2.2	The REV2 Framework: Structured Review Modeling for Trust Inference	29
3.2.3	Indirect Sybil Attacks: Structural Exploitation for Trust Manipulation	30
3.2.4	Graph Structural Attacks and Their Influence on Node Labeling Stability	30
3.3	Problem Statements	31
3.3.1	Threat Model in Trust Manipulation	31
3.3.2	Formal Problem Formulation	32
3.3.3	Vicinage Attack Implementation Pipeline	36
3.4	Adversarial Preference over Strong-Tie Paths	38
3.4.1	Defining and Classifying Tie Structures	38
3.4.2	Axiomatic and Structural Analysis of Tie Structures in FGA	42
3.4.3	Empirical Analysis of Tie-Driven Attack Preferences in REV2	47
3.5	Adversarial Optimization Framework	50
3.5.1	Unified Adversarial Optimization Formulation	50
3.5.2	From Discrete to Continuous Graph Perturbation in FGA	54
3.5.3	From Discrete to Continuous Ranking Manipulation in REV2	60
3.6	Evaluation	65
3.6.1	Dataset Description for Weighted Directed Signed Trust Networks	65

3.6.2	Baseline Attack Strategies for Trust Manipulation in Signed Graphs	66
3.6.3	Evaluating Strong Tie Effectiveness and Prerequisite Constraints in FGA	68
3.6.4	Empirical Evaluation and Preferential Path Discovery in REV2	72
3.7	Application Scenarios of Trust Manipulation via Structural Preferences	78
3.7.1	Assessing Structural Vulnerabilities in Trust Systems	79
3.7.2	Discovering Trust Chains in Graph-Based Systems	80
3.8	Challenges and Future Work in Structural Trust Manipulation	81
3.9	Summary	82
4	Adversarial Robustness in Node Labeling via Strong-Tie Adaptive Defense	84
4.1	Motivation	84
4.2	Preliminary	87
4.3	Graph Adaptive Neural Network (GANN)	90
4.3.1	Adaptive Propagation Strategy in GANN	90
4.3.2	Designing of GANN's Adaptive Classifier	92
4.4	Defender-Centric Adversarial Simulation	92
4.5	Tie-Sensitive Robustness Evaluation Under Adversarial Perturbation	94
4.5.1	Mathematical Formulation of Simulated Attacks	94
4.5.2	Selection of Test Nodes Based on Tie-Sensitive Zones	95

4.5.3	Fuzzy-Theoretic Evaluation of Trust Confidence Across Node Types	96
4.6	Formal Analysis of Trust Zone Confidence	97
4.6.1	Granular Analysis of Trust Zones	99
4.6.2	Theoretical Guarantees of Trust Zone Confidence	100
4.7	Structure-Guided Adaptive Defense for Trust-Aware Poisoning Mitigation	103
4.7.1	Theoretical Deviation Analysis under Adaptive Trust Propagation	104
4.7.2	Adaptive Mitigation via Trust Zone–Attack Preference Alignment	108
4.8	Experimental Results	112
4.9	Summary	114
5	Leveraging Strong-Tie Structures for LLM-Driven Annotation and GNN Label Stability	115
5.1	Motivation	115
5.2	Preliminaries	117
5.2.1	Structural Neighbor Typologies in Directed Graphs	119
5.3	Proposed Method	120
5.3.1	Strong-Tie-Guided Prompt Design for Crowdsourced LLM Annotation	120
5.3.2	Two-Stage Information Bottleneck for Active Node Selection .	124
5.4	Strong-Tie-Guided Homophily Propagation: A Theoretical Perspective	127

5.4.1	Transition Matrix and Label Transition Probabilities	127
5.4.2	Homophily Label Transition under 1-Hop Strong-Tie	129
5.4.3	Homophily Label Transition under 2-Hop Strong-Tie	131
5.4.4	Homophily Label Transition under Multi-Hop Strong-Tie	132
5.5	Evaluation	134
5.6	Summary	135
6	Conclusion	136
	References	140

List of Figures

1.1	Illustration of the interactions among human annotation, LLM-enhanced annotation, label prediction, adversarial threats, and defense mechanisms in graph-based learning. The diagram highlights the sources of instability and the need for robust, stable label prediction.	3
3.1	FGA as a trust/reputation model in signed graphs, demonstrating attack vulnerabilities.	28
3.2	Indirect Sybil attack on FGA: visualization of attacker, intermediary, and target nodes.	31
3.3	Concepts of perturbation space, strong ties, weak ties, and polymorphic strong ties.	39
3.4	Representation of weak ties, strong ties, and polymorphic strong ties in FGA.	43
3.5	Strength analysis of strong ties in trust networks.	44
3.6	General architecture of the proposed method across trust models. . .	50
3.7	Overall architecture of vicinage-attack on FGA	57
3.8	Performance comparison between <i>vicinage</i> -attack and baseline methods.	68
3.9	Comparative analysis of attack impacts on strong and weak ties. . . .	68

3.10	Fairness and goodness distribution in FGA alongside edge weight analysis.	70
3.11	Distribution of fairness scores within the REV2 framework.	70
3.12	Joint distribution analysis of fairness and degree in REV2.	71
3.13	Performance evaluation of vicinage-attack on bitcoin-otc dataset. . . .	72
3.14	Baseline comparison of vicinage-attack on bitcoin-alpha dataset. . . .	73
3.15	Comparative study of vicinage-attack across bitcoin-alpha and bitcoin-otc datasets.	73
3.16	Preferential path analysis over 2-hop connections in bitcoin-otc. . . .	74
3.17	Preferential path analysis over 2-hop cases in bitcoin-alpha.	75
3.18	1-hop preferential path analysis on bitcoin-otc dataset.	75
3.19	1-hop preferential path analysis on bitcoin-alpha dataset.	76
3.20	Supplementary 2-hop preferential path analysis for bitcoin-alpha using constructed perturbation spaces.	77
4.1	Role of GANN in secure graph queries under adversarial settings. . .	85
4.2	(a) Impact of poisoning attacks on Toy-1 & Toy-2. (b) Proposed method's response to attacks. (c) Trust confidence distribution in trust and risk zones. (d) Previous research on strong vs. weak ties in FGA, highlighting polymorphic strong ties.	88
4.3	Graph visualization of strong ties in GANN under varying dataset. .	93
4.4	Strong tie visualization in GANN under poisoning attacks ($\alpha = 0.0$, $\beta = 1.0$) at 5% perturbation.	94

4.5	Strong tie visualization in GANN under poisoning attacks ($\alpha = 1.0, \beta = 0.0$) at 5% perturbation.	94
4.6	Node Labeling Accuracy Trends for Zones N1–N6 under Varying Perturbation Intensities	97
4.7	Zone-Based Accuracy Evaluation: N7 Extracted as Worst-Case from N1–N6 by Removing Stable Overlapping Nodes	99
4.8	Construction of Subspaces O_E , $O_{E'}$, and O_o	101
4.9	Accuracy variation with perturbation rates for GANN, Toy-1, and Toy-2.	102
4.10	Heatmap of attacked accuracy across different (α, β) configurations at 5% perturbation.	103
4.11	Analysis of aggregation depth (k) impact on poisoned graphs.	104
5.1	Crowdsourced annotation using LLMs to support GNN-based Node Classification.	118
5.2	Performance evaluation of LLM as an annotation tool.	120
5.3	Clustering of nodes based on C-Density metric.	124
5.4	Accuracy trends with respect to C-Density distance and confidence score.	124
5.5	Training vs. testing accuracy for GCN under CSA-LLM framework. .	132

List of Tables

3.1	Structural Properties of Multi-Hop Tie Structures	41
3.2	Structural patterns of indirect attack ties within FGA.	41
3.3	Compliance with Status Theory in Trust Propagation on REV2. This table shows whether different tie structures conform to status theory under the fairness metric in REV2. A checkmark (✓) indicates compliance. Incorporating status theory helps guide the construction of more effective perturbation spaces.	48
3.4	Empirical Analysis of Tie-Driven Attack Preference in REV2. ♥ indicates strong adversarial preference under trust promotion, i.e., these paths are particularly conducive to attack effectiveness.	49
3.5	Impact of edge count differences between the original graph G and perturbation spaces, illustrating potential model distortion.	59
4.1	Test accuracy (%) under non-adaptive adversarial attack conditions on Cora.	106
4.2	Test accuracy (%) under non-adaptive adversarial attack conditions on Citeseer.	109
4.3	Test accuracy (%) under non-adaptive adversarial attack conditions on Pubmed.	110

4.4	Dataset characteristics, including statistics for Citeseer, Cora, Pol- blogs, and PubMed.	111
4.5	Test accuracy (%) under poisoning attacks, comparing GANN against Dir-GNN and Magnet under adaptive adversarial settings.	112
5.1	Example of a complete zero-shot annotation prompt for the CORA dataset [1]	123
5.2	Comparative analysis of model performance across baselines and filter- ing techniques on Cora and Citeseer datasets.	129
5.3	Hyperparameters and Test Accuracy for Cora and Citeseer Datasets .	133

Chapter 1

Introduction

1.1 Research Background

1.1.1 The Importance of Stable Node Labeling in Graph-Based Machine Learning

Graph-structured learning systems are at a critical juncture where the interplay between structural learning biases, adversarial robustness, and annotation scarcity must be systematically addressed. The ability of machine learning models to infer high-level insights from complex relational data is fundamental to applications such as social network analysis [2], molecular interaction modeling [3], recommendation systems, and fraud detection. However, a core challenge stems from the reliance of Graph Neural Networks (GNNs) on labeled data, which is expensive to obtain and prone to inconsistencies. This scarcity of labels has led to the development of automated annotation strategies and unsupervised learning methods to infer labels from graph structures, often relying on heuristic or structural assumptions.

Compounding these issues, graph-based learning models are increasingly vulnerable to

adversarial perturbations as they scale in complexity. Attackers can exploit structural biases or inconsistencies in the annotation pipeline to launch targeted manipulations that distort model predictions. This necessitates the development of structurally-aware defense mechanisms that enhance robustness in the presence of adversarial threats, while ensuring stability and consistency in the annotation process.

To address the annotation bottleneck, recent approaches have incorporated Large Language Models (LLMs) [4, 5] to generate pseudo-labels based on semantic and structural cues. Although LLM-based annotation offers a scalable alternative to manual labeling, node labeling under this paradigm faces new challenges: structural biases inherent in LLMs may lead to misalignment with the graph topology, and structure-agnostic, context-insensitive generation can introduce inconsistencies in label assignment [6, 7].

To better illustrate the intricate interplay between label prediction biases, adversarial threats, defense mechanisms, and LLM-enhanced annotation, we present a conceptual framework (Figure 1.1). This diagram illustrates the interdependence among machine learning-based label prediction, human annotation, and LLM-driven annotation, and shows the adversarial threats that emerge at various stages of the annotation and prediction pipeline. Furthermore, the diagram emphasizes the importance of incorporating defense mechanisms to mitigate these threats, thereby reinforcing stable and attack-resilient label propagation in graph-based systems.

1.1.2 Exploiting Strong-Tie Structures to Mislead Node Labeling in Trust Prediction

Strong-tie structures play a foundational role in unsupervised learning-based trust prediction, especially within dynamic financial networks such as the Bitcoin transaction ecosystem [8]. These structures significantly influence how trust scores are propagated throughout the graph, which is particularly critical in settings where la-

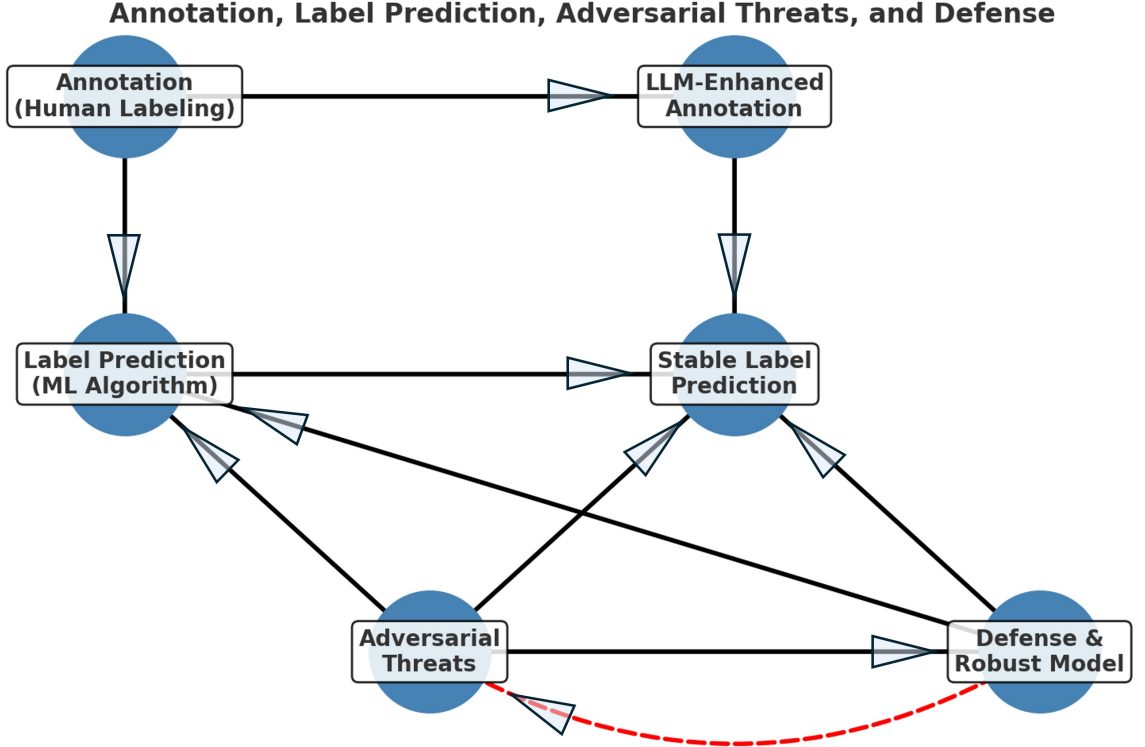


Figure 1.1: Illustration of the interactions among human annotation, LLM-enhanced annotation, label prediction, adversarial threats, and defense mechanisms in graph-based learning. The diagram highlights the sources of instability and the need for robust, stable label prediction.

beled data is unavailable. In such unsupervised trust systems, node-level credibility is inferred purely from network topology, without explicit supervision, rendering the system vulnerable to adversarial manipulation. Our study reveals that trust annotations tend to propagate preferentially along strong-tie connections, allowing adversaries to subtly manipulate a small subset of edges and still induce substantial deviations in trust score distributions.

In graph-based trust prediction, models often rely on structural properties to infer node-level credibility without labeled data. For example, in decentralized financial systems like Bitcoin, trust relationships can be represented as signed networks[9],

where edges denote trust or distrust between users. Methods such as Fairness and Goodness Algorithm (FGA) [10] use these structures to compute trust scores. However, this structural dependence makes the system vulnerable to adversarial manipulation. Even minor modifications—such as altering 2–3% of edges—can significantly distort trust estimation, enabling Sybil attacks and trust exploitation [11]. This highlights the need for structurally-aware frameworks that can maintain robust trust evaluation under adversarial conditions.

1.1.3 Adversarial Robustness in Node Labeling via Strong-Tie Adaptive Propagation

The role of strong-tie structures extends further into semi-supervised learning, where label propagation mechanisms in GNNs can be manipulated by poisoning attacks. Our research demonstrates that graph poisoning attacks exhibit structural preferences, disproportionately targeting specific strong-tie substructures to amplify their impact. This insight motivates the development of Graph Adaptive Neural Network (GANN), a framework that introduces dynamic trust and risk zones to mitigate adversarial propagation. By leveraging fuzzy-theoretic strong-tie graphs (STiG), GANN dynamically adjusts message passing mechanisms, preventing adversaries from exploiting structural vulnerabilities while preserving high-confidence annotations.

Traditional defense mechanisms such as anomaly detection [12] and graph purification—which removes structurally suspicious elements based on spectral analysis or feature inconsistency [13, 14, 15]—struggle against precision attacks that manipulate polymorphic strong-tie subgraphs, i.e., structures with both high trust influence and adversarial sensitivity [16, 17]. Most purification-based defenses are target-agnostic and thus overlook attacks that exploit intermediate nodes to form strong ties with the target, causing injected edges to resemble legitimate connections and evade removal. The need for a paradigm shift in adversarial defense is evident: rather than solely

relying on pre-defined heuristics or static filters, we must integrate robust annotation, adversarial attack surface analysis, and dynamic defense strategies to ensure the resilience of graph-based learning systems.

1.1.4 Structurally-Informed Prompting: Leveraging Strong-Ties for LLM-Guided Labeling

Beyond unsupervised and semi-supervised learning settings, recent advances suggest that structural priors—particularly strong-tie relationships—can significantly enhance LLM-assisted annotation. By integrating graph-aware prompt engineering, LLMs can be guided to generate more contextually aligned labels, reducing inconsistencies associated with naive token-based inference. Strong tie structures characterized by high-trust homophily-driven node clusters offer a natural constraint to improve the coherence of LLM-generated annotations. These insights open new opportunities for deploying structurally-informed LLM annotation in large-scale knowledge graphs, citation networks, and recommendation systems, where label quality and semantic consistency are essential.

1.2 Research Objectives

This dissertation advances a robust node labeling framework for graph-based learning by leveraging strong-tie structures. The core objective is to enhance node labeling stability across trust prediction, adversarial defense, and LLM-driven annotation. These objectives are pursued through a methodologically progressive three-phase study:

Phase I: Exploiting Strong-Tie Structures in Node Manipulation

This phase formalizes how adversaries exploit strong-tie structures to manipulate node labels in unsupervised trust prediction. Building upon graph-based trust propagation models, we demonstrate that even small-scale perturbations—affecting as little as 1.8% of edges—can substantially distort node reputation scores. We identify structurally preferred attack routes within strong-tie subgraphs and show that modifying a minimal subset of these ties can trigger cascading disruptions in fairness-goodness metrics and compromise the stability of trust-based label predictions.

Phase II: Enhancing Node Labeling Robustness via Strong-Tie-Aware Defense

Building on insights from Phase I, we extend our study to semi-supervised learning by proposing GANN (Graph Adaptive Neural Network), an adaptive defense framework designed to enhance label robustness under adversarial conditions. Rather than assuming a fixed message-passing structure, GANN incorporates strong-tie-informed propagation dynamics to counter structural poisoning attacks through three core components.

First, we introduce a fuzzy-theoretic strong-tie graph abstraction, which captures attacker behavior patterns and provides an interpretable basis for understanding how adversaries exploit structurally dominant regions. This component enables transparent analysis of attack surfaces and supports model-level reasoning about perturbation vulnerability. Second, GANN monitors dynamic trust zones and adaptively mitigates adversarial preference via label propagation adjustments. By observing propagation shifts in real-time, the model adapts to structural drift caused by poisoning attempts,

reducing the attacker’s ability to consistently exploit topological bias. Third, we formalize the concept of trust zones and risk zones to establish provable confidence in prediction reliability. These zones are used to identify structurally reliable subgraphs, enabling robust query evaluation with quantifiable safety guarantees. Overall, the GANN framework shows that adaptive propagation disrupts the attacker’s reliance on fixed strong-tie patterns for effective exploitation. As a result, GANN significantly reduces attack impact and outperforms existing baseline defenses in preserving label integrity across multiple benchmark datasets.

Phase III: Enhancing LLM Annotation via Strong-Tie Structures

We propose CSA-LLM (Crowdsourced Homophily Ties-Based Graph Annotation via Large Language Models), a framework that embeds structural priors into prompt engineering for LLM-assisted node annotation. By incorporating strong-tie context, CSA-LLM addresses structural-semantic disconnects and improves consistency in automated labeling. This approach mitigates common issues associated with structure-agnostic generation in LLMs, enhancing robustness and reducing inconsistencies in annotation outcomes by aligning generated labels with graph topology.

By integrating adversarial analysis, defense strategy, and LLM annotation, this dissertation provides a comprehensive framework for stable and robust graph-based learning systems. The proposed methods are validated across applications including social network analysis, financial fraud detection, and decentralized trust platforms.

1.3 Thesis Organization

This dissertation is organized into a three-phase investigation, each addressing key challenges in stable and robust node labeling for graph-based learning systems. This investigation integrates adversarial vulnerability analysis, adaptive defense mechanisms, and strong-tie-guided annotation. The research addresses how strong-tie structures influence trust prediction, defend against poisoning attacks, and improve LLM-assisted annotation, forming a complete attack-defense loop.

Chapter 2 presents a comprehensive literature review of graph node labeling paradigms, including human annotation, machine learning based labeling and LLM-assisted annotation. It identifies the limitations of existing methods under adversarial conditions and motivates the need for strong-tie-informed, robust labeling frameworks.

Chapter 3 investigates how strong ties can be exploited to manipulate node labeling in unsupervised trust prediction. Building upon Fairness-Goodness-Aware (FGA) model, this chapter formalizes indirect Sybil attacks and quantifies how minor structural perturbations (as low as 1.8% edge modifications) can significantly distort trust scores. A hybrid optimization framework, combining combinatorial search and continuous graph embeddings, is proposed to explore the feasibility of attack. Furthermore, this chapter empirically analyzes preferential attack paths using anomaly-based models (e.g., REV2), identifying strong-tie substructures that adversaries exploit for maximal impact.

Chapter 4 introduces Graph Adaptive Neural Network (GANN), a defense framework that builds upon the structural vulnerabilities discovered in prior chapters. GANN leverages strong-tie segmentation and fuzzy-theoretic reasoning to partition graphs into dynamic trust zones. GANN mitigates attacker preference exploitation by dynamically adjusting trust zone boundaries and enforcing provable confidence guarantees for robust node labeling. This chapter demonstrates how structural defense guided by adaptive strong ties can improve resilience in semi-supervised GNNs.

Chapter 5 presents a homophily-aware LLM annotation framework that integrates strong-tie priors into graph-aware prompt engineering. By aligning structural and semantic signals, this approach improves the reliability of LLM-generated labels and reduces annotation inconsistencies. Compared with naive prompt designs, the proposed framework offers superior performance in node classification under limited supervision.

The final chapter of this dissertation revisits the overall contributions, connecting unsupervised vulnerability, adaptive defense, and LLM annotation stability. This work ultimately addresses the annotation trilemma—balancing accuracy, security, and scalability—providing a unified strong-tie-driven solution for trustworthy graph learning in adversarial environments.

Chapter 2

Literature Review

2.1 Trust Prediction in Signed Graphs: Structural Risks and Attack Strategies

2.1.1 Unsupervised Trust Prediction and Structural Vulnerabilities in Financial Networks

Trust modeling is integral to financial transaction systems, where it plays a pivotal role in identifying anomalous behaviors and assessing user credibility [18, 19, 20, 21]. Unlike conventional machine learning tasks that rely on explicit labels, unsupervised trust annotation infers node reliability through structural patterns and historical interactions [22, 23]. In cryptocurrency networks such as Bitcoin, algorithmically computed trust scores serve as indicators of financial integrity [23, 24, 25, 26, 8], guiding decision-making in fraud detection [10], risk assessment [27, 28], and credit evaluation [29, 30].

A fundamental challenge in trust-based annotation is its vulnerability to structural adversarial attacks [31, 32, 33, 34]. In Bitcoin-like ecosystems, iterative algorithms

are commonly used to compute trust scores; however, even minor perturbations in the network structure can lead to significant deviations in the final trust estimation [35, 36]. Recent studies reveal that adversaries can systematically distort trust scores through targeted structural attacks, leading to misclassifications in financial risk assessment [16, 17, 37]. Specifically, manipulation of signed edges allows adversaries to amplify or suppress trust scores, disrupting fairness and goodness metrics in fraud detection.

These findings highlight the necessity of adversarially robust trust metrics [9, 38] to mitigate annotation errors in unsupervised financial learning. Defense strategies such as adversarial training [39, 40], homophily-aware filtering [41, 42], and trust-aware propagation models [43, 44] have been proposed to enhance resilience. However, designing scalable and interpretable trust prediction frameworks remains an ongoing research challenge [45, 23, 46].

2.1.2 Adversarial Vulnerabilities in Trust Modeling with Signed Directed Graphs

Signed directed graphs provide a richer framework for trust modeling by explicitly representing both positive (trust) and negative (distrust) relationships [38, 9]. This expressiveness is particularly important in financial systems, where adversarial interactions—such as fraud or dishonest transactions—are common [18, 19, 20, 21, 22, 23]. Bitcoin transaction networks illustrate how signed directed graphs support trust modeling, with edge signs capturing transaction credibility and directions encoding financial hierarchy [23, 24, 25, 26, 8, 10]. Signed networks provide a structured framework for multi-dimensional trust assessment, leveraging trust metrics to evaluate user reputation [10, 31].

However, signed graphs are inherently susceptible to adversarial manipulation [32, 33]. Attackers exploit trust propagation dependencies to amplify or suppress trust scores,

influencing automated risk assessments [34, 35, 36]. Targeted attacks exploiting structural biases in trust propagation can lead to disproportionate distortion in trust prediction results [37, 47]. Studies demonstrate that adversarial modifications affecting as little as 2–3% of the edges can lead to trust inversion, falsely categorizing unreliable entities as trustworthy and vice versa [10, 16, 17].

To mitigate these vulnerabilities, researchers have explored defensive mechanisms such as robust trust propagation [43, 44], adaptive graph filtering [25, 26], and fairness-aware network modeling [48, 49, 45]. Nevertheless, signed directed graphs remain challenging due to their sensitivity to edge perturbations and the difficulty of detecting adversarially induced structural shifts [23, 46].

2.1.3 Computational Trust Metrics and the Sensitivity to Structural Bias

Trust prediction systems include classical graph-based methods such as PageRank [50] and Hyperlink-Induced Topic Search (HITS) [51], along with a wide range of enhanced variants [52, 53, 54, 55, 56, 57, 9, 58, 59, 60, 61] that incorporate signed edges, interaction weights, or domain-specific constraints. These iterative methods propagate influence along directed edges, making them susceptible to structured perturbations [62, 63, 23, 46]. Even minor modifications to key structural links can significantly distort trust propagation results, leading to erroneous rankings of node credibility.

The Fairness and Goodness Algorithm (FGA) [10] estimates two trust metrics: goodness, indicating a node’s trustworthiness, and fairness, reflecting the credibility of its outgoing ratings. It has been widely applied in Bitcoin transaction networks [64, 24, 25, 23], but its reliance on structural consistency makes it vulnerable to adversarial perturbations [36, 37, 45]. The Review-to-Reviewer model (REV2) [46] extends FGA by introducing a third metric—reliability—alongside goodness and fair-

ness. Although REV2 offers improved modeling of rating dynamics, its recursive design remains sensitive to structural attacks. Despite advancements in trust propagation [58, 59, 65, 66, 67, 68], existing trust models remain vulnerable to targeted adversarial attacks [69, 33, 70, 71, 72, 73, 74]. Studies highlight that well-calibrated trust models can be bypassed using gradient-based perturbation techniques, fundamentally altering credibility assessments [75, 76, 77].

2.1.4 Exploiting Structural Dependencies in Adversarial Attacks on Signed Graphs

While signed networks enhance trust assessment by incorporating both positive and negative interactions, their structural complexity introduces new vulnerabilities [18, 19, 20, 21, 22, 23]. Attackers exploit trust propagation dependencies to distort credibility scores, undermining trust estimation accuracy and reducing the reliability of graph-based decision processes [10, 64, 24, 25, 26]. Adversarial threats in signed trust networks often arise from a variety of graph attack techniques, some of which are not limited to signed graphs.:

Adversarial threats in signed trust networks generally fall into two broad categories. The first involves structural manipulation, where attackers strategically alter the graph topology to distort trust propagation. This includes node injection, edge injection, and sign flipping—modifying positive edges to negative and vice versa—each capable of perturbing the inferred trust structure [78, 77, 79, 36, 69, 80, 81, 82]. Among these, this study primarily focuses on edge injection, as it represents a more feasible and stealthy attack strategy in real-world trust networks, where edge deletion is often impractical due to immutable records or transactional constraints. The second category includes query-based inference attacks, where adversaries interact with the trust estimation framework through surrogate queries to infer the underlying scoring mechanisms [83, 84, 85]. These attacks are typically framed within black-box

settings, in which attackers use observed responses to train surrogate models that approximate the target system, allowing them to launch targeted manipulations without direct access to model parameters.

These adversarial strategies underscore the fragility of current trust modeling architectures in signed networks [23, 46, 16, 17]. Without robust defenses, automated trust annotation and reputation-based financial systems remain susceptible to manipulation, necessitating the development of adversarially resilient trust propagation models.

2.2 Semi-Supervised Node Labeling in Graphs: Vulnerabilities and Defense Strategies

In semi-supervised learning, annotations exist for only a limited subset of nodes, making label propagation a critical component of Graph Neural Network (GNN) predictions [86, 87]. The inherent structural dependencies of graphs influence how labels spread across the network, meaning that adversarial perturbations targeting labeled nodes can significantly impact global model performance. Attackers often exploit these labeled nodes as focal points, strategically manipulating their connections to distort classification boundaries and amplify cascading errors [34, 35, 36, 37].

Recent studies indicate that adversarial attack strategies exhibit a distinct preference for modifying specific structural components, particularly in directed graphs [16, 17, 88, 89]. In graph-based node classification, label prediction relies on the propagation of information from labeled nodes to their neighbors. Consequently, structural perturbations near labeled nodes can significantly disrupt the stability of label propagation [90, 91]. This makes structurally sensitive regions particularly susceptible to adversarial manipulation. In directed graphs, we further observe that attackers tend to target specific subgraph patterns—those that maintain fixed direc-

tional relationships with labeled nodes. These subgraphs amplify the influence of small perturbations, leading to disproportionate impact on downstream predictions. The heterogeneity in neighbor influence complicates defense strategies, as certain structural motifs are inherently more vulnerable than others [92, 93].

Emerging evidence shows that adversarial agents exploit topological biases — particularly strong ties — to manipulate node labeling in adversarial settings [37, 30, 27, 51]. For example, Sybil and poisoning attacks can achieve significant disruption through minimal perturbations along strong-tie paths [94, 28], revealing a clear attacker preference for structurally influential subgraphs. This behavior undermines the stability of label propagation—a core component of node classification in semi-supervised learning, where labels are only partially observed. Since GNN-based models rely heavily on structural dependencies to propagate label information, even minor annotation inconsistencies or topological perturbations can compromise model predictions [95, 93]. Additionally, query-bounded attacks pose a significant threat by probing labeled nodes to uncover exploitable patterns in label propagation. These attacks often operate in a black-box setting using surrogate models and gradient-based inference to approximate the target model’s behavior [96, 84, 85]. This threat is particularly critical in information-centric graph systems, such as transaction networks and knowledge graphs, where trust-based inference plays a key role in retrieval, ranking, or decision-making [45, 10, 46, 97, 98]. Even without large-scale structural changes, attackers can distort trust scores over time [34, 35, 36], highlighting the need for adaptive defense mechanisms that dynamically adjust to adversarial behaviors while preserving model integrity [99, 100].

To mitigate structural vulnerabilities in graph-based learning, recent defense methods explore graph partitioning techniques that differentiate between high-risk and stable regions. In particular, labeled nodes with low confidence or high adversarial sensitivity form a structurally vulnerable subgraph, whose fixed relational patterns with the training set define a sensitive propagation interface. These interfaces often manifest as

strong-tie connections, revealing attacker preferences in perturbing label propagation. By identifying such ties, strong-tie-based trust zone partitioning provides a principled way to contain adversarial influence within bounded regions [101, 102, 103]. This partitioning framework improves robustness by anchoring label diffusion to resilient structures and reducing the propagation of noise across vulnerable subgraphs [48].

Federated learning offers an additional layer of resilience by providing a decentralized framework for label propagation [104]. By structuring trust zone partitioning within federated settings, distributed learning enables localized trust estimation without requiring direct access to the entire graph topology [105]. This approach mitigates the risk of exposing sensitive node relationships to adversarial exploitation, ensuring that trust computations remain independent across different subgraphs [106]. Moreover, federated mechanisms facilitate the identification of structurally significant boundaries, guiding the formation of trust zones that optimize the balance between security and classification accuracy [104, 105, 106].

Beyond partitioning-based defenses, graph purification techniques reinforce model robustness by filtering adversarial perturbations before they propagate through the network [13]. Spectral decomposition methods have been widely adopted to detect structural irregularities and restore the integrity of trust-based models [14]. By refining adjacency matrices and removing adversarially injected connections, these techniques limit the ability of attackers to manipulate label propagation through structural infiltration [107, 76]. This process strengthens the reliability of label propagation and preserves the interpretability of semi-supervised learning models in adversarial settings [107, 76].

These observations motivate the need for adaptive propagation strategies that can respond dynamically to structural vulnerabilities, thereby mitigating the influence of adversarial preferences and enhancing the robustness of graph-based learning systems. By incorporating strong-tie-aware propagation defenses and structurally adaptive learning frameworks, future research can develop more resilient GNN architec-

tures. These methodologies hold the potential to enhance the robustness of label propagation while ensuring stable and secure semi-supervised learning in adversarial graph environments [37].

2.3 LLM-Assisted Node Annotation in Graphs: Challenges and Structural Solutions

Accurate data annotation is fundamental to machine learning, serving as the foundation for supervised and semi-supervised learning paradigms [108, 2, 109]. In graph-based learning, high-quality labels are particularly crucial, as they serve as anchors for information propagation and greatly influence the effectiveness of downstream predictions [3, 110]. Traditional graph annotation methods have relied on either algorithmic techniques or human expertise, each presenting distinct advantages and limitations [1, 4].

While traditional algorithmic annotation methods rely on statistical heuristics or propagation-based techniques to infer labels from structural patterns [5, 111, 112], they suffer from three major limitations in real-world graph-based systems. First, these methods lack semantic or domain knowledge, making them prone to inconsistencies when structural signals alone are insufficient to distinguish between node classes [113, 114]. Second, the reliance on structural assumptions introduces vulnerabilities that can be exploited through targeted adversarial manipulation. For instance, in Bitcoin transaction networks, unsupervised models are commonly used to detect anomalous entities [23, 24, 25, 26, 8]; however, trust scores and node attributes generated by these models can be easily distorted through subtle perturbations that exploit structural weaknesses [10, 27, 28, 32, 33, 29, 30, 31]. Third, although human annotation remains the gold standard for high-quality labels [34, 35], it is prohibitively expensive and inefficient for large-scale or dynamically evolving graph structures [110].

These challenges motivate the development of more robust, scalable, and semantically grounded annotation frameworks—particularly those that integrate Large Language Models (LLMs) with graph structures to improve label quality and resilience.

To address these limitations, Large Language Models (LLMs) have recently emerged as a promising tool for automated graph annotation. LLM-assisted annotation integrates structural and textual cues to infer pseudo-labels [1, 4, 5, 111, 112, 115, 6, 7, 116, 117]. While these methods substantially reduce annotation costs [118, 119, 120, 121], concerns persist regarding bias propagation [95, 93], misalignment between inferred labels and graph topology [37], and the computational complexity of incorporating LLM outputs with graph structures. These challenges necessitate a framework that enhances annotation robustness while maintaining efficiency.

One promising direction involves leveraging strong-tie structures to refine LLM assisted annotation, ensuring structural consistency and reducing reliance on local text-based inference. In the context of LLM-driven annotation, strong ties—characterized by homophily-based structural proximity—enhance annotation accuracy by grounding prompts in structurally consistent neighborhoods, leading to higher-quality label generation. By embedding strong-tie constraints into the design of prompt input, graph annotation can align more closely with network topology, mitigating semantic drift and reducing annotation noise [122, 42].

To enhance the stability and robustness of node labeling in graph-based systems, this work integrates three complementary strategies. First, we analyze trust prediction systems and reveal that strong ties—reflecting attacker preferences—serve as critical propagation channels in unsupervised learning, providing new perspectives on adversarial vulnerabilities [123, 124]. Second, we propose an adaptive defense mechanism that dynamically adjusts propagation behavior to disrupt these preferential paths and constrain adversarial influence [52, 53, 54, 55, 56, 57]. Third, we incorporate homophily-aware LLM annotation strategies that embed structural priors into prompt design, improving annotation quality and reducing semantic drift in low-label

scenarios [122, 42, 1]. Together, these strategies form a unified framework for robust, structure-sensitive node labeling.

2.4 Research Challenges and Opportunities

Ensuring accurate and efficient node label remains a critical challenge in graph-based learning [110, 1, 4, 5]. In unsupervised learning, our research reveals that existing algorithms exhibit inherent learning biases, which create vulnerabilities that adversaries can exploit [18, 16, 17]. Specifically, the dependence on structural heuristics in trust estimation introduces predictable annotation patterns [19, 20, 21], enabling adversarial entities to manipulate trust score through targeted perturbations. This challenge is particularly severe in financial networks, where even minor distortions in trust scores can lead to incorrect risk assessments and fraudulent transactions [22, 23, 24, 25, 26, 8]. The presence of such biases necessitates the development of robust annotation frameworks that preserve computational efficiency while mitigating adversarial influence [10, 27, 28].

In semi-supervised learning, the fixed nature of the training set and its associated label propagation introduce additional risks. In contrast to unsupervised learning, where labels are inferred solely based on structural or attribute similarities without any supervision, semi-supervised models rely on feature aggregation and label propagation from a small set of labeled nodes to classify previously unseen nodes. [125, 126]. This reliance on propagation mechanisms exposes structural vulnerabilities, as adversaries can exploit these processes to bias model predictions [34, 35]. Structural perturbations targeting labeled nodes can propagate throughout the graph, distorting classification boundaries and increasing the likelihood of misclassification [69, 33, 127, 128, 35, 36]. This issue is especially critical in poisoning attack scenario, where adversarial modifications to a small subset of edges can disproportionately affect model performance [35, 77, 79].

Label acquisition is often expensive and labor-intensive, especially in large-scale applications where manual annotation is impractical [111, 112]. The emergence of large language models (LLMs) offers a promising alternative by generating pseudo-labels based on structural and semantic information [115, 6, 7, 116, 117]. This approach significantly enhances annotation efficiency while reducing costs [118, 119, 120, 121]. However, the reliability of LLM-generated annotations remains a key concern, as misalignments between inferred labels and the underlying graph topology may introduce instability and reduce overall annotation quality [122].

Given these challenges, achieving high-quality initial annotations and ensuring the robustness of model-predicted labels are key objectives of our research. Addressing these issues requires a multi-faceted approach that integrates adversarial modeling, adaptive defense mechanisms, and annotation refinement. [123, 124, 99, 100]. Our work contributes to this field by developing:

- **Adversarial Risk Analysis in Unsupervised Trust Estimation:** Quantifying how algorithmic learning biases introduce exploitable vulnerabilities in annotation processes [23, 46, 10, 16, 17].
- **Structural Defense Mechanisms in Semi-Supervised Learning:** Implementing strong-tie-based trust zone partitioning and federated learning-based boundary inference to enhance the resilience of label propagation [12, 104, 10, 46].
- **LLM-Guided Annotation Optimization:** LLM-Guided Annotation Optimization: Leveraging strong-tie-aware LLM frameworks to enhance label reliability, improving annotation accuracy and GNN performance [113, 114, 122, 116].

By addressing these challenges, our research aims to improve the reliability of label-based graph learning, ensuring that both initial labels and model-predicted annotations maintain high integrity in adversarial environments [108, 111, 117, 18, 22, 29]. Furthermore, this study establishes a pathway for designing robust graph learning

systems that integrate adversarially resilient label propagation, adaptive learning architectures, and strong-tie-aware annotations. These methodologies will play a critical role in advancing the security and interpretability of graph neural networks in real-world applications.

Chapter 3

Strong-Tie Exploitation in Node Labeling: A Sybil Attack Perspective on Trust Prediction

3.1 Motivation

Trust and authenticity form the cornerstone of online social networks, particularly in financial ecosystems where security and integrity are paramount. With the increasing reliance on decentralized platforms, ensuring the robustness of trust mechanisms has become a pressing challenge [18, 19, 20, 21]. Signed graphs provide a fundamental framework for modeling trust relationships, where edges carry positive or negative signs to represent cooperative or adversarial interactions [47]. These structures are widely used in various domains, including social networks [9], sentiment analysis [38], and trust evaluation [64]. For instance, in online peer-to-peer marketplaces [24, 25, 26], user feedback forms a signed network, where positive edges signify trust and negative edges indicate distrust. Similarly, in cryptocurrency ecosystems, signed graphs facilitate fraud detection and transaction flow analysis, offering insights into

both legitimate and fraudulent behaviors [22, 24, 25, 26].

Bitcoin transaction networks, as a representative of signed financial ecosystems, inherently encode trust relationships through user interactions. Traditional transaction models, which classify exchanges as either successful or failed [8], fail to capture the nuanced interplay of trust, fraud, and deception. Signed networks, by contrast, offer a more detailed trust representation, where directed edges encode whether a transaction reflects a trustworthy (positive) or untrustworthy (negative) interaction [22]. While this additional structure enhances security assessments [22], the anonymity inherent in transaction networks introduces new vulnerabilities—for instance, enabling Sybil attacks through identity forgery [25, 26].

Trust prediction within signed graphs is crucial for maintaining the integrity of interactions. The Fairness and Goodness Algorithm (FGA) and the Review to Reviewer (REV2) framework have emerged as two prominent unsupervised learning models for trust estimation [10, 46]. FGA relies on recursively defined fairness and goodness metrics to assess user credibility, while REV2 extends these metrics by refining network-based fraud analysis. Both models aim to infer trust scores from network structures without requiring manually labeled data, making them essential for large-scale and dynamic environments. Despite their effectiveness, these trust prediction models exhibit inherent structural biases, which adversaries can exploit to manipulate trust scores [48, 49, 45]. The robustness of trust prediction is challenged by adversarial attacks, particularly Sybil attacks [30, 94, 28, 27, 16].

In particular, this study investigates the vulnerabilities of FGA and REV2 under indirect Sybil attacks—an adversarial strategy that systematically manipulates trust propagation without direct infiltration of the victim’s immediate network. Lizurej et al. introduced a model where adversaries inject edges through intermediary nodes to affect target trust scores, revealing structural vulnerabilities in propagation mechanisms [23]. These attacks exploit intermediary nodes to affect target trust scores and have traditionally been considered weaker than direct Sybil attacks. However, our

findings challenge this assumption by demonstrating that strong-tie structures and preferential paths significantly amplify adversarial influence, enabling even indirect attacks to become highly effective.

Specifically, our study explores the structural vulnerabilities in trust prediction by analyzing both FGA and REV2 under adversarial conditions. We introduce a novel gradient-based adversarial framework, termed *vicinage-attack*, which strategically inserts edges to maximize adversarial influence while maintaining stealthy edge insertions that blend into the network’s structural regularities. Initially, we differentiate strong ties and weak ties in FGA, identifying their roles in trust propagation. We further extend this analysis to preferential paths in REV2, uncovering their role in adversarial exploitation. Our results show that certain structural formations within the graph create preferential pathways that accelerate the spread of adversarial influence. Through optimized attack design, we demonstrate that leveraging strong-tie structures significantly enhances the manipulation potential of indirect attacks. Our key contributions include:

- Identifying strong ties and preferential paths as topological structures that serve as key vulnerabilities in trust prediction models.
- Demonstrating how trust propagation biases in FGA and REV2 create susceptibilities to adversarial influence, allowing attackers to exploit these structural tendencies for manipulation.
- Highlighting the impact of structural preferences in amplifying adversarial attacks, challenging prior assumptions about the effectiveness of indirect Sybil attacks.
- Developing a refined understanding of adversarial strategies that leverage network topology to enhance the effectiveness of trust score manipulation.
- Proposing a novel framework to quantify structural vulnerabilities in signed

networks, providing insights into defense strategies for mitigating adversarial threats.

By uncovering these structural weaknesses, this study lays the groundwork for a more nuanced understanding of adversarial trust manipulation and informs the development of more resilient defense mechanisms. The remainder of this chapter is organized as follows:

- **Section II** presents the FGA and REV2 trust models, along with the adversarial techniques used in indirect Sybil attack scenario.
- **Section III** presents the specific problem statements and challenges addressed in this research, highlighting existing gaps in trust prediction security.
- **Section IV** explores network structures, distinguishing strong ties and weak ties in FGA and preferential paths in REV2, and their implications for adversarial threats.
- **Section V** introduces our proposed vicinage-attack approach, detailing its design principles and adversarial objectives.
- **Section VI** presents empirical results comparing our attack model against baseline adversarial strategies, demonstrating the effectiveness of structural exploitation.
- **Section VII** discusses the broader implications and practical applications of our findings, particularly in trust-aware financial and social network systems.
- **Section VIII** outlines limitations and future research directions, paving the way for enhanced defense mechanisms against trust manipulation.
- **Section IX** concludes the first study on mislabeling in unsupervised trust prediction and outlines directions for improving robustness.

3.2 Preliminaries

The primary objective of the target model is to compute trust scores for each node by leveraging graph-level information. This is accomplished using the Fairness and Goodness Algorithm (FGA) and Review to Reviewer (REV2), both of which iteratively define fairness $f(u)$ and goodness $g(v)$. The fairness metric represents how impartially node u assesses others, while the goodness metric quantifies the degree of trustworthiness assigned to node v by its evaluators. In REV2, an additional edge-level metric, reliability $r(u, v)$, is incorporated into the iterative learning process, ensuring that ratings given by fairer users are more aligned with the goodness scores of the nodes they evaluate.

Formally, consider a directed, weighted signed graph $G = (U, R, W)$, where a user $u \in U$ assigns a rating $(u, v) \in R$ to another user $v \in U$. Here, U represents the complete set of users, R denotes the set of ratings, and W contains the numerical values associated with these ratings. The weight $W(u, v) \in [-1, 1]$ encodes the degree of trust or distrust expressed by node u toward node v . To systematically model trust propagation and evaluation, we define:

- $\text{In}(v)$ as the collection of ratings received by node v .
- $\text{Out}(u)$ as the collection of ratings issued by node u .
- $|\text{In}(v)|$ as the total number of ratings node v has received.
- $|\text{Out}(u)|$ as the total number of ratings node u has provided.

This structured approach facilitates the computation of trust scores while offering insights into the intricate dynamics of trust within the network.

3.2.1 The FGA Trust Model: Fairness-Goodness Propagation in Signed Networks

The Fairness and Goodness Algorithm (FGA) provides a foundational trust inference mechanism for weighted signed networks, where both cooperative and adversarial interactions are encoded through edge signs. Given a directed signed graph $G = (V, E, W)$, each edge weight $W(u, v) \in [-1, 1]$ represents the level of trust node u assigns to node v . A value close to 1 indicates high trust, while a value near -1 signals distrust.

Let $In(u)$ denote the set of incoming ratings (i.e., ratings received by u), and $Out(v)$ denote the set of outgoing ratings (i.e., ratings given by v). The fairness of a node v , denoted as $f(v) \in [0, 1]$, captures how reliably it evaluates others. The goodness of node u , denoted as $g(u) \in [-1, 1]$, represents its overall trustworthiness as perceived by others. FGA operates through recursive propagation of fairness and goodness values, defined as:

$$\begin{cases} g^{t+1}(u) = \frac{1}{|In(u)|} \sum_{v \in In(u)} f^t(v) \times W(u, v), \\ f^{t+1}(v) = 1 - \frac{1}{|Out(v)|} \sum_{u \in Out(v)} \frac{|W(u, v) - g^{t+1}(u)|}{2} \end{cases} \quad (3.1)$$

Both fairness and goodness values are initialized as 1 for all nodes and iteratively updated until convergence. The update process halts when the change in values across iterations falls below a small threshold ε , as defined by the convergence criteria:

$$|g^{t+1}(u) - g^t(u)| < \varepsilon \quad \text{or} \quad |f^{t+1}(u) - f^t(u)| < \varepsilon.$$

This iterative process ensures that trust scores stabilize over the signed network. In practical applications, FGA has been widely adopted in domains such as peer-to-peer marketplaces and financial networks due to its ability to infer credibility from

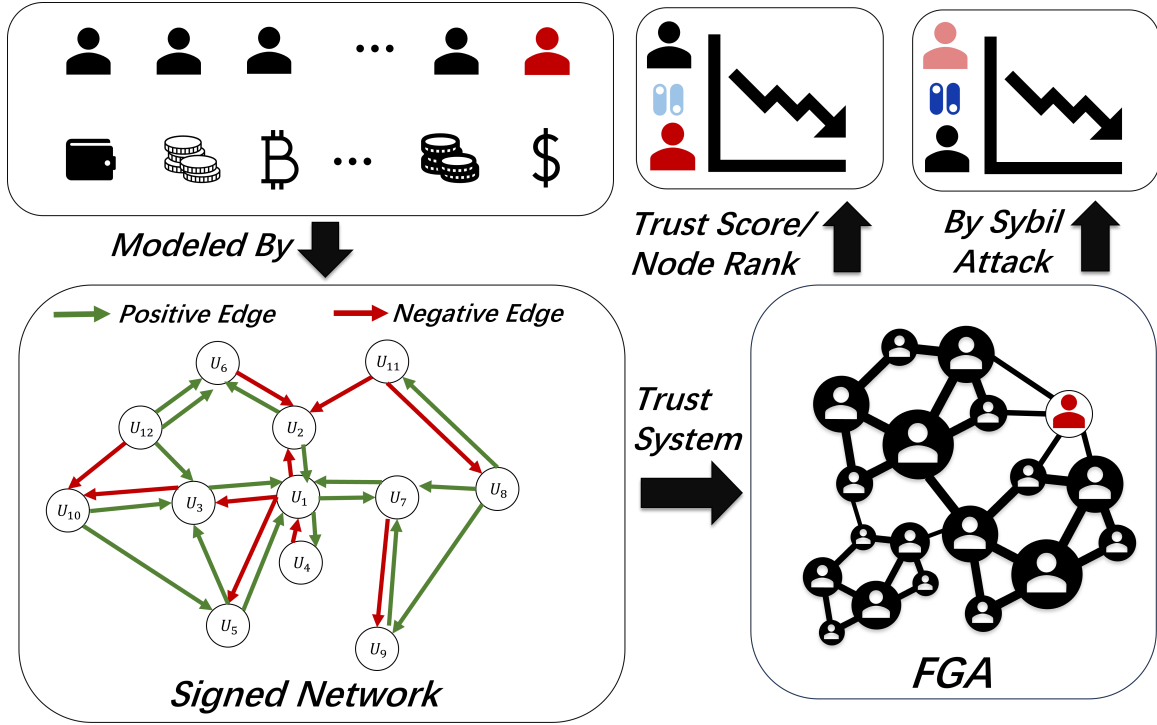


Figure 3.1: FGA as a trust/reputation model in signed graphs, demonstrating attack vulnerabilities.

structure alone. A visual overview of the trust propagation architecture in FGA is illustrated in Figure 3.1.

Importantly, Lizurej et al. [22] demonstrated that manipulating FGA goodness scores through indirect adversarial attacks is NP-hard, posing a significant challenge in trust system security. As such, the goodness metric is often selected as the primary target for adversarial influence, particularly under budget-constrained edge perturbation scenarios where attackers aim to subtly degrade trust values while maintaining plausible structural appearances.

3.2.2 The REV2 Framework: Structured Review Modeling for Trust Inference

The recursive formulation of the Review to Reviewer (REV2) algorithm is presented as follows:

$$\begin{cases} g^{t+1}(v) = \frac{\sum_{(u,v) \in In(v)} r^{t+1}(u,v) \cdot W(u,v) + \beta_1 \cdot \mu_g}{|In(v)| + \beta_1}; \\ r^{t+1}(u,v) = \frac{\gamma_1 \cdot f^{t+1}(u) + \gamma_2 \left(1 - \frac{|W(u,v) - g^t(v)|}{2}\right)}{\gamma_1 + \gamma_2}; \\ f^{t+1}(u) = \frac{\sum_{(u,v) \in Out(u)} r^t(u,v) + \alpha_1 \cdot \mu_f}{|Out(u)| + \alpha_1} \end{cases} \quad (3.2)$$

Within this framework, the values of fairness $f(u)$, goodness $g(v)$, and reliability $r(u,v)$ are constrained to the range $[0, 1]$. The parameters μ_f and μ_g represent the mean fairness and goodness scores across all users, respectively. Initially, all values of $f(u)$, $g(v)$, and $r(u,v)$ are set to 1 and iteratively refined through an update process until convergence is reached. The convergence criterion is established as:

$$\max\{|g^{t+1}(u) - g^t(u)|, |f^{t+1}(u) - f^t(u)|, |r^{t+1}(u,v) - r^t(u,v)|\} < \varepsilon,$$

where ε is a predefined small threshold ensuring numerical stability. As highlighted in [23], the ranking of fairness scores serves as a robust indicator for detecting fraudulent users. To maintain consistency in calculations, the hyperparameters α_1 , γ_1 , γ_2 , and β_1 are uniformly set to 1 throughout the evaluation process.

3.2.3 Indirect Sybil Attacks: Structural Exploitation for Trust Manipulation

The Sybil attack is a widely recognized threat in peer-to-peer networks, where an adversary generates multiple fake identities to impersonate legitimate users and disrupt the network’s intended functionality [27, 28, 29]. Douceur demonstrated that trusted certification remains the only foolproof method to completely prevent Sybil attacks [30]. Lizurej et al. introduced the concept of an indirect Sybil attack targeting the FGA model, where the attacker’s objective is to significantly degrade the trust scores of specific target nodes by strategically injecting edges into the original graph under a limited edge budget [31].

More specifically, the problem involves a weighted signed network, a set of attacker nodes S , a set of target nodes T , an intermediary set I , a predefined edge budget k , and a trust threshold $r \in [-1, 1]$. Figure 3.2 provides a detailed visualization of this attack model. The attacker’s goal is to determine whether they can add at most k edges to manipulate the network structure in such a way that the trust scores of all nodes in T fall below the threshold r . Notably, the attacker can only influence the target nodes through interactions with the intermediary set I . It has been formally proven that solving this problem is NP-hard [23].

3.2.4 Graph Structural Attacks and Their Influence on Node Labeling Stability

Graph structure attacks aim to alter the adjacency matrix of a graph to induce significant classification errors in nodes or edges [31, 32]. In a gray-box setting, the attacker leverages gradients on the adjacency matrix in a poisoning manner to determine optimal modifications [33, 34, 35]. In contrast, in a black-box scenario, where direct access to the victim model is unavailable, the attacker employs a surrogate model to

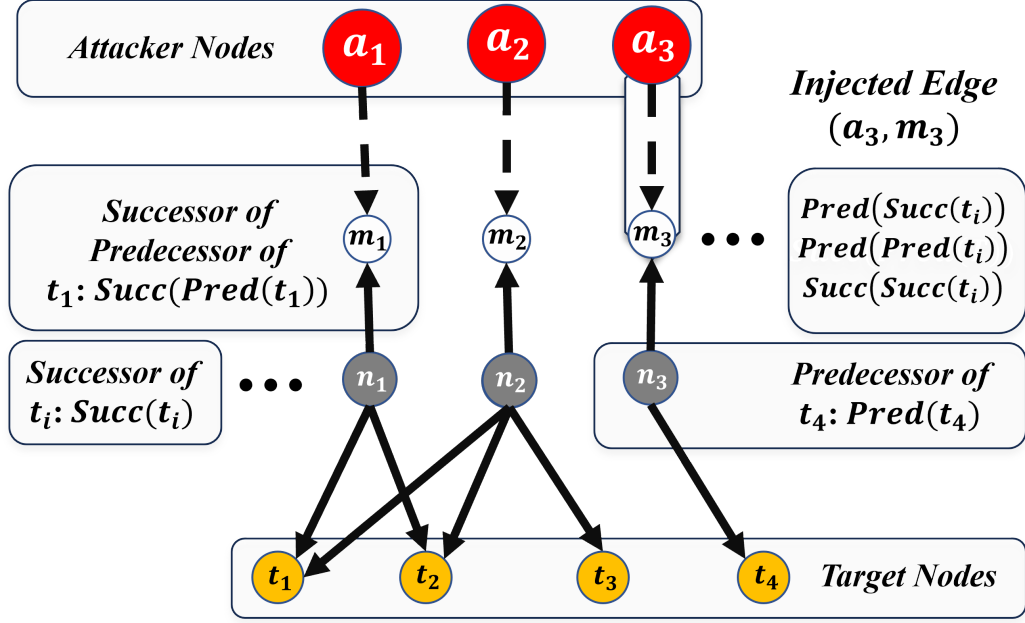


Figure 3.2: Indirect Sybil attack on FGA: visualization of attacker, intermediary, and target nodes.

approximate the target system and identify perturbations that maximize adversarial impact [129].

3.3 Problem Statements

3.3.1 Threat Model in Trust Manipulation

This study develops a comprehensive threat model for adversarial attacks on trust prediction in signed social networks, focusing on trust evaluation systems such as FGA and REV2. The model encapsulates the following critical components:

Attacker's Goal: The primary objective of adversaries is to manipulate trust prediction mechanisms to achieve malicious ends. These goals may include degrading the trustworthiness of specific users, masking deceptive behavior, or artificially inflat-

ing trust scores to gain undue credibility. Attackers accomplish this by introducing adversarial perturbations that strategically alter the network structure.

Attacker’s Knowledge and Capabilities: It is assumed that adversaries possess full knowledge of the trust prediction algorithm and the network topology. They have the ability to modify the network structure by injecting new edges, modifying existing relationships, or altering weight distributions to influence trust scores. Among these techniques, edge injection is the primary focus, as edge deletion is often infeasible due to technical and operational constraints. For any injected edge (i, j) directed from node i to node j , it is referred to as a positive attack originating from i or a passive attack affecting j .

Stealth Characteristics of Attacks: In adversarial scenarios, particularly in indirect attacks, stealth plays a pivotal role in ensuring success. Unlike direct attacks, which are more conspicuous, indirect attacks seamlessly integrate into genuine trust propagation processes, making them more difficult to detect. Attackers strategically manipulate trust relationships to evade detection while leveraging target nodes, their neighbors, and designated attacker nodes. The effectiveness of these attacks hinges on their ability to remain undetected within the network.

In summary, this threat model underscores the methods adversaries use to exploit strong ties and preferential paths in signed graphs. By leveraging their understanding of network structures, trust propagation algorithms, and perturbation mechanisms, attackers introduce subtle yet impactful modifications, highlighting the significance of stealth in adversarial trust manipulation.

3.3.2 Formal Problem Formulation

We formalize indirect trust manipulation attacks as a bi-level combinatorial optimization problem with structural constraints, targeting signed networks modeled as $G = (V, E, W)$, where V is the set of users/nodes, $E \subseteq V \times V$ is the edge set, and

$W : E \rightarrow \{-1, 0, +1\}$ is the signed weight function representing distrust, neutral, or trust relationships. Let $\mathcal{Y}_0 : V \rightarrow \mathbb{R}$ denote initial trust scores estimated by the FGA framework [10].

The attacker injects a set of perturbations $B \subseteq V \times V \setminus E$ under a cardinality constraint $|B| \leq k$, resulting in a perturbed graph $\hat{G} = (V, E \cup B)$. The objective is to maximize the cumulative distortion of trust scores over a target set $T \subseteq V$, where the attacker seeks to either degrade (demote) or elevate (promote) the perceived credibility of those nodes. This yields the following general bilevel optimization:

$$\begin{aligned}
& \underset{B \subseteq \Phi_{S,T}}{\text{maximize}} & \mathcal{L}_{\text{attack}} &= \sum_{v \in T} \phi(\mathcal{Y}_\epsilon(v)) - \phi(\mathcal{Y}_0(v)) \\
& \text{subject to} & |B| &\leq k, \quad B \cap (S \times T) = \emptyset \\
& & \mathcal{Y}_\epsilon &= f_{\text{FGA}}(G + B)
\end{aligned} \tag{3.3}$$

where $\phi(y) = y$ corresponds to *trust demotion* and $\phi(y) = -y$ reflects *trust promotion*. This direction-dependent formulation allows the attacker to either decrease or increase node scores based on specific adversarial goals. To capture the structural constraints of indirect Sybil attacks, we restrict the feasible perturbation space $\Phi_{S,T}$ to intermediary-connected edges only. Specifically:

$$\Phi_{S,T} = \bigcup_{i=0}^3 \Phi_{S,T}^{(i)} \subseteq (S \times V) \setminus (S \times T) \tag{3.4}$$

where each region $\Phi_{S,T}^{(i)}$ reflects a distinct 2-hop connectivity pattern between attacker set S and target set T , defined as:

$$\Phi_{S,T}^{(0)} = S \times \text{Succ}(\text{Pred}(T)) \quad (3.5)$$

$$\Phi_{S,T}^{(1)} = S \times \text{Pred}(\text{Succ}(T)) \quad (3.6)$$

$$\Phi_{S,T}^{(2)} = S \times \text{Succ}(\text{Succ}(T)) \quad (3.7)$$

$$\Phi_{S,T}^{(3)} = S \times \text{Pred}(\text{Pred}(T)) \quad (3.8)$$

Each candidate edge $(u, v) \in \Phi_{S,T}$ is assigned a signed weight $w_{uv} \in \{-1, +1\}$, and the goal becomes selecting an optimal set B^* that maximizes distortion under cost constraints. This can be further rewritten in two alternative forms depending on the design of the attack loss.

Variant A: Absolute Trust Difference Objective

$$B^* = \arg \max_{\substack{B \subseteq \Phi_{S,T} \\ |B| \leq k}} \sum_{v \in T} |\mathcal{Y}_\epsilon(v) - \mathcal{Y}_0(v)| - \lambda \|B\|_c \quad (3.9)$$

This variant treats trust score changes symmetrically, penalizing deviation magnitude regardless of direction. It is suitable for generalized robustness testing or untargeted attacks.

Variant B: Directional Promotion/Demotion Objective

$$B^* = \arg \max_{\substack{B \subseteq \Phi_{S,T} \\ |B| \leq k}} \sum_{v \in T} [\phi(\mathcal{Y}_\epsilon(v)) - \phi(\mathcal{Y}_0(v))] - \lambda \|B\|_c \quad (3.10)$$

where

$$\phi(y) = \begin{cases} y & \text{(Demotion: lower trust is desired)} \\ -y & \text{(Promotion: higher trust is desired)} \end{cases}$$

This version allows targeted control over score manipulation direction and is more consistent with realistic adversarial goals in Sybil attacks, fraud amplification, or black-market promotion in decentralized networks.

Discussion: Promotion vs. Demotion Strategies In trust manipulation scenarios over signed graphs, two general strategies are available to attackers: promotion and demotion, corresponding to the insertion of positive or negative edges, respectively. When the target node is fixed, the attacker faces a combinatorial space of possible 2-hop pathways, making optimization difficult due to the multitude of interaction types between the attacker and target via intermediate nodes.

To simplify this challenge and enable targeted evaluation, we isolate demotion attacks under the FGA model and promotion attacks under the REV2 framework, aligning each model with its respective vulnerability. In FGA, demotion is achieved by injecting negative edges into structurally influential positions, which directly **degrade the goodness scores of target nodes**. These perturbations propagate recursively through inbound paths and are particularly effective when placed within strong-tie subgraphs, where the amplification of negative influence is most pronounced. The adversarial impact emerges from disrupting fairness-aware aggregation in structurally sensitive neighborhoods.

In contrast, REV2 supports promotion through positive edge insertions aimed at **improving a node’s fairness-based ranking**. The attacker seeks to embed nodes into high-fairness regions by establishing transitive links through reliable intermediaries. This strategy depends on status-aware propagation, where subtle edge placements into highly ranked subgraphs can disproportionately elevate the perceived trustworthiness of the attacker. Unlike demotion, promotion often demands higher stealth and careful alignment with fairness constraints to avoid detection or inconsistent scoring.

By separately modeling demotion in FGA and promotion in REV2, we reduce the complexity of edge-space search while capturing the essential vulnerabilities of each system. This separation allows for a clearer understanding of attacker preference and provides a structured foundation for evaluating perturbation effects in signed trust networks.

3.3.3 Vicinage Attack Implementation Pipeline

The adversarial execution process consists of three operational phases designed to systematically identify, select, and inject edges in a way that maximizes attack effectiveness while maintaining stealth.

Phase 1: Topology Analysis. The attacker begins by analyzing the graph topology to identify optimal intermediary nodes for indirect edge insertion. Given a source node set $S \subseteq V$ and a target node set $T \subseteq V$, the attacker computes the betweenness centrality $\sigma_{st}(v)$ for each node $v \in V \setminus (S \cup T)$, which represents the number of shortest paths from any source $s \in S$ to any target $t \in T$ that pass through v . A subset of nodes with the highest centrality values is selected as the intermediary set $I \subseteq V \setminus (S \cup T)$. Using this set, the attacker constructs a bipartite connection graph $\mathcal{B}(S, I)$, representing all possible indirect attack paths from S to T via injected edges targeting intermediary nodes.

Phase 2: Edge Selection. With intermediary nodes identified, the attacker constructs a candidate edge set $\mathcal{C} = \{(s, i) \mid s \in S, i \in I\}$, representing potential edges to be injected. For each edge $e \in \mathcal{C}$, an edge criticality score is computed as $\rho(e) = \frac{\partial \mathcal{L}_{\text{attack}}}{\partial e}$, where $\mathcal{L}_{\text{attack}}$ denotes the adversarial loss function. To identify the most effective edges under a budget constraint k , the attacker solves the following knapsack-style optimization problem:

$$\max \sum_{e \in \mathcal{C}} \rho(e) x_e \quad \text{subject to} \quad \sum_{e \in \mathcal{C}} x_e \leq k,$$

where $x_e \in \{0, 1\}$ indicates whether edge e is selected for injection.

Phase 3: Stealth Enforcement. To maintain stealth, the attacker minimizes structural deviation from the original graph. This is achieved by enforcing a Frobenius norm constraint on the difference between the perturbed adjacency matrix $\hat{G}$ and the original graph G , such that:

$$\|\hat{G} - G\|_F \leq \epsilon,$$

where ϵ is a small perturbation threshold. Furthermore, for each intermediary node $v \in I$, the attacker ensures that the degree change is bounded by a tolerance δ , satisfying:

$$|\deg_{\hat{G}}(v) - \deg_G(v)| \leq \delta,$$

where $\deg_{\hat{G}}(v)$ and $\deg_G(v)$ represent the degrees of node v in the perturbed and original graphs, respectively. This three-phase strategy enables the attacker to introduce high-impact perturbations while evading detection by preserving structural consistency.

The efficacy of trust metric manipulation is highly dependent on the structural pathways through which influence propagates—particularly the strength of social ties [130]. In this study, we define strong ties as 2-hop paths within the perturbation space that exhibit high adversarial sensitivity. These ties often involve intermediary nodes with high structural influence or alignment with attacker objectives, making them prime targets for manipulation due to their disproportionate impact on trust scores [16, 17]. In contrast, weak ties refer to other 2-hop indirect paths that show attenuated adversarial effects. These paths propagate trust signals with reduced amplification and thus contribute less to the attack’s overall efficacy [37, 27]. As shown in Figure 3.3, we empirically analyze these two categories within the 2-hop perturbation space. This distinction reveals two propagation regimes that inform the design of both attack strategies and defense mechanisms in signed trust networks. The perturbation space architecture connects attacker nodes S to targets T through four strategic intermediary classes:

$$\mathcal{I} = \underbrace{Succ(Pred(T))}_{\text{Second-order neighbor 1}} \cup \underbrace{Pred(Succ(T))}_{\text{Second-order neighbor 2}} \cup \underbrace{Pred(Pred(T))}_{\text{Second-order neighbor 3}} \cup \underbrace{Succ(Succ(T))}_{\text{Second-order neighbor 4}} \quad (3.11)$$

3.4 Adversarial Preference over Strong-Tie Paths

3.4.1 Defining and Classifying Tie Structures

To rigorously analyze trust propagation vulnerabilities, we define tie structures — fundamental building blocks of adversarial influence in trust networks. These structures define how attacker nodes $\mathcal{S}$ manipulate target nodes $\mathcal{T}$ through intermediaries $\mathcal{I}(\mathcal{T}) \subseteq \mathcal{V} \setminus (\mathcal{S} \cup \mathcal{T})$. By analyzing the role of strong and weak ties, we can better understand the structural weaknesses that adversaries exploit in trust prediction models.

Definition 1 (Tie Structure). *A tie structure $\tau(s, t)$ between an attacker $s \in \mathcal{S}$ and a target $t \in \mathcal{T}$ is a minimal subgraph $\mathcal{G}_\tau \subseteq \mathcal{G}$ that facilitates trust manipulation. Formally, it consists of:*

$$\mathcal{G}_\tau = (\{s, t\} \cup \mathcal{I}_\tau, \mathcal{E}_\tau)$$

where:

- $\mathcal{I}_\tau \subseteq \mathcal{I}(\mathcal{T})$ is the intermediary set, representing nodes that relay trust information between s and t .
- $\mathcal{E}_\tau$ is the set of edges, defined as:

$$\mathcal{E}_\tau = \{(s, i)\} \cup \mathcal{P}(i, t)$$

where $\mathcal{P}(i, t)$ denotes a trust propagation path from an intermediary i to the target t .

Definition 2 (Strong Tie). *A tie structure $\tau(s, t)$ is strong if it satisfies the conductance condition:*

$$C(\tau) = \prod_{e \in \mathcal{E}_\tau} w_e \cdot \frac{\partial f(t)}{\partial f(s)} \geq \gamma_{strong}$$

where:

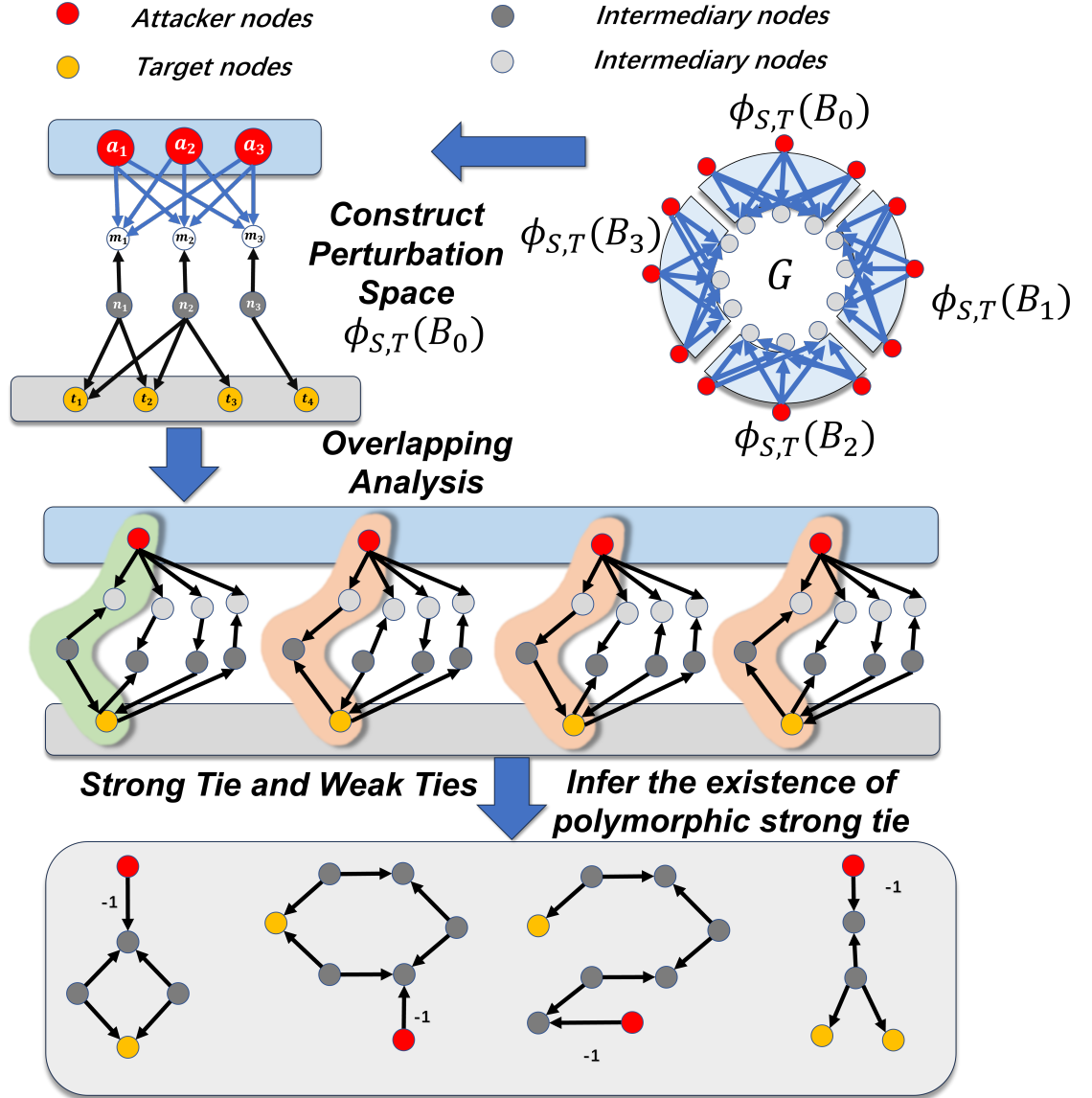


Figure 3.3: Concepts of perturbation space, strong ties, weak ties, and polymorphic strong ties.

- $C(\tau)$ represents the strength of trust propagation.
- w_e is the weight of an edge e , indicating the trust influence between nodes.
- $\frac{\partial f(t)}{\partial f(s)}$ measures the sensitivity of the target node's trust score to the attacker's trust score.
- γ_{strong} is a threshold defining strong ties.

A strong tie in FGA towards goodness typically consists of:

1. Direct predecessor manipulation: $s \rightarrow \text{Pred}(t)$
2. Successor-of-predecessor attack: $s \rightarrow \text{Succ}(\text{Pred}(t))$

Definition 3 (Weak Tie). A tie structure $\tau(s, t)$ is weak if its conductance satisfies:

$$C(\tau) < \gamma_{weak}$$

where γ_{weak} is a lower threshold indicating reduced trust propagation strength. Weak ties arise in scenarios where the attack must propagate through multiple intermediaries:

$$\exists i_1, i_2 \in \mathcal{I}_\tau : s \rightarrow i_1 \rightarrow i_2 \rightarrow t$$

or involves non-propagating configurations such as:

$$s \rightarrow \text{Succ}(t) \quad \text{or} \quad s \rightarrow \text{Pred}(\text{Pred}(t))$$

Definition 4 (Polymorphic Strong Tie). A polymorphic strong tie $\tau_p(s, t)$ is a composite structure formed by overlapping strong ties:

$$\tau_p = \bigcup_{k=1}^n \tau_k(s, t)$$

satisfying the enhanced conductance condition:

$$C(\tau_p) = \sum_{k=1}^n C(\tau_k) \geq n\gamma_{strong}$$

Types of Poly-ST structures include:

Table 3.1: Structural Properties of Multi-Hop Tie Structures

Tie Type	Conductance Strength	Propagation Depth
Strong Tie I	High	3
Strong Tie II	Moderate to High	2
Weak Tie I-X	Low to Very Low	2 or 3
Poly-ST IV	Very High	≥ 2

- *Type I: Multiple independent strong ties between s and t .*
- *Type II: Hybrid compositions of strong and weak ties, reinforcing adversarial influence.*
- *High-Order: Recursively constructed structures where multiple strong ties interact, amplifying attack effectiveness.*

Table 3.2: Structural patterns of indirect attack ties within FGA.

Categories	Positive attack	Passive attack
Pred(T)	✗	✓
Succ(T)	✗	✗
Succ(Pred(T))	✓	✗
Pred(Succ(T))	✗	✗
Succ(Succ(T))	✗	✗
Pred(Pred(T))	✗	✗

✓ indicates the case capable of generating perturbation on the target node’s goodness, which is more likely to be a *strong tie*. ✗ indicates cases that cannot influence the target node’s goodness. The positive and passive attack is in the view of attackers.

3.4.2 Axiomatic and Structural Analysis of Tie Structures in FGA

The iterative learning mechanism in FGA exhibits a discernible propagation bias. This bias, reflected in the way trust propagates across *tie structures*, can be analyzed through the concepts of *conductivity* and *resistance*. Trust propagation is naturally inclined to follow a *preferential path*. A detailed theoretical derivation of these properties can be found in [10, 23].

Axiom 1. (*Direction-Oriented Rigging*) *A perturbation applied to $f(Pred(t))$ affects $g(t)$, whereas a perturbation on $g(Succ(t))$ influences $f(t)$.*

Since an incoming edge of t originates from $Pred(t)$ and an outgoing edge of t leads to $Succ(t)$, trust scores are recursively interdependent. As shown in Eq. (3.1), altering $f(Pred(t))$ modifies $g(t)$, and adjusting $g(t)$ subsequently impacts $f(Pred(t))$. Likewise, perturbing $g(Succ(t))$ influences $f(t)$, and modifying $f(t)$ results in changes to $g(Succ(t))$. This axiom underscores the mutual dependency of trust metrics in recursive learning. Prior work [10] has similarly characterized this interdependence through the *Goodness Axiom* and *Fairness Axiom*.

Axiom 2. (*Smooth Goodness*) *An increase in $f(Pred(t))$ leads to a proportional increase in $g(t)$.*

This property aligns with the *Citation Axiom* presented in [23] (Axiom 1), which describes the smooth and structured propagation of trustworthiness.

Axiom 3. (*Fairness Metric Consistency*) *If node $Pred(t)$ assigns ratings to all its successors with a rating bias of $\gamma = |g(u) - W(Pred(t), u)| = 0$, then its fairness score is maximized at $f(Pred(t)) = 1$. Conversely, if $Pred(t)$ assigns ratings with a bias of $\gamma = |g(u) - W(Pred(t), u)| = 2$, its fairness score is minimized at $f(Pred(t)) = 0$.*

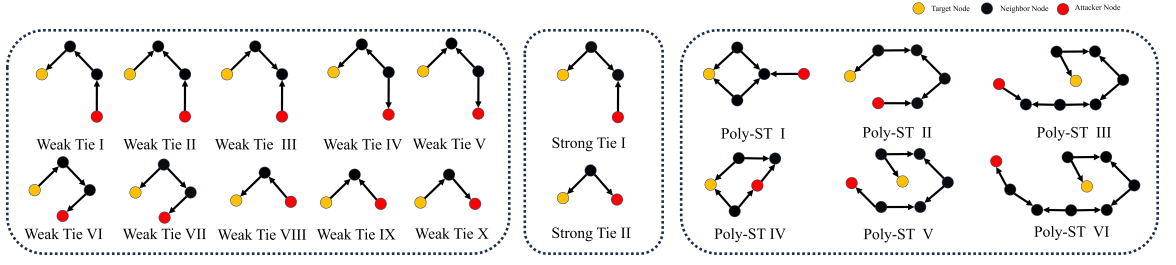


Figure 3.4: Representation of weak ties, strong ties, and polymorphic strong ties in FGA.

This axiom is a reformulation of the *Citation Axiom* (Axiom 9) from [23], emphasizing that fairness scores are determined by the deviation between assigned ratings and actual trustworthiness.

The axioms presented above provide a theoretical foundation for understanding the propagation dynamics within signed networks and their susceptibility to adversarial manipulation. These insights are critical in designing robust trust prediction mechanisms. The process of trust propagation exhibits notable differences in *strong ties* and *weak ties*, where *strong ties* facilitate conductivity and *weak ties* impose resistance. In the following, we analyze the underlying factors contributing to this propagation bias by employing idealized topological models to quantitatively examine trust propagation in FGA. Our findings demonstrate that the effectiveness of *strong ties* is governed by both the sign and weight of the injected edge. To streamline the analysis, we assume that the edge weights w_1 and w_2 in Fig. 3.5 are positive, as negative-weighted edges are comparatively rare in signed networks. For Strong Tie I, the adversarial influence operates by reducing $g(q)$, which sequentially decreases $f(p)$ and subsequently lowers $g(t)$. In contrast, for Strong Tie II, the attack mechanism revolves around reducing $f(q)$, which leads to a decline in $g(t)$.

Theorem 1. (*Resistance of Weak Ties*) Across all weak ties from Weak Tie I to Weak Tie X, the trust score $g(t)$ remains unaffected.

Proof. A perturbation on $f(\text{Pred}(t))$ directly influences $g(t)$, formalized as $\Delta g(t) \leftrightarrow$

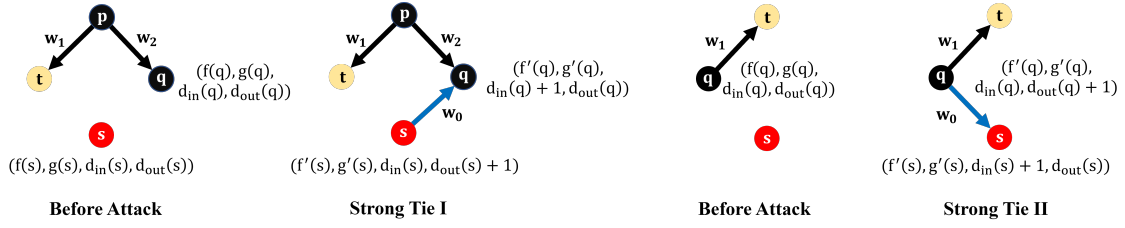


Figure 3.5: Strength analysis of strong ties in trust networks.

$\Delta f(\text{Pred}(t))$. Likewise, a perturbation on $g(\text{Succ}(t))$ modifies $f(t)$, represented as $\Delta f(t) \leftrightarrow \Delta g(\text{Succ}(t))$. The structural impact on $g(t)$ is exclusively mediated through two key propagation sequences:

$$\Delta g(t) \leftrightarrow \Delta f(\text{Pred}(t)) \leftrightarrow \Delta g(\text{Succ}(\text{Pred}(t))), \quad (3.12)$$

and

$$\Delta g(t) \leftrightarrow \Delta f(\text{Pred}(t)). \quad (3.13)$$

These pathways highlight the indirect influence mechanisms within the trust propagation process, emphasizing how upstream structural changes at $\text{Pred}(t)$ cascade through structural dependencies to affect the final trust score $g(t)$. As illustrated in Figure 3.4, we enumerate all possible two-hop attack configurations under the ideal assumption that only a single path exists between attacker and target. By contraposition, all remaining configurations that do not follow these two valid propagation pathways are structurally incapable of influencing $g(t)$, and are thus classified as weak ties. ■

The above analysis establishes that only *strong ties* facilitate meaningful adversarial perturbations in FGA, while *weak ties* serve as insulating elements that prevent the transmission of adversarial influence. This insight is crucial for understanding the effectiveness of different perturbation strategies in adversarial attacks on trust-based systems.

Theorem 2. (Conductivity of Strong Tie I) For Strong Tie I, if the injected edge (s, q) has weight w_0 and satisfies $f(s)w_0 < g(q) < w_2$, then $\Delta g(t) < 0$. Alternatively, if $g(q) > \max(w_2, f(s)w_0)$ and $w_0 < \frac{2w_2-1}{f(s)}$, then $\Delta g(t) < 0$.

Proof. As illustrated in Fig. 3.5, an injected edge modifies the trust score such that $g'(q) = \frac{g(q)d_{in}(q)+f(s)w_0}{d_{in}(q)+1}$. Consequently, the perturbation is given by:

$$\Delta g(q) = g'(q) - g(q) = \frac{f(s)w_0 - g(q)}{d_{in}(q) + 1}.$$

If $g(q) > f(s)w_0$, then $g(q)$ decreases; otherwise, it increases. The corresponding perturbation of $\Delta f(p)$ is expressed as:

$$\Delta f(p) = \frac{1}{2d_{out}(p)} (|w_2 - g(q)| - |w_2 - g'(q)|).$$

Four cases arise:

$$w_2 > g(q), w_2 > g'(q), \quad \Delta f(p) = \frac{\Delta g(q)}{2d_{out}(p)}; \quad (3.14a)$$

$$w_2 < g(q), w_2 > g'(q), \quad \Delta f(p) = \frac{2g(q) - 2w_2 + \Delta g(q)}{2d_{out}(p)}; \quad (3.14b)$$

$$w_2 < g(q), w_2 < g'(q), \quad \Delta f(p) = -\frac{\Delta g(q)}{2d_{out}(p)}; \quad (3.14c)$$

$$w_2 > g(q), w_2 < g'(q), \quad \Delta f(p) = \frac{2w_2 - 2g(q) - \Delta g(q)}{2d_{out}(p)}. \quad (3.14d)$$

To ensure $\Delta g(q) < 0$ and $\Delta f(p) < 0$, we require that $\Delta g(t) = \frac{\Delta f(p)w_1}{d_{in}(t)} < 0$ for $w_1 > 0$. Among these cases, (3.14c) leads to $\Delta f(p) > 0$ when $\Delta g(q) < 0$, and (3.14d) implies $\Delta g(q) > 0$ if $g(q) < w_2 < g'(q)$. Thus, we need to analyze the remaining two cases.

Case I: If $f(s)w_0 < g(q) < w_2$, then $\Delta g(q) < 0$, $\Delta f(p) < 0$, and $\Delta g(t) < 0$.

Case II: Given that $\Delta g(q) = \frac{f(s)w_0 - g(q)}{d_{in}(q)+1} < 2w_2 - 2g(q)$, we derive:

$$-1 < g(q) < \frac{2(d_{in}(q) + 1)w_2 - f(s)w_0}{2d_{in}(q) + 1} < 1.$$

$$\frac{2w_2 - f(s)w_0 - 1}{2(1 - w_2)} < d_{in}(q) < \frac{2w_2 - f(s)w_0 - g(q)}{2(g(q) - w_2)}.$$

Thus, if $g(q) > \max(w_2, f(s)w_0)$ and $w_0 < \frac{2w_2-1}{f(s)}$, then $\Delta g(q) < 0$, $\Delta f(p) < 0$, and $\Delta g(t) < 0$. ■

Theorem 3. (*Conductivity of Strong Tie II*) For Strong Tie II, if an injected edge (q, s) carries weight w_0 and satisfies $f(p)w_0 < g(s) < w_0$, then $\Delta g(t) < 0$. Alternatively, if $g(q) > \max(w_0, f(s)w_0)$ and $w_0 > \frac{1}{2-f(s)}$, then $\Delta g(t) < 0$.

Proof. As depicted in Fig. 3.5, an injected edge attack leads to:

$$\Delta g(s) = g'(s) - g(s) = \frac{f(q)w_0 - g(s)}{d_{in}(s) + 1},$$

$$\Delta f(q) = \frac{1}{2d_{out}(q)}(|w_0 - g(s)| - |w_0 - g'(s)|).$$

We require $\Delta f(q) < 0$ so that $\Delta g(t) = \frac{\Delta f(q)w_1}{d_{in}(t)} < 0$ for $w_1 > 0$.

Case I: If $f(p)w_0 < g(s) < w_0$, then $\Delta g(s) < 0$, $\Delta f(q) < 0$, and $\Delta g(t) < 0$.

Case II: Given that $\Delta g(s) = \frac{f(q)w_0 - g(s)}{d_{in}(s) + 1} < 2w_0 - 2g(q)$, we obtain:

$$-1 < g(s) < \frac{2(d_{in}(s) + 1)w_0 - f(s)w_0}{2d_{in}(s) + 1} < 1.$$

$$\frac{2w_0 - f(s)w_0 - 1}{2(1 - w_0)} < d_{in}(s) < \frac{2w_0 - f(s)w_0 - g(q)}{2(g(q) - w_0)}.$$

Therefore, if $g(q) > \max(w_0, f(s)w_0)$ and $w_0 > \frac{1}{2-f(s)}$, then $\Delta g(s) < 0$, $\Delta f(q) < 0$, and $\Delta g(t) < 0$. ■

3.4.3 Empirical Analysis of Tie-Driven Attack Preferences in REV2

Our objective is to determine which nodes play a crucial role in shaping *preferential paths* within the REV2 framework. To aid in this analysis, we incorporate *status theory*, as summarized in Table 3.3. In a signed directed graph, a positive edge from node A to node B signifies that A perceives B to have a higher status than itself. Conversely, a negative edge from node C to node D implies that C considers D 's status to be lower than its own. We use the fairness score as a proxy for status assessment.

By leveraging *status theory*, we refine the construction of the perturbation space in adversarial settings. Status theory posits that in a hierarchical or trust-based social system, the direction and sign of edges should follow certain consistency rules—such as trust flowing from lower to higher status nodes and distrust flowing in the opposite direction. In this context, we identify and prioritize structural regions in the graph where edge injections that violate these principles are most likely to introduce significant distortions to the trust propagation process.

Specifically, we focus on analyzing injected edges that contradict status theory assumptions, as these violations tend to disrupt the fairness and consistency constraints encoded in models like REV2. Empirical results show that such violations are highly conducive to adversarial influence, making them strategic points for trust manipulation. This theory-guided selection of perturbation candidates allows us to construct a more targeted and effective adversarial space, increasing the likelihood of successful trust score degradation or inversion under limited perturbation budgets. Key violations include:

- A positive edge directed from a high-fairness node to a low-fairness node.
- A negative edge directed from a low-fairness node to a high-fairness node.

Table 3.3: Compliance with Status Theory in Trust Propagation on REV2. This table shows whether different tie structures conform to status theory under the fairness metric in REV2. A checkmark (✓) indicates compliance. Incorporating status theory helps guide the construction of more effective perturbation spaces.

Categories	Positive edge	Negative edge
From low fairness to high fairness	✓	✗
From high fairness to low fairness	✗	✓

Table 3.4 presents a comprehensive summary of *preferential paths* derived from empirical analysis of propagation sensitivity in the REV2 model. While FGA allows for a tractable theoretical exploration of how strong and weak ties influence trust propagation, REV2 introduces significantly higher complexity. Even when simplified in implementation, REV2 maintains an iterative co-evolution process of fairness and goodness scores, coupled with user behavior-aware updates. This layered architecture makes it difficult to derive analytical expressions for how specific structural perturbations affect trust outcomes. Consequently, theoretical modeling of attacker preference becomes highly challenging under the REV2 framework.

To address this, we adopt an empirical approach to investigate which tie structures are more susceptible to adversarial exploitation. The concept of *preferential paths* is introduced to characterize tie patterns that, when perturbed, consistently yield more effective trust manipulation. These paths are identified through repeated attack simulations and statistical evaluation of adversarial impact on trust scores. Unlike conventional strong ties—defined via homophily, interaction intensity, or frequency—*preferential paths* are empirically emergent: they are the subset of strong ties most frequently and effectively exploited by attackers.

The experiments in Table 3.4 are conducted under a trust promotion setting, where adversaries aim to artificially inflate the reputation of untrustworthy entities. The black heart symbol (♥) marks tie configurations that exhibit high attack preference in this scenario, i.e., those that lead to significant increases in trust scores when manipulated.

It is important to note that REV2’s design inherently reduces propagation bias across the network. In theory, any structurally valid edge—particularly those classified as strong ties—could participate in trust propagation. This suggests that trust updates in REV2 are not inherently skewed toward specific pathways. However, our findings show that even in this low-bias setting, attackers still exhibit clear structural preferences, selectively exploiting certain strong-tie paths over others. This asymmetry in attack behavior—despite a uniformly designed propagation mechanism—highlights a hidden layer of vulnerability. Uncovering and characterizing these preferential paths thus forms the foundational motivation of this study, bridging the gap between structural tie definitions and adversarial realities in graph-based trust systems.

Table 3.4: Empirical Analysis of Tie-Driven Attack Preference in REV2. ♥ indicates strong adversarial preference under trust promotion, i.e., these paths are particularly conducive to attack effectiveness.

Categories	Passive Attack (+1)	Positive Attack (−1)
Pred(T)	♥	♡
Succ(T)	♥	♡
Succ(Pred(T))	♡	♥
Pred(Succ(T))	♥	♡
Succ(Succ(T))	♡	♥
Pred(Pred(T))	♥	♡

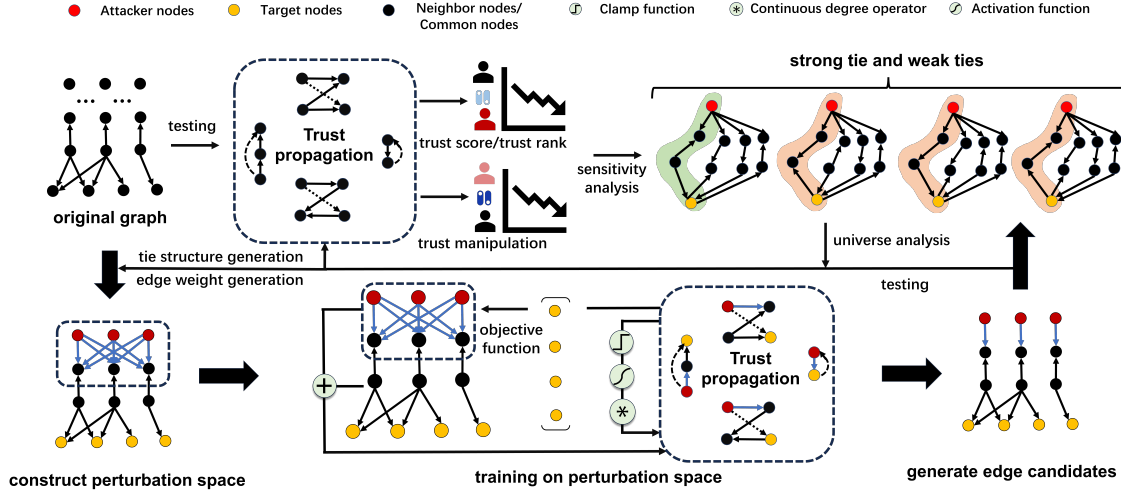


Figure 3.6: General architecture of the proposed method across trust models.

3.5 Adversarial Optimization Framework

The proposed *Vicinage Attack* architecture, depicted in Figure 3.6, operationalizes through two core analytical modules: *gradient-based sensitivity analysis* and *perturbation universe decomposition*. This dual mechanism enables discriminative treatment of preferential attack paths versus non-critical tie structures within FGA and REV2 trust propagation models. The framework’s novelty lies in its *adversarial transferability* - perturbation spaces constructed during attack generation enable efficient model retraining through adversarial curriculum learning.

3.5.1 Unified Adversarial Optimization Formulation

The sensitivity module identifies ϵ -critical paths based on the magnitude of the trust gradient, defined as:

$$\Gamma(v) = \left\| \frac{\partial \mathcal{F}_T(v)}{\partial E(S, I)} \right\|_2 \cdot \sigma_{ST}^{-1} \quad (3.15)$$

Here, $\mathcal{F}_T(v)$ is the trust score function for the target node v , $E(S, I)$ denotes the set of possible edges connecting attacker nodes S to intermediary nodes I , and σ_{ST} captures

the social intimacy between attacker nodes S and target nodes T . The inverse scaling by σ_{ST} reflects the intuition that more distant (i.e., less socially intimate) pathways are more susceptible to indirect attack manipulations. The term $\Gamma(v)$ thus measures the normalized gradient sensitivity of node v with respect to structural perturbations initiated by S . A path $p \in \mathcal{P}(S, T)$ is considered critical if its sensitivity exceeds a threshold:

$$\mathcal{P}_{\text{crit}} = \{p \in \mathcal{P}(S, T) \mid \Gamma(p) \geq \tau_{\Gamma}\} \quad (3.16)$$

where $\mathcal{P}(S, T)$ denotes the set of all paths from the attacker set S to the target set T , and $\tau_{\Gamma} = \mu_{\Gamma} + 2\sigma_{\Gamma}$ is a data-driven threshold derived from the mean and standard deviation of Γ across paths. To systematically model the attack surface, we define the perturbation space as a 4-tuple:

$$\Phi_{S,T}(B) = \langle \mathcal{V}, \mathcal{E}, \mathcal{W}, \mathcal{C} \rangle \quad (3.17)$$

In this formulation: $\mathcal{V} = S \cup I(T) \cup T$ denotes the vertex set, consisting of the attacker nodes S , the set of identified intermediaries $I(T)$ (which may vary depending on the topology), and the target nodes T . $\mathcal{W} : \mathcal{E} \rightarrow \{-1, +1\}$ is the edge weight function assigning polarity to each edge, representing trust (+1) or distrust (-1). $\mathcal{C} : \mathcal{E} \rightarrow \mathbb{R}^+$ defines the injection cost model, assigning a non-negative cost to each edge injection.

In the case of overlapping perturbation bases B_i and B_j , their intersection is resolved via sensitivity dominance:

$$\Phi_{S,T}(B_i \cap B_j) = \{e \in \Phi_{S,T}(B_i) \mid \Gamma(e) \geq \max(\tau_i, \tau_j)\} \quad (3.18)$$

This ensures that only the most influential edges (according to Γ) are retained in overlapping regions. For 2-hop attack scenarios, $\mathcal{E} = \bigcup_{i=0}^3 \Phi_{S,T}(B_i)$ defines the edge universe, formed by the union of all perturbation bases B_i . We define the following perturbation subspaces:

$$\Phi_{S,T}(B_0) = S \times \text{Succ}(\text{Pred}(T)) \quad (3.19)$$

$$\Phi_{S,T}(B_1) = S \times \text{Pred}(\text{Succ}(T)) \quad (3.20)$$

$$\Phi_{S,T}(B_2) = S \times \text{Succ}(\text{Succ}(T)) \quad (3.21)$$

$$\Phi_{S,T}(B_3) = S \times \text{Pred}(\text{Pred}(T)) \quad (3.22)$$

Each subspace B_i represents a specific semantic relationship between the attacker nodes S and the extended 2-hop neighborhood of target nodes T , constructed through directed compositions of predecessor (**Pred**) and successor (**Succ**) operations. These candidate regions collectively form the adversarial perturbation space, within which optimal edge injection strategies are designed.

The discrete combinatorial optimization challenge can be addressed through measure-theoretic relaxation of the perturbation space. We redefine the binary edge selection variable B as a continuous probability measure $\tilde{B} \in [0, 1]^{|\Phi_{S,T}|}$, where each element $\tilde{b}_{ij}$ represents the Bernoulli probability of edge (i, j) being injected. This induces a probabilistic graph:

$$\mathcal{M}_G = \{\tilde{G} = (V, E \cup \tilde{B}) \mid \tilde{B} \in [0, 1]^{|\Phi_{S,T}|}, \|\tilde{B}\|_0 \leq k\} \quad (3.23)$$

The initial probability configuration follows a uniform distribution $\tilde{B}^{(0)} = \frac{1}{2}\mathbf{1}$, ensuring maximum entropy before optimization begins. The discrete edge injection process is relaxed into a probabilistic framework via Bernoulli sampling, resulting in a stochastic continuous graph traversal formulation:

$$\begin{aligned} \min_{\tilde{B}} \mathbb{E}_{\epsilon \sim \text{Ber}(\tilde{B})} \left[\sum_{t \in T} \mathcal{F}_T(t, G \oplus \epsilon) \right] &+ \lambda \text{KL}(\tilde{B} \parallel \frac{k}{|\Phi_{S,T}|}) \\ \text{s.t. } \|\tilde{B}\|_1 &\leq k + \delta \\ \text{rank}(\nabla^2 \mathcal{L}) &\geq d_{\min} \end{aligned} \quad (3.24)$$

Here, the attack loss is computed as the expected degradation of the trust metric across all target nodes. The Kullback-Leibler (KL) divergence regularizer encourages a sparse and uniform perturbation distribution over the candidate edge space, promoting edge diversity and ensuring that the attack pattern remains stealthy and indistinguishable from natural structural variations. The hyperparameter λ governs the trade-off between attack potency and stealthiness. The total injection probability is softly constrained by δ , a tolerance term reflecting the expected deviation from a hard budget. The curvature rank constraint further regularizes the search space to prevent gradient collapse in flat perturbation manifolds.

Trust Demotion vs. Trust Promotion.

The above formulation assumes a *trust demotion* scenario, wherein the attacker aims to reduce the trust scores of the target set T . This is the most common setting in adversarial trust attacks, especially for scenarios such as reputation degradation or fraud exposure.

However, the same formulation can be adapted for *trust promotion* by inverting the optimization objective. The promotion case targets elevating trust scores of malicious users, useful in backdoor or collusive Sybil attacks:

$$\begin{aligned} \max_{\tilde{B}} \mathbb{E}_{\epsilon \sim \text{Ber}(\tilde{B})} \left[\sum_{t \in T} \mathcal{F}_T(t, G \oplus \epsilon) \right] - \lambda \text{KL}(\tilde{B} \| \frac{k}{|\Phi_{S,T}|}) \\ \text{s.t. } \|\tilde{B}\|_1 \leq k + \delta \end{aligned} \quad (3.25)$$

In both cases, the KL divergence plays a crucial role in spreading the injected probabilities across the perturbation subspace, helping the attack remain undetected while maintaining structural plausibility. While the formulation supports adaptive and high-dimensional perturbation modeling, in practical implementations we simplify the formulation by approximating the expectation via Monte Carlo sampling and retaining only the top- k perturbation candidates for final injection.

To enable gradient flow through discrete topology, we develop smooth degree operators using adaptive activation functions:

$$\begin{aligned} d_{\text{in}}(v) &= \sum_{(u,v) \in E} w_{uv} + \sum_{(u,v) \in \Phi_{S,T}} \phi_{\beta}(\tilde{b}_{uv}) \\ d_{\text{out}}(v) &= \sum_{(v,u) \in E} w_{vu} + \sum_{(v,u) \in \Phi_{S,T}} \phi_{\beta}(\tilde{b}_{vu}) \end{aligned} \quad (3.26)$$

where the activation function ϕ_{β} combines tanh and sigmoid characteristics:

$$\phi_{\beta}(x) = \frac{\tanh(\beta x) + \tanh(\beta(1-x))}{2 \tanh(\beta/2)} \quad (3.27)$$

The adaptive smoothing parameter β follows an annealing schedule:

$$\beta^{(t)} = \beta_{\min} + (\beta_{\max} - \beta_{\min}) \cdot \frac{t}{T} \quad (3.28)$$

3.5.2 From Discrete to Continuous Graph Perturbation in FGA

The NP-hard combinatorial optimization in Eq. (2) can be relaxed using a probabilistic formulation:

$$\Phi_{S,T}(\tilde{B}) = \mathbb{E}_{\epsilon \sim \text{Bernoulli}(\tilde{B})} [\Phi_{S,T}(\epsilon)] \quad (3.29)$$

Here, $\tilde{B} \in [0, 1]^{|V| \times |V|}$ is a probabilistic perturbation matrix, where each entry represents the probability of adding or removing an edge. The expectation is taken over binary perturbation matrices ϵ sampled from $\text{Bernoulli}(\tilde{B})$. This induces a continuous perturbed graph:

$$\tilde{G} = G \oplus \Phi_{S,T}(\tilde{B}) = \tilde{B} \odot (\mathbf{1} - A) \odot (-1) + A \odot W \quad (3.30)$$

In this expression, A is the original adjacency matrix, W is the corresponding edge weight tensor, and $\mathbf{1}$ is an all-one matrix. The operator $\odot$ denotes element-wise multiplication. The term $\tilde{B} \odot (\mathbf{1} - A) \odot (-1)$ applies soft perturbations to non-edges for negative edge demotion scenarios, while $A \odot W$ preserves the structure of the original graph. The operator $\oplus$ denotes the addition of the perturbation to the original graph. To ensure the perturbation remains bounded, we control the graph deviation using the Frobenius norm:

$$\|\tilde{G} - G\|_F \geq \gamma(k), \quad \text{where } \gamma(k) \sim \mathcal{O}(k^{1/2}) \quad (3.31)$$

To control the structural distortion introduced by continuous perturbations $\|\tilde{G} - G\|_F$, we adopt two complementary strategies: differentiable degree preservation and attribute clamping.

1. Differentiable Degree Preservation. We approximate the degree constraint in a differentiable form to ensure that node degrees do not deviate excessively during perturbation. The relaxed degree of a target node v is computed as:

$$\tilde{d}(v) = d(v) + \sum_{u \in S \cup I} \sigma(\tilde{B}_{uv}) \cdot \tanh'(\tilde{B}_{uv}) \quad (3.32)$$

where $\sigma(\cdot)$ is the sigmoid function, $\tanh'(\cdot)$ denotes the derivative of the hyperbolic tangent function, and $S \cup I$ represents the source and intermediary nodes that contribute to perturbations on v . A concrete example of this degree adjustment is shown for node m_1 in Figure 3.7. Its perturbed degree evolves as:

$$\tilde{d}(m_1) = 1 + \sum_{i=1}^3 \frac{\tanh(\tilde{B}_{a_i m_1}) + 1}{2} \approx \begin{cases} 3.19 & (\text{initial } \tilde{B} = 0.5) \\ 3.55 & (\text{optimized } \tilde{B} = 0.9, 0.9, 0.8) \end{cases} \quad (3.33)$$

This demonstrates how controlled perturbation can be used to gradually increase the degree while maintaining differentiability for gradient-based optimization.

2. Attribute Clamping. In addition to structural constraints, we also regularize feature-space distortion by clamping the predicted trust scores within a statistically reasonable range:

$$\mathcal{F}_T(v) = \text{Clip} \left(f_T(v, \tilde{G}), [\mu_f - 3\sigma_f, \mu_f + 3\sigma_f] \right) \quad (3.34)$$

Here, $f_T(v, \tilde{G})$ denotes the trust score of node v in the perturbed graph, and $[\mu_f \pm 3\sigma_f]$ defines the clipping range based on the empirical mean and standard deviation of predictions. This prevents the optimization process from generating unrealistically high or low scores due to structural over-perturbation.

Our attack optimization leverages constrained gradient descent within the probabilistic perturbation continuous graph, extending prior work on adversarial graph optimization [31, 32, 33, 34, 35]. We assume that the small-world network property [131] guarantees an optimization convergence within $\lceil \log_2 \text{diam}(G) \rceil$ iterations, where $\text{diam}(G)$ denotes the graph diameter. We formalize the optimization process as:

$$\tilde{B}^{(t+1)} = \Pi_{\mathcal{M}_k} \left(\tilde{B}^{(t)} - \eta_t \nabla_{\tilde{B}} \mathcal{L}_{\text{attack}} \right) \quad (3.35)$$

Here, $\eta_t = \eta_0 e^{-t/\tau}$ is a decaying learning rate. $\Pi_k(\cdot)$ keeps the top- k values in $\tilde{B}$ and zeros out the rest. We initialize with $\tilde{B}^{(0)} = \frac{1}{2} \mathbf{1}_{S \times T}$. The final perturbation matrix $\tilde{B}^*$ exhibits ϵ -discrete convergence:

$$\|\tilde{B}^* - \lfloor \tilde{B}^* \rfloor\|_F \leq \epsilon \quad (\epsilon = 0.05) \quad (3.36)$$

Edge selection follows quantile-based thresholding:

$$B^* = \left\{ (i, j) \mid \tilde{B}_{ij}^* > F_{\tilde{B}^*}^{-1}(1 - k/|S||T|) \right\} \quad (3.37)$$

where F^{-1} denotes the quantile function of perturbation probabilities. Under the unit cost assumption ($C_{ij} = 1, \forall(i, j)$), the perturbation task can be formalized as a discrete optimization:

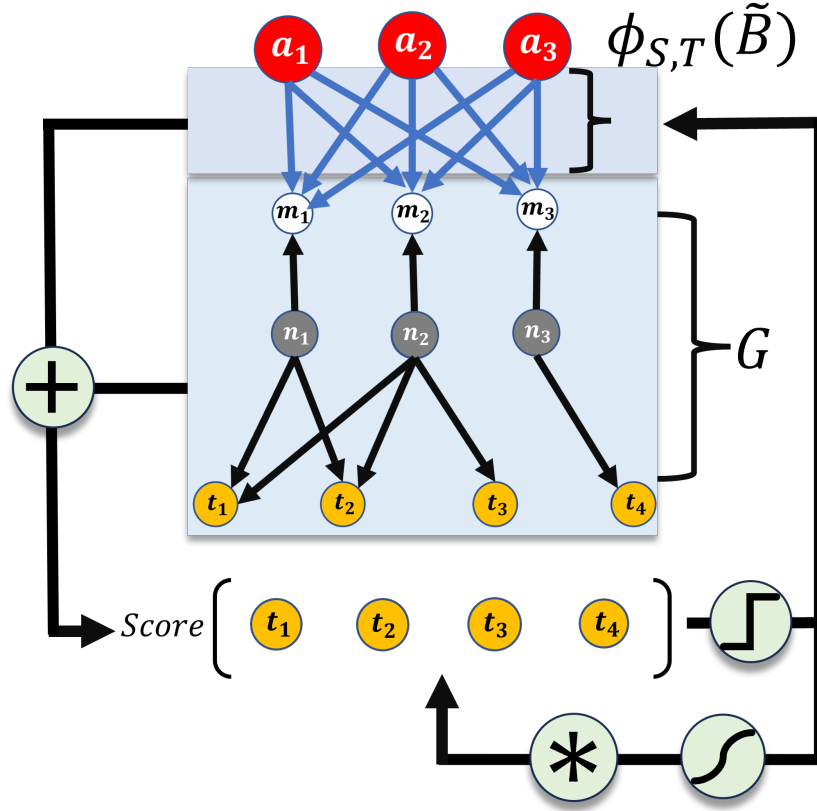
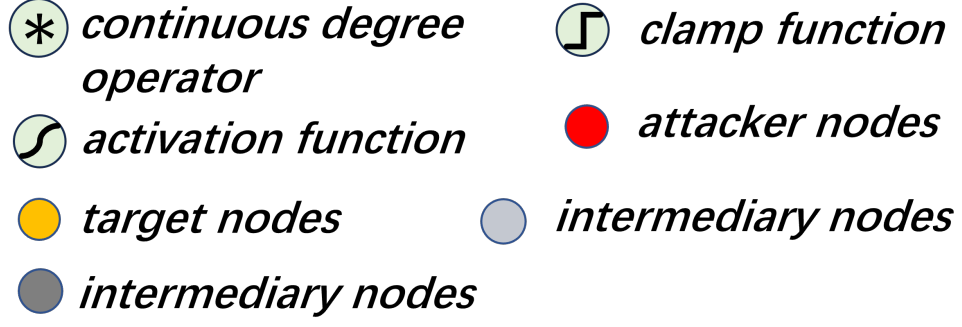


Figure 3.7: Overall architecture of vicinage-attack on FGA

$$\begin{aligned}
& \underset{B \subseteq \Phi_{S,T}}{\text{minimize}} && \sum_{t \in T} [\mathcal{F}_T(t, G) - \mathcal{F}_T(t, G_\epsilon)] + \lambda \|B\|_F^2 \\
& \text{subject to} && G_\epsilon = G \oplus \Phi_{S,T}(B) \\
& && \|B\|_0 \leq k, \quad B \in \{0, 1\}^{|\Phi_{S,T}|}
\end{aligned} \tag{3.38}$$

Here, $\oplus$ denotes the edge injection operator, and $\mathcal{F}_T(\cdot)$ represents the trust scoring function. The optimal attack B^* is selected by computing gradients over the edge subspace:

$$B^* = \text{Top}_k \left(\frac{\partial \mathcal{L}_{\text{attack}}}{\partial \Phi_{S,T}} \odot \sigma(\Gamma(\Phi_{S,T})) \right) \tag{3.39}$$

where $\sigma(\cdot)$ is a sigmoid-based selection gate, and $\Gamma(\cdot)$ denotes edge activation scores.

To enhance gradient flow and allow differentiable attack planning, we extend the discrete formulation to a continuous graph space. The relaxed optimization objective is given by:

$$\begin{aligned}
& \min_{\tilde{B}} && \sum_{v \in T} \mathcal{F}_T(v) + \lambda_1 \|\tilde{B}\|_1 + \lambda_2 \text{TV}(\tilde{G}) \\
& \text{s.t.} && \|\tilde{B}\|_0 \leq k, \quad \text{rank}(\tilde{G}) \leq \text{rank}(G) + \Delta_r
\end{aligned} \tag{3.40}$$

Here, $\tilde{B} \in [0, 1]^{|\Phi_{S,T}|}$ represents continuous edge weights, and $\text{TV}(\tilde{G}) = \sum_{i,j} |\nabla_{ij} \tilde{G}|$ is the total variation regularization term enforcing topological smoothness. The rank constraint is introduced to ensure the low-rank nature of the perturbed graph, drawing from matrix completion theory [132].

We implement a practical gradient-based indirect Sybil attack strategy via Algorithm 1, which operates in a continuous relaxation space and iteratively updates edge probabilities. One major challenge in continuous relaxation is the explosion in candidate edge space. Table 3.5 illustrates the difference in edge counts between perturbation candidates and the original graph, highlighting scalability and feasibility

Algorithm 1: Indirect Sybil Attack: *vicinage-attack***Input:** Function $\mathcal{FGA}$ with defined operators,Continuous graph $\Phi_{S,T}(\tilde{B}) + G$,Budget k , objective function $\mathcal{F}(\tilde{\mathbf{B}})$ **Output:** Selected top- k largest entriesInitialize $\tilde{\mathbf{B}} \leftarrow 0.5$;Set $n \leftarrow 5$, $L \leftarrow 12$;**for** $i \leftarrow 1$ *to* n **do** **for** $j \leftarrow 1$ *to* L **do** Execute $\mathcal{FGA}(\Phi_{S,T}(\tilde{B}) + G)$; Update $\tilde{\mathbf{B}} \leftarrow \tilde{\mathbf{B}} - \eta \frac{\partial \mathcal{F}(\tilde{\mathbf{B}})}{\partial \tilde{\mathbf{B}}}$;Sort $\tilde{\mathbf{B}}^*$ and select the top- k largest entries ;**return** Selected top- k largest entries ;

concerns.

The computational complexity comprises three principal components:

1. FGA Iteration Complexity:

$$\mathcal{C}_{\text{FGA}} = O(|\mathcal{T}|d_{\text{avg}}|V|) \quad (3.41)$$

where d_{avg} is the average degree, leveraging sparse matrix operations.Table 3.5: Impact of edge count differences between the original graph G and perturbation spaces, illustrating potential model distortion.

Datasets	$\#\Phi_{S,T}(\tilde{B}_i)$ -Edges	$\#G$ -Edges
Bitcoin-Alpha	[100, 000, 150, 000]	24, 000+
Bitcoin-OTC	[120, 000, 200, 000]	35, 000+

2. Continuous Graph Projection Cost:

$$\mathcal{C}_{\text{proj}} = O(L|V| \log |V| + |E|) \quad (3.42)$$

using Fibonacci heap-based priority queues for projection.

3. Thresholding Overhead:

$$\mathcal{C}_{\text{select}} = O(|\mathcal{T}||V| \log(|\mathcal{T}||V|)) \quad (3.43)$$

The composite complexity becomes:

$$\mathcal{C}_{\text{total}} = O(n(\mathcal{C}_{\text{proj}} + \mathcal{C}_{\text{FGA}}) + \mathcal{C}_{\text{select}}) \quad (3.44)$$

3.5.3 From Discrete to Continuous Ranking Manipulation in REV2

Adversarial manipulation in trust-based ranking systems such as REV2 presents a particularly challenging global optimization problem. Unlike traditional classification tasks, trust prediction involves recursive fairness propagation and ranking adjustments over signed graphs, making direct discrete attack strategies computationally infeasible at scale. In this context, our objective is to approximate global perturbation effects using limited adversarial resources.

To address this, we adopt a continuous relaxation framework inspired by adversarial retraining and transfer attack principles. The idea is to iteratively learn perturbation patterns that generalize across structurally similar subspaces—thereby extending the efficacy of local gradient feedback beyond isolated graph snapshots. By embedding differentiable operators into REV2’s propagation mechanism, we enable ranking-based adversarial objectives to be optimized through gradient descent, facilitating scalable and transferable manipulation strategies across different network topologies and target configurations.

This continuous formulation bridges discrete trust perturbation and differentiable optimization, allowing attackers (or defenders) to explore a much larger perturbation space without violating hard structural constraints. The resulting framework provides a tractable approach to understanding and mitigating adversarial preference in complex propagation models such as REV2. The constructed perturbation spaces enable two key reuse scenarios:

1. **Adversarial Retraining:** In adversarial retraining, we aim to improve the robustness of the trust prediction model f by incorporating adversarially perturbed graphs during training. Here, $B \sim \mathcal{U}(\Phi_{S,T})$ denotes a set of adversarial edges sampled from the perturbation space $\Phi_{S,T}$ under a uniform distribution. The objective is to minimize the expected loss $\mathcal{L}_{\text{unsup}}$ over perturbed graphs $G \oplus B$, where G is the original signed graph.

$$\min_{B \sim \mathcal{U}(\Phi_{S,T})} \mathcal{L}_{\text{unsup}}(f(G \oplus B)) \quad (3.45)$$

Adversarial retraining in our setting denotes the structural-level reinforcement of adversarial configurations, where perturbation samples are accumulated and reused to progressively shape a harder input distribution for evaluation.

2. **Transfer Attacks.** We treat perturbation spaces as adversarial sets that can be composed and manipulated algebraically. Given a source domain Φ_{src} and an initial target domain Φ_{dst}^0 , we construct a fused target space via convex interpolation:

$$\Phi_{dst} = \alpha \Phi_{src} + (1 - \alpha) \Phi_{dst}^0 \quad (3.46)$$

Alternatively, we may define operations between spaces using set-theoretic ana-

logues:

$$\Phi_{\text{union}} = \Phi_{\text{src}} \cup \Phi_{\text{dst}}^0 \quad (3.47)$$

$$\Phi_{\text{novel}} = \Phi_{\text{src}} \setminus \Phi_{\text{dst}}^0 \quad (3.48)$$

$$\Phi_{\text{core}} = \Phi_{\text{src}} \cap \Phi_{\text{dst}}^0 \quad (3.49)$$

These formulations allow flexible reuse of adversarial knowledge across domains with different node sets and topologies. For example, removing overlapping perturbations Φ_{core} from the source yields a “novel” subspace Φ_{novel} that contains transferable, previously unseen attack strategies. We define the universal adversarial reservoir as:

$$\mathbb{U}_{\text{adv}} = \bigcup_{\text{domain } i} \Phi_i \quad (3.50)$$

which serves as a transferable knowledge base across multi-graph environments. This universe can be sampled, filtered, or projected into task-specific spaces to support robust trust prediction under structural uncertainty. This reusable attack paradigm provides promised efficiency gain over single-use strategies in our experiments.

The REV2 trust evaluation framework generates node rankings through non-differentiable sorting operations:

$$\mathcal{R}(p(G)) = \text{argsort} \left(\{s_i\}_{i=1}^{|U|} \right) \quad (3.51)$$

where $s_i \in \mathbb{R}$ represents the trust score of node u_i . Direct gradient-based optimization of rank positions is prohibited by the discontinuous nature of the argsort operator. We adopt and extend the *SoftRank* methodology [133] through probabilistic rank expectation. For node pair (u_i, u_j) , define pairwise rank dominance probability:

$$\pi_{ij} = \Pr(s_i > s_j) = \frac{1}{1 + e^{-\beta(s_i - s_j)}} \quad (3.52)$$

where $\beta > 0$ controls approximation sharpness. The expected rank position becomes:

$$\mathbb{E}[r_i] = \sum_{j \neq i} \pi_{ij} + \frac{1}{2} \quad (3.53)$$

The attack goal is formulated as a constrained optimization over rank expectations:

$$\begin{aligned} \min_{\tilde{B}} \mathcal{F}_{\text{atk}} \left(\sum_{u \in T} \mathbb{E}[r_u(\tilde{G})] \right) + \lambda \|\tilde{B}\|_1 \\ \text{where } \mathcal{F}_{\text{atk}}(x) = \begin{cases} x & \text{(Demotion attack)} \\ -x & \text{(Promotion attack)} \end{cases} \end{aligned} \quad (3.54)$$

The complete attack pipeline integrates:

1. **Probabilistic Edges Initialization:** $\tilde{B}^{(0)} = \frac{1}{2} \mathbf{1}$

2. **Expectation Propagation:**

$$\frac{\partial \mathbb{E}[r_i]}{\partial s_j} = \begin{cases} \beta \pi_{ij}(1 - \pi_{ij}) & i \neq j \\ -\beta \sum_{k \neq i} \pi_{ik}(1 - \pi_{ik}) & i = j \end{cases} \quad (3.55)$$

3. **Gradient Composition:**

$$\frac{\partial \mathcal{L}}{\partial \tilde{B}} = \sum_{u \in T} \frac{\partial \mathcal{F}_{\text{atk}}}{\partial \mathbb{E}[r_u]} \cdot \frac{\partial \mathbb{E}[r_u]}{\partial s_u} \cdot \frac{\partial s_u}{\partial \tilde{B}} \quad (3.56)$$

4. **Projected Optimization:**

$$\tilde{B}^{(t+1)} = \Pi_{\Phi_{S,T}} \left(\tilde{B}^{(t)} - \eta_t \nabla_{\tilde{B}} \mathcal{L} \right) \quad (3.57)$$

To solve the optimization problem efficiently, we employ gradient descent, as outlined in prior studies [33, 11, 127, 128]. The initialization of $\tilde{B}$ is set to 0.5, and it is iteratively updated according to the following update rule:

$$\tilde{B} \leftarrow \Pi_{[0,1]} \left(\tilde{B} - \alpha \nabla_{\tilde{B}} \mathcal{F}_{\text{atk}}(\mathcal{C}_{\tilde{B}}) \right), \quad (3.58)$$

Algorithm 2: Gradient-based Indirect Attack

Input: Constructed perturbation space $\Phi_{S,T}(\tilde{B})$, original graph

$G_0 = (U_0, R_0, W_0)$, budget k , iteration n , l , target node T ,

learning rate α

Output: Candidate edges $G_\epsilon - G_0$

Initialize $\tilde{\mathbf{B}} = 0.5$

for $i = 1$ *to* n **do**

for $j = 1$ *to* l **do**

for *each node* u *in* U_0 **do**

 Update $g(u)$ via $In(\Phi_{S,T}(\tilde{B}) \cup G_0)$

$g(u) = \Pi_{[0,1]}g(u)$

end

for *each edge* (u, v) *in* $\Phi_{S,T}(\tilde{B}) \cup G_0$ **do**

 Update $r(u, v)$

$r(u, v) = \Pi_{[0,1]}r(u, v)$

end

for *each node* u *in* U_0 **do**

 Update $f(u)$ via $Out(\Phi_{S,T}(\tilde{B}) \cup G_0)$

$f(u) = \Pi_{[0,1]}f(u)$

end

end

for *each node* t *in* T **do**

$\mathcal{C} \leftarrow \mathcal{C} + \mathcal{R}_{\tilde{B}}^t(\Phi_{S,T}(\tilde{B}) \cup G_0)$

end

for *each edge* $\tilde{\mathbf{b}}_i$ *in* $\tilde{\mathbf{B}}$ **do**

$\tilde{\mathbf{b}}_i \leftarrow \Pi_{[0,1]}(\tilde{\mathbf{b}}_i - \alpha \nabla_{\tilde{\mathbf{b}}_i} \mathcal{F}_{atk}(\mathcal{C}_{\tilde{B}}))$

end

end

Select Candidate Edges: $\bigcup_1^k (i^*, j^*) = \sigma_{\Phi}^k(B^*)$

where α is the learning rate, and $\mathcal{C}_{\tilde{B}}$ denotes the cumulative trust rank of target nodes derived from $\Phi_{S,T}(\tilde{B}) \cup G_0$. The projection operator $\Pi_{[0,1]}(\cdot)$ ensures that all elements of $\tilde{B}$ remain within the valid range $[0, 1]$. This projection mechanism is also employed during updates to fairness, goodness, and reliability scores to maintain numerical stability. With successive optimization steps, $\tilde{B}$ gradually converges towards binary values, where elements approach either 1 or 0, indicating an increased likelihood of edge injection. Upon convergence, the gradient distribution, sorted in descending order, is utilized to prioritize and select the most effective candidate edges for injection.

To summarize the proposed adversarial strategy, Algorithm 2 outlines the complete *vicinage*-attack method. The algorithm operates in a two-loop structure: an outer loop handling the optimization iterations and an inner loop performing iterative learning updates. The parameters n and l correspond to the number of iterations in the outer and inner loops, respectively. The central principle behind our adversarial attack approach is to treat the perturbation space as a hyperparameter and compute the gradient of the attacker’s objective with respect to it, thereby refining the attack through structured optimization.

3.6 Evaluation

3.6.1 Dataset Description for Weighted Directed Signed Trust Networks

To evaluate the proposed adversarial trust manipulation framework, we conduct experiments on two real-world signed and directed networks obtained from the Stanford Network Analysis Project (SNAP: <http://snap.stanford.edu/index.html>). These networks are derived from Bitcoin-based web-of-trust platforms and have been widely adopted in prior studies on trust evaluation and adversarial graph learning.

The two datasets are:

- **Bitcoin-Alpha:** This network comprises 3,783 nodes and 24,186 edges.
- **Bitcoin-OTC:** This network consists of 5,881 nodes and 35,592 edges.

In both datasets, each node represents a user participating in cryptocurrency transactions, and each directed edge corresponds to a trust rating given by one user to another. The original trust ratings range from -10 (complete distrust) to +10 (complete trust). For experimental consistency, these scores are normalized to the range of $[-1, 1]$, reflecting signed edge weights suitable for trust propagation modeling.

These signed graphs serve as a representative testbed for our analysis, as they naturally encode both positive and negative trust signals and exhibit structural complexity that poses challenges for adversarial robustness. Their real-world origin and widespread adoption in related literature make them an ideal benchmark for evaluating the effectiveness of edge-level perturbation strategies in signed trust networks.

3.6.2 Baseline Attack Strategies for Trust Manipulation in Signed Graphs

To comprehensively evaluate the effectiveness of our proposed attack strategy, we compare it against three widely recognized adversarial baselines:

1. Random attack: This baseline injects k edges randomly from the perturbation space:

$$B_{\text{rand}} = \text{Sample}_k \left(\Phi_{S,T} \setminus E; p = \frac{1}{|\Phi_{S,T}|} \right) \quad (3.59)$$

Edges are selected uniformly without regard to structural properties or propagation effects.

2. Indirect attack: Based on the brute-force strategy introduced by [23], this method evaluates the influence of every candidate edge on the trust score of the target set

and greedily selects the top- k most effective edges. The procedure is summarized in Algorithm 3:

3. Modified indirect attack: This variant improves the brute-force attack by dividing the total injection budget across multiple iterations:

$$B = \bigcup_{t=1}^n B_t \quad \text{where } |B_t| = \lfloor k/n \rfloor \quad (3.60)$$

At each epoch, edge influence is recalculated on the updated graph $G_{t-1} = (V, E \cup \bigcup_{i=1}^{t-1} B_i)$. The top- $|B_t|$ influential edges are selected per epoch, ensuring progressive adaptation of attack behavior.

Following empirical tuning, we set $n = 5$ for FGA and $n = 4$ for REV2. Even with this adjustment, the brute-force nature leads to significant computational overhead. Our proposed method leverages differentiable optimization techniques to solve the attack objective efficiently. To ensure fairness, we limit optimization steps in our method to 5 iterations for FGA and 4 for REV2, aligned with the epoch settings of the adaptive baselines.

Algorithm 3: Greedy Edge Selection

Input: Perturbation space Φ , attack loss function $\mathcal{L}_{\text{attack}}$, selection budget k

Output: Selected edge set B

Initialize $B \leftarrow \emptyset$;

for $i \leftarrow 1$ **to** k **do**

$e^* \leftarrow \arg \max_{e \in \Phi \setminus B} \Delta \mathcal{L}_{\text{attack}}(B \cup \{e\})$;
 $B \leftarrow B \cup \{e^*\}$;

end

return B ;

3.6.3 Evaluating Strong Tie Effectiveness and Prerequisite Constraints in FGA

Figure 3.8 demonstrates the superior performance of the proposed *vicinage*-attack when compared with several state-of-the-art adversarial strategies, including the *Indirect*-attack and the *Modified Indirect*-attack. While the latter two baseline methods exhibit localized strengths, they fail to fully address the combinatorial optimization complexities inherent in trust-based network manipulation. These methods generally rely on brute-force search or staged perturbation, which, although effective in specific cases, are not scalable and do not capture the intricate dependencies that shape trust propagation in signed networks.

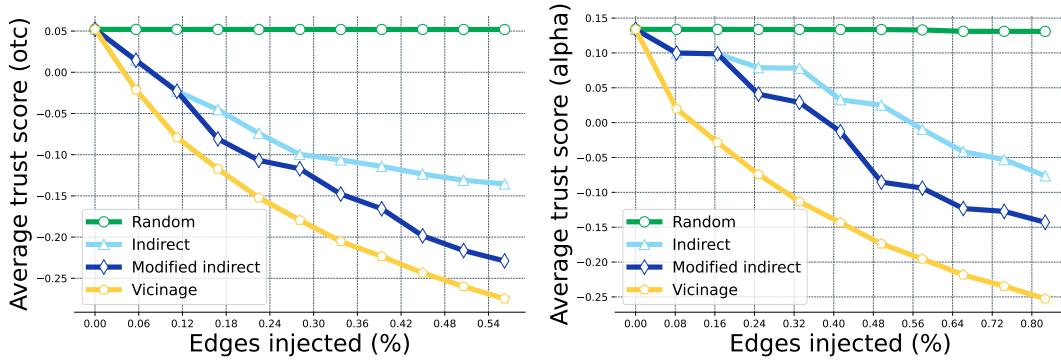


Figure 3.8: Performance comparison between *vicinage*-attack and baseline methods.

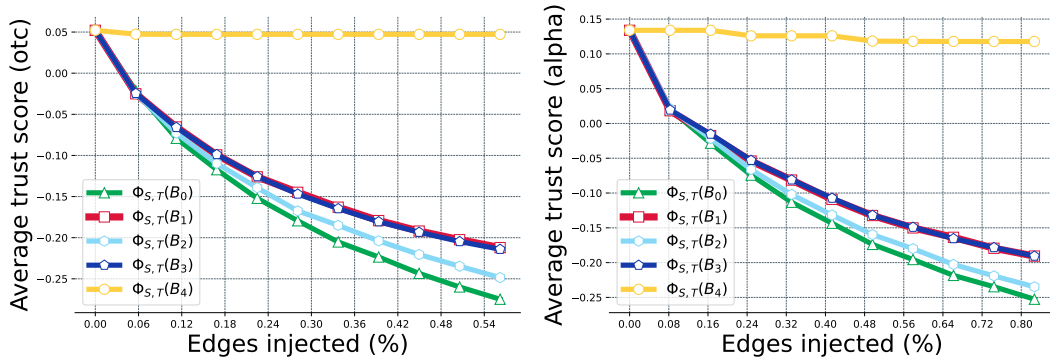


Figure 3.9: Comparative analysis of attack impacts on strong and weak ties.

By contrast, the *vicinage*-attack method adopts a gradient-based optimization framework that leverages continuous perturbation modeling and tie-structure awareness to identify highly impactful attack vectors. This approach enables a more comprehensive and adaptive exploration of the perturbation space, significantly enhancing the adversarial effect on the FGA trust scores. As shown in the results, the *vicinage*-attack yields more substantial trust distortion with fewer injected edges, validating its superior effectiveness.

To further investigate the mechanisms through which structural vulnerabilities influence adversarial effectiveness, we analyze the performance of attacks executed within different perturbation subspaces of $\Phi_{S,T}$. Specifically, we evaluate perturbation results over $\Phi_{S,T}(B_0)$, $\Phi_{S,T}(B_1)$, $\Phi_{S,T}(B_2)$, and $\Phi_{S,T}(B_3)$, as illustrated in Figure 3.9. These represent distinct structural layers derived from the 2-hop connectivity taxonomy introduced in earlier sections. To validate our hypothesis regarding the relative importance of strong versus weak ties, we construct a complementary perturbation space defined as:

$$\Phi_{S,T}(B_4) = \Phi_{S,T}(B_1) \cup \Phi_{S,T}(B_2) \cup \Phi_{S,T}(B_3) - \Phi_{S,T}(B_0)$$

The set $\Phi_{S,T}(B_4)$ contains edge candidates that do not qualify as strong ties, and is therefore expected to represent predominantly weak ties. When the *vicinage*-attack is applied to this subset, the resulting perturbations yield minimal trust distortion. This observation confirms the hypothesis that weak ties, although numerous, are structurally less effective for adversarial manipulation in FGA. In contrast, perturbations targeting strong ties—especially those in $\Phi_{S,T}(B_0)$ —consistently produce the most significant degradation in trust scores. This comparison is critical for understanding adversarial efficiency: it highlights that not all structurally valid perturbations contribute equally to trust score distortion. Rather, those concentrated in regions with high tie strength and centrality play a disproportionately large role in facilitating indirect Sybil attacks. Therefore, identifying and securing these strong-tie zones is essential for enhancing the robustness of trust systems.

Chapter 3. Strong-Tie Exploitation in Node Labeling: A Sybil Attack Perspective on Trust Prediction

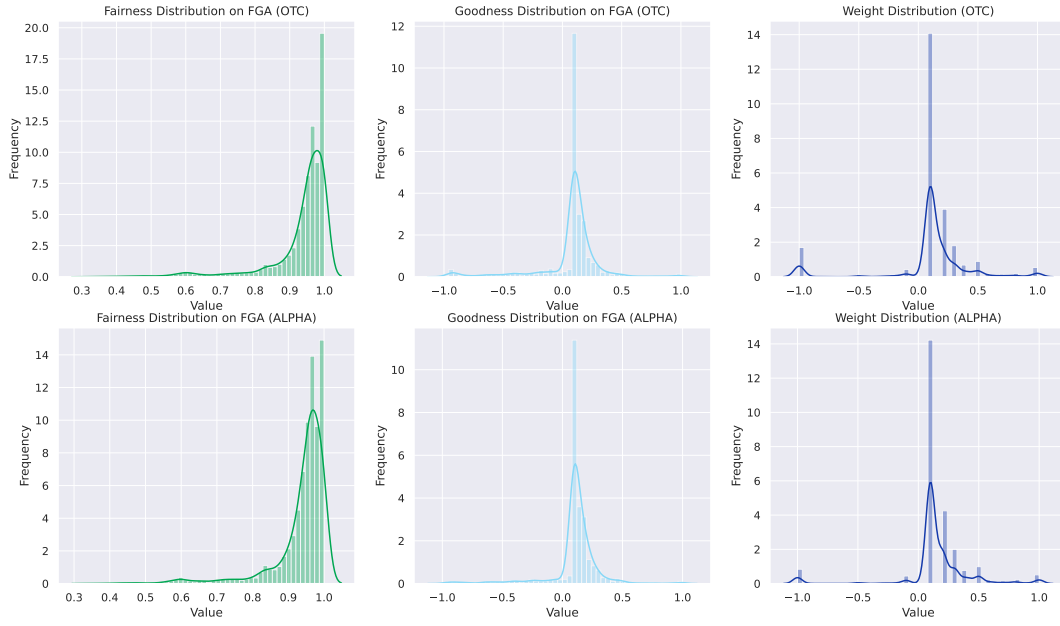


Figure 3.10: Fairness and goodness distribution in FGA alongside edge weight analysis.

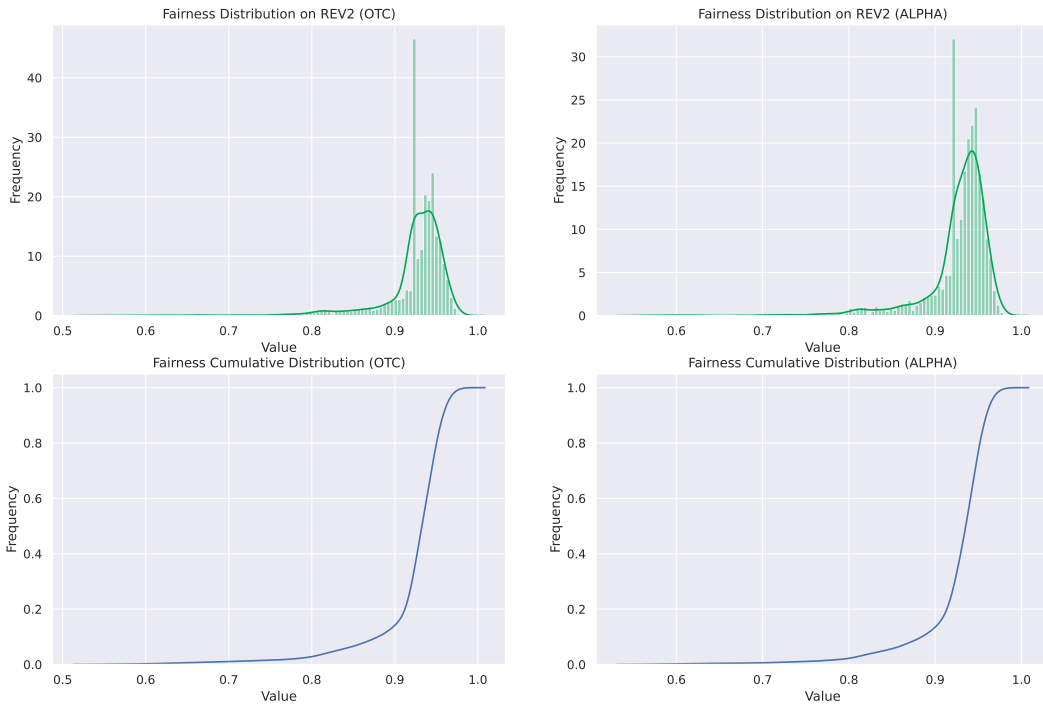


Figure 3.11: Distribution of fairness scores within the REV2 framework.

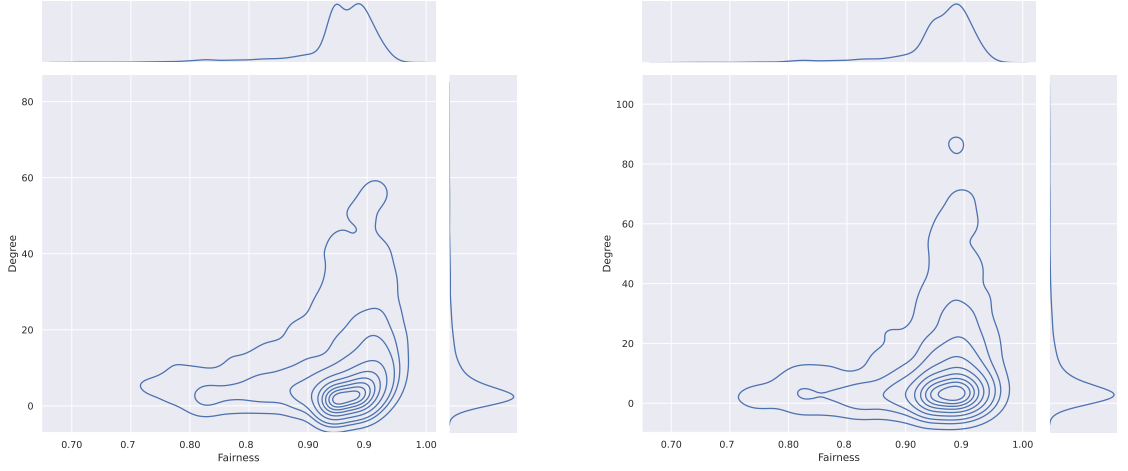
(a) Joint distribution in *bitcoin-otc*.(b) Joint distribution in *bitcoin-alpha*.

Figure 3.12: Joint distribution analysis of fairness and degree in REV2.

Importantly, the experiments also underline the critical role of *polymorphic strong ties* in the propagation of adversarial influence. These structurally significant ties serve as conduits for trust propagation, and their manipulation leads to widespread score distortion. This suggests that tightly knit, high-impact subgraphs—often centered on strong ties—are key vulnerabilities that indirect Sybil attackers can exploit to undermine the integrity of trust systems. Such findings shift the attention from purely structural centrality to functionally influential pathways.

Our analysis indicates that Strong Tie II becomes ineffective when the weight of the injected edge is set to -1 . We investigate the underlying reasons for this observation using statistical analysis. For Strong Tie I, Case I is defined as $f(s)w_0 < g(q) < w_2$, while Case II requires $g(q) > \max(w_2, f(s)w_0)$ and $w_0 < \frac{2w_2-1}{f(s)}$. This can be approximated as $g(q) > w_2$ and $w_0 < \frac{2w_2-1}{f(s)}$. Given the dominant distribution trends, the requirement simplifies to $w_0 < \frac{2w_2-1}{f(s)}$. This condition is more likely to hold when $w_0 = -1$, thus explaining why Strong Tie I remains viable under such settings.

For Strong Tie II, Case I is defined by the condition $f(p)w_0 < g(s) < w_0$. This

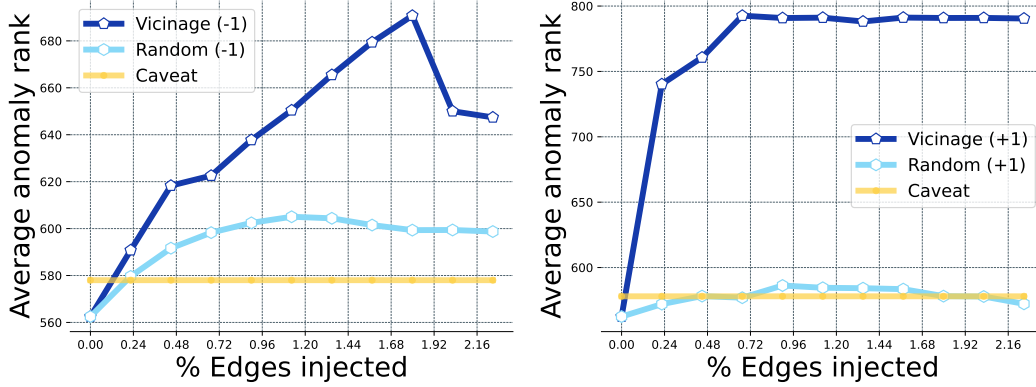


Figure 3.13: Performance evaluation of vicinage-attack on bitcoin-otc dataset.

constraint cannot be satisfied if $w_0 < 0$, where w_0 denotes the weight of the injected edge. Meanwhile, Case II is characterized by $g(q) > \max(w_0, f(s)w_0)$ and $w_0 > \frac{1}{2-f(s)}$. Figure 3.10 presents the distributions of fairness and goodness values in FGA, alongside the distribution of edge weights. The majority of fairness values fall within the range $(0.9, 1)$, while goodness values predominantly lie in the interval $(0.5, 0.25)$. Likewise, edge weights exhibit a similar distribution in the range $(0.5, 0.25)$. Given that satisfying Case II requires $w_0 > \frac{1}{2-f(s)}$, it follows that w_0 must be close to 1. However, this condition contradicts the assumption of $w_0 = -1$, making the realization of Strong Tie II infeasible under these settings.

3.6.4 Empirical Evaluation and Preferential Path Discovery in REV2

In the REV2 experiments, we assess the impact of the *vicinage*-attack by analyzing how trust propagates throughout the network. Specifically, we focus on nodes exhibiting low fairness scores as our primary targets, under the assumption that such nodes may seek to evade classification as anomalies or unfair entities. As a result, adversarial agents might attempt to artificially boost the perceived trustworthiness

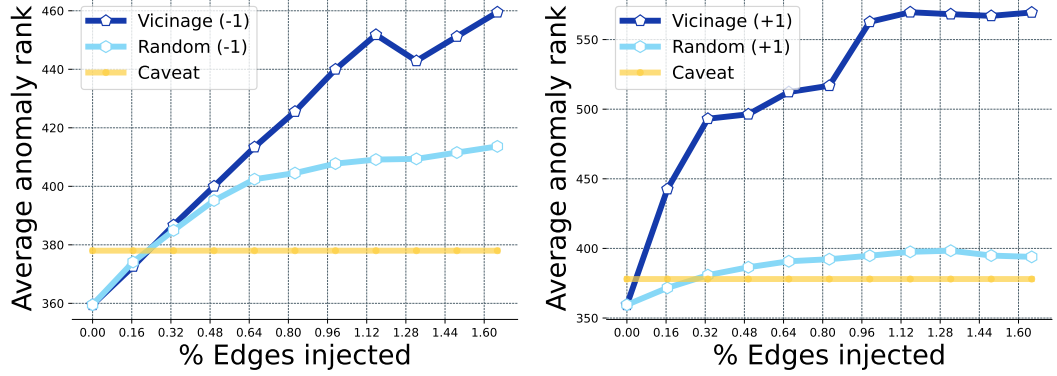


Figure 3.14: Baseline comparison of vicinage-attack on bitcoin-alpha dataset.

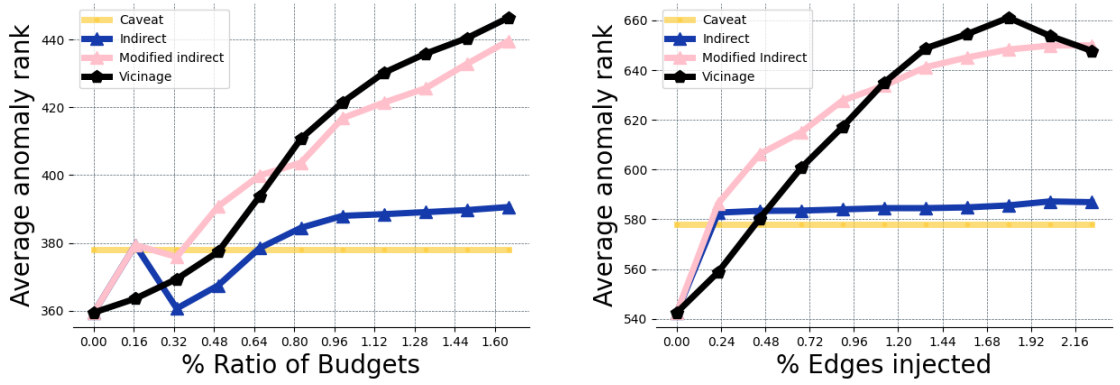


Figure 3.15: Comparative study of vicinage-attack across bitcoin-alpha and bitcoin-otc datasets.

of these target nodes.

Figure 3.11 illustrates the cumulative distribution function (CDF) of fairness values. The fairness distribution shows a steady increase within the interval $(0.8, 0.9)$, followed by a sharp rise in the range $(0.9, 1)$. This pattern reveals a potential vulnerability that attackers could exploit to manipulate trust rankings. In our experiment, we define the set of target nodes T as those satisfying $f(t) < 0.88$, while high-fairness nodes S are characterized by $f(s) > 0.96$ and degree values exceeding a predefined threshold, $d(s) > d_{\text{threshold}}$. Additionally, Figure 3.12 presents the distribution of degree

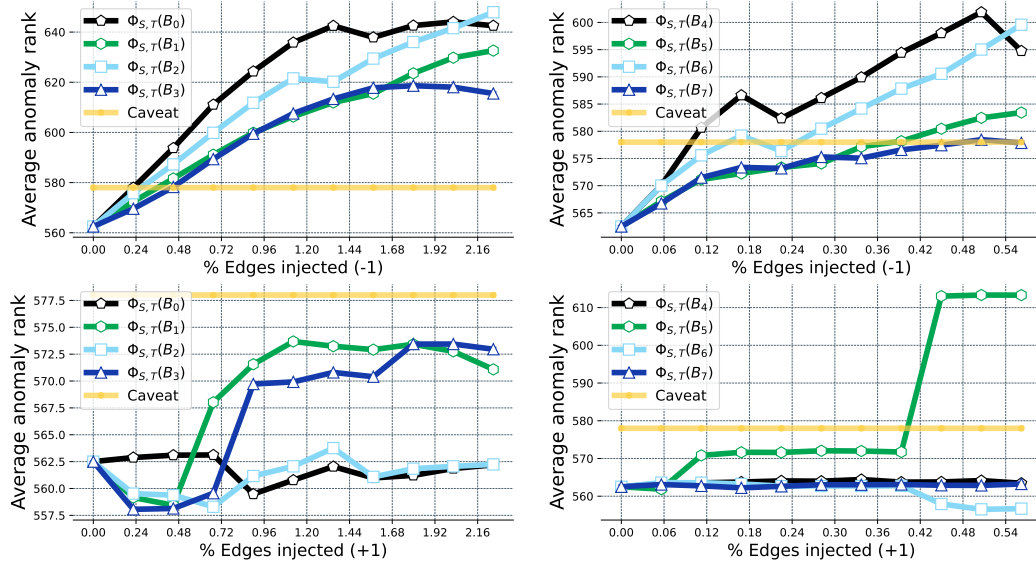


Figure 3.16: Preferential path analysis over 2-hop connections in bitcoin-otc.

values, demonstrating that node degrees predominantly cluster around the threshold value $d_{\text{threshold}}$. This observation further reinforces the feasibility of adversarial trust manipulation in the network.

In this section, we investigate the presence of *preferential paths* within the REV2 framework. The perturbation space is composed of both *positive attacks* and *passive attacks*, incorporating interactions with both 1-hop and 2-hop neighbors. As depicted in Figures 3.13, 3.14, and 3.15, the *vicinage*-attack consistently outperforms baseline methods in selecting the most effective perturbation edges. The *caveat line* in these figures denotes the 10% of nodes with the lowest fairness scores, which are considered anomalous or untrustworthy.

For different values of n , we observe the following trends: 1. When $n < 4$: The *vicinage*-attack significantly surpasses both the indirect and modified indirect attacks, demonstrating superior efficiency with fewer iterations. 2 When $n = 4$: The performance of the modified indirect attack approaches that of the *vicinage*-attack. However, despite this improvement, it remains more computationally expensive. 3.

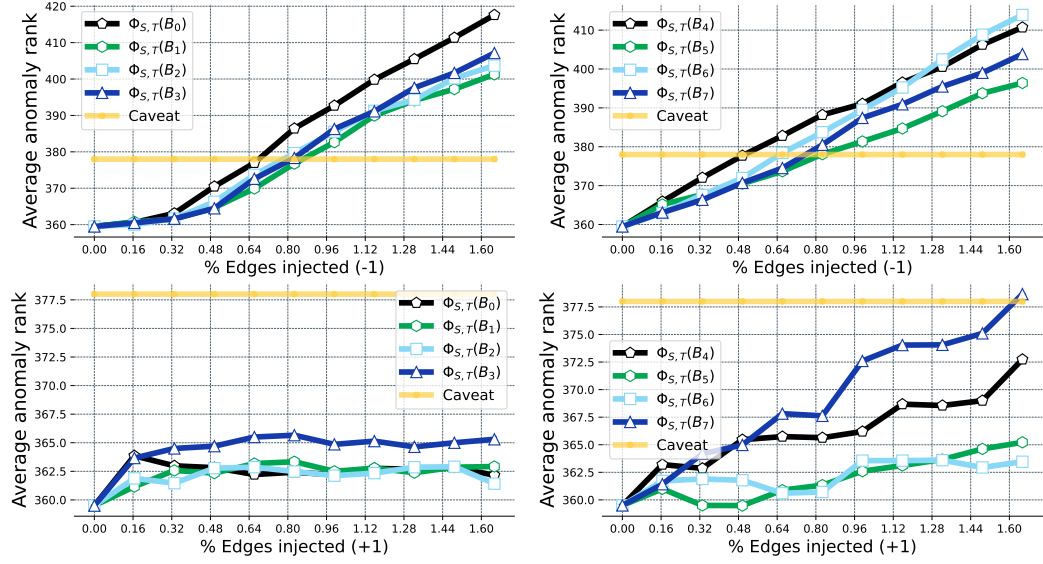


Figure 3.17: Preferential path analysis over 2-hop cases in bitcoin-alpha.

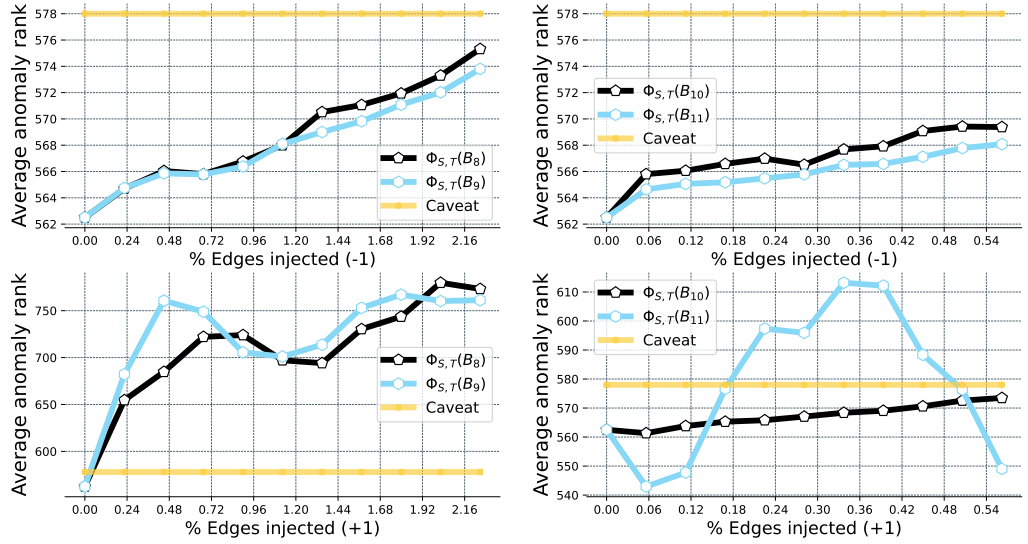


Figure 3.18: 1-hop preferential path analysis on bitcoin-otc dataset.

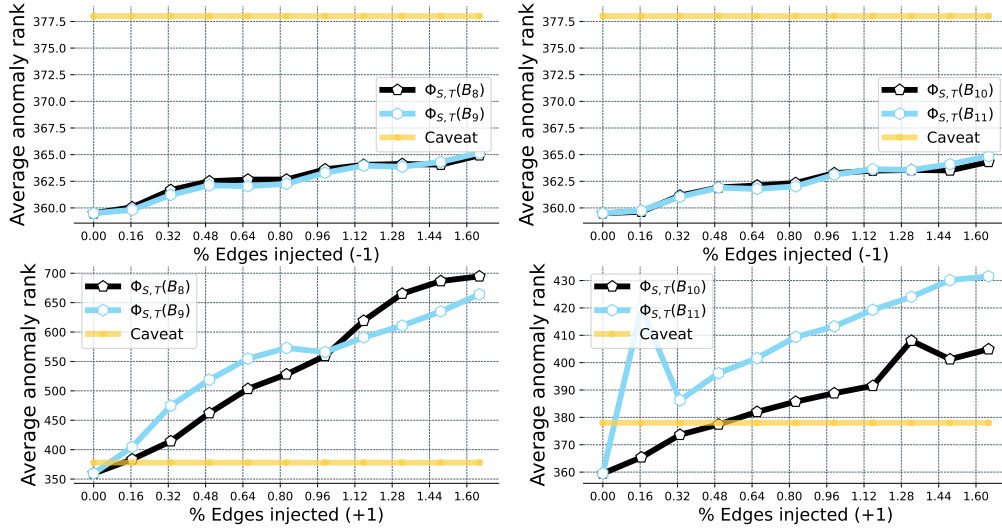


Figure 3.19: 1-hop preferential path analysis on bitcoin-alpha dataset.

When $n > 4$: The modified indirect attack becomes increasingly inefficient, requiring additional time without noticeable performance gains, further highlighting the advantage of the *vicinage*-attack.

By leveraging a gradient-based approach, the *vicinage*-attack method facilitates a more effective search within the perturbation space. This efficiency allows for faster identification of *strong ties* and *preferential paths*, which play a pivotal role in the success of adversarial attacks against trust prediction systems.

Figures 3.16 to 3.19 further illustrate our findings. We define four primary perturbation spaces: $\Phi_{S,T}(B_0)$, $\Phi_{S,T}(B_1)$, $\Phi_{S,T}(B_2)$, and $\Phi_{S,T}(B_3)$. These spaces represent edges linking high-fairness nodes S to the sets $Succ(Pred(T))$, $Pred(Succ(T))$, $Succ(Succ(T))$, and $Pred(Pred(T))$, respectively. Additionally, we derive four subset perturbation spaces, $\Phi_{S,T}(B_4)$ through $\Phi_{S,T}(B_7)$, defined as:

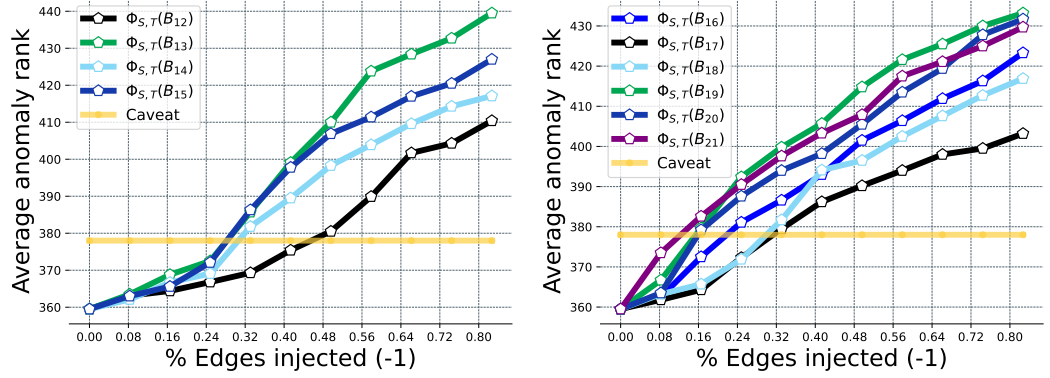


Figure 3.20: Supplementary 2-hop preferential path analysis for bitcoin-alpha using constructed perturbation spaces.

$$\begin{cases} \Phi_{S,T}(B_4) = \Phi_{S,T}(B_0) - \Phi_{S,T}(B_1) - \Phi_{S,T}(B_2) - \Phi_{S,T}(B_3); \\ \Phi_{S,T}(B_5) = \Phi_{S,T}(B_1) - \Phi_{S,T}(B_0) - \Phi_{S,T}(B_2) - \Phi_{S,T}(B_3); \\ \Phi_{S,T}(B_6) = \Phi_{S,T}(B_2) - \Phi_{S,T}(B_0) - \Phi_{S,T}(B_1) - \Phi_{S,T}(B_3); \\ \Phi_{S,T}(B_7) = \Phi_{S,T}(B_3) - \Phi_{S,T}(B_0) - \Phi_{S,T}(B_1) - \Phi_{S,T}(B_2). \end{cases} \quad (3.61)$$

Furthermore, we introduce perturbation spaces for 1-hop neighbors, denoted as $\Phi_{S,T}(B_8)$ and $\Phi_{S,T}(B_9)$, where edges connect high-fairness nodes S to $Pred(T)$ and $Succ(T)$, respectively. We then derive two additional subsets, $\Phi_{S,T}(B_{10})$ and $\Phi_{S,T}(B_{11})$, as follows:

$$\begin{cases} \Phi_{S,T}(B_{10}) = \Phi_{S,T}(B_8) - \Phi_{S,T}(B_9); \\ \Phi_{S,T}(B_{11}) = \Phi_{S,T}(B_9) - \Phi_{S,T}(B_8). \end{cases} \quad (3.62)$$

Figure 3.20 extends the analysis by incorporating broader perturbation subsets from the universal perturbation space, defined as follows:

$$\left\{ \begin{array}{l} \Phi_{S,T}(B_{12}) = \Phi_{S,T}(B_1) \cup \Phi_{S,T}(B_2) \cup \Phi_{S,T}(B_3) - \Phi_{S,T}(B_0); \\ \Phi_{S,T}(B_{13}) = \Phi_{S,T}(B_0) \cup \Phi_{S,T}(B_2) \cup \Phi_{S,T}(B_3) - \Phi_{S,T}(B_1); \\ \Phi_{S,T}(B_{14}) = \Phi_{S,T}(B_0) \cup \Phi_{S,T}(B_1) \cup \Phi_{S,T}(B_3) - \Phi_{S,T}(B_2); \\ \Phi_{S,T}(B_{15}) = \Phi_{S,T}(B_0) \cup \Phi_{S,T}(B_1) \cup \Phi_{S,T}(B_2) - \Phi_{S,T}(B_3). \end{array} \right. \quad (3.63)$$

Additionally, further perturbation subsets are formulated to explore interactions between different spaces:

$$\left\{ \begin{array}{l} \Phi_{S,T}(B_{16}) = \Phi_{S,T}(B_2) \cup \Phi_{S,T}(B_3) - \Phi_{S,T}(B_0) - \Phi_{S,T}(B_1); \\ \Phi_{S,T}(B_{17}) = \Phi_{S,T}(B_1) \cup \Phi_{S,T}(B_3) - \Phi_{S,T}(B_0) - \Phi_{S,T}(B_2); \\ \Phi_{S,T}(B_{18}) = \Phi_{S,T}(B_1) \cup \Phi_{S,T}(B_2) - \Phi_{S,T}(B_0) - \Phi_{S,T}(B_3); \\ \Phi_{S,T}(B_{19}) = \Phi_{S,T}(B_0) \cup \Phi_{S,T}(B_3) - \Phi_{S,T}(B_1) - \Phi_{S,T}(B_2); \\ \Phi_{S,T}(B_{20}) = \Phi_{S,T}(B_0) \cup \Phi_{S,T}(B_2) - \Phi_{S,T}(B_1) - \Phi_{S,T}(B_3); \\ \Phi_{S,T}(B_{21}) = \Phi_{S,T}(B_0) \cup \Phi_{S,T}(B_1) - \Phi_{S,T}(B_2) - \Phi_{S,T}(B_3). \end{array} \right. \quad (3.64)$$

These analyses provide a comprehensive examination of the perturbation space, illustrating the significance of *strong ties* and *preferential paths* in the context of adversarial attacks on trust prediction systems.

3.7 Application Scenarios of Trust Manipulation via Structural Preferences

The observed results affirm the hypothesis that strong ties in signed social networks play a disproportionate role in the network's vulnerability to Sybil attacks. The findings highlight the deceptive nature of these polymorphic strong ties, which can be manipulated to a far greater extent than weaker connections. This insight not only contributes to our understanding of adversarial dynamics in trust-based systems but

also points to a fundamental limitation in current defense mechanisms, which often overlook the complexity of these relationships.

Furthermore, the success of the *vicinage*-attack underscores the potential for advanced optimization techniques to exploit structural weaknesses in social networks. By leveraging continuous graph perturbation, gradient-driven edge selection, and stability-aware optimization, our method can be applied to a wide range of adversarial attack scenarios beyond the current study. This approach offers valuable insights for the design of more robust trust and security mechanisms in social networks, where the nature of relationships—especially those involving strong ties—must be carefully considered.

Our study reveals the presence of *preferential paths* within trust systems, demonstrating a significant bias in the way trust propagates across networks. This insight has crucial implications for improving the robustness of trust mechanisms, particularly in domains such as e-commerce, online review platforms, and decentralized peer-to-peer systems. Additionally, our findings underscore the necessity of *vulnerability assessment* to identify and mitigate adversarial threats that exploit *preferential paths*. Beyond adversarial applications, these paths can also be utilized in *trust chain discovery* to uncover implicit trust relationships within a network.

3.7.1 Assessing Structural Vulnerabilities in Trust Systems

Preferential paths in trust networks can function as potential weak points that adversaries may exploit. In graph-based applications, adversarial techniques have been widely studied to identify such vulnerabilities, including backdoor attacks in graph classification tasks [134, 135, 136, 137]. In the context of trust systems, adversaries can selectively target specific nodes labeled by administrators, strategically manipulating network structures to identify the most critical pathways for attack. Our proposed *vicinage*-attack serves as a proactive mechanism for stress-testing these

vulnerabilities.

As outlined in [70], three primary conditions define an effective adversarial attack: *stealthiness*, *consistency*, and *inconsistency*. We adapt these principles to the trust prediction domain: 1. *Stealthiness* ($\mathcal{S}(S, T)$): This property ensures that attacker nodes S are positioned several hops away from their target nodes T , allowing for an inconspicuous yet effective attack. Since only a minimal number of perturbations are introduced, these injected edges remain largely indistinguishable from normal interactions within the original network structure. 2. *Consistency* ($\mathcal{C}(G_0, G_\varepsilon)$): A successful attack should maintain stability in trust predictions for most non-target nodes, ensuring that modifications appear localized rather than widespread. While the attack follows an intentional strategy, it must remain indistinguishable from natural variations in trust propagation. 3. *Inconsistency* ($\mathcal{I}(G_0, G_\varepsilon)$): In contrast to consistency, this property ensures that the trust scores of the designated target nodes undergo significant changes post-attack. The goal is to either inflate or degrade the trustworthiness of specific entities while leaving the rest of the network largely unaffected.

By systematically evaluating these properties, the *vicinage*-attack provides a powerful tool for assessing trust system resilience, exposing vulnerabilities that might otherwise remain undetected. This evaluation framework serves as a foundation for designing robust defense mechanisms against adversarial manipulations in trust-based environments.

3.7.2 Discovering Trust Chains in Graph-Based Systems

In the context of trust systems, the perturbation space $\Phi_T(B) \in G_0$ represents a refined modification of the original network, specifically focusing on edges that establish indirect connections with target nodes. This concept aligns with the principles of *adversarial examples* [70, 71, 72, 73], wherein *preferential paths* play a pivotal role in the discovery of trust chains. By identifying and leveraging these pathways, it becomes

possible to reveal implicit trust relationships that might otherwise remain concealed within the network structure [68, 74, 75]. This methodology is particularly useful in scenarios where direct trust endorsements are absent, enabling a deeper assessment of indirect trust connections.

Furthermore, our findings indicate that trust chain discovery incorporating *preferential paths* extends beyond the static evaluation of trustworthiness. Rather than merely analyzing a fixed network snapshot, this approach enables continuous monitoring of dynamic trust evolution over time. By tracing the influence of trust propagation along *preferential paths*, it is possible to observe fluctuations in trust scores, detect emerging trends, and uncover potential adversarial manipulations within the network. This dynamic perspective provides a more comprehensive understanding of trust relationships, enhancing both predictive capabilities and security mechanisms in trust-based environments.

3.8 Challenges and Future Work in Structural Trust Manipulation

While the proposed *vicinage*-attack method significantly advances adversarial strategies in trust prediction systems, certain challenges remain. One primary limitation concerns scalability. As network size increases, the computational complexity of identifying *preferential paths* and executing *vicinage*-attacks grows substantially. This poses difficulties in applying the approach to large-scale social networks or online platforms containing millions of nodes and edges.

Additionally, the current study focuses specifically on the FGA and REV2 trust prediction frameworks, and the generalizability of the *vicinage*-attack across different trust prediction algorithms remains an open question. While the fundamental principles of preferential path exploitation may extend to other systems, further empirical

validation is needed to ensure adaptability and effectiveness in diverse trust modeling paradigms.

To address these limitations and contribute to the broader field of trust prediction security, future research will focus on two key aspects. First, a graph partitioning approach will be explored, segmenting large networks into subgraphs based on density and sparsity. This decomposition not only enhances scalability but also facilitates the development of explainable defense mechanisms by isolating distinct structural patterns and vulnerabilities within different subgraphs. Second, incorporating graph neural networks (GNNs) into trust system analysis may provide new insights into preferential behaviors under adversarial conditions. GNN-based models can potentially refine attack detection, improve robustness against adversarial perturbations, and enhance the interpretability of trust propagation. By leveraging deep learning frameworks, future work aims to advance both adversarial attack modeling and defense mechanisms in trust prediction, ultimately enabling more resilient and explainable security systems.

3.9 Summary

This chapter presents a unified summary of our research on structural trust manipulation in unsupervised learning, with a focus on how strong ties influence attack effectiveness in signed networks. We analyzed two representative models—FGA and REV2—and showed that their internal mechanisms can be exploited by adversaries to manipulate trust scores through carefully designed edge modifications. In the FGA part, we demonstrated that indirect Sybil attacks, often considered weak, can be highly effective when the attacker targets specific strong ties. We proposed the *vicinage*-attack, an optimization-based method that selects high-impact edges under budget constraints. Experimental results showed that this method outperforms existing baselines and confirms that strong ties play a central role in amplifying attack

impact. We also analyzed real-world trust data and found that while two types of strong ties are theoretically possible, only one type is likely to appear in practice due to fairness and weight limitations. This explains why some theoretical attacks are ineffective in real settings. In the REV2 part, due to its complexity and iterative design, we adopted an empirical approach to study attacker behavior. We found that although REV2 is designed to reduce bias in score updates, some tie structures are still preferred by attackers. These preferred connections, which we call *preferential path*, show clear patterns in how attacks are constructed. This demonstrates that, despite balanced trust propagation rules, the model’s robustness can be undermined by subtle perturbations targeting a few structurally pivotal edges. Our contributions can be summarized as follows:

- We reveal that strong ties significantly affect the success of trust attacks in both FGA and REV2.
- We propose a new attack framework that balances accuracy, efficiency, and structure awareness through continuous optimization.
- We provide detailed evaluations on real signed networks, showing clear distinctions between effective and ineffective perturbation strategies.
- We connect theoretical modeling with practical constraints, offering insights for future design of defense-aware trust models.

Overall, our work shows that structure plays a decisive role in trust prediction. Even small changes to certain links can cause major shifts in trust scores. Understanding and addressing these structural risks is essential for building safer systems in online platforms, financial applications, and decentralized environments.

Chapter 4

Adversarial Robustness in Node Labeling via Strong-Tie Adaptive Defense

4.1 Motivation

Despite significant advancements in graph-based machine learning, adversarial threats — particularly poisoning attacks — continue to pose critical challenges. Existing defenses often fall short in mitigating these attacks, especially when adversaries exploit structural vulnerabilities to manipulate label propagation and disrupt label prediction. This highlights the necessity of developing a more resilient defense framework that not only counteracts direct adversarial modifications but also enhances the robustness of trust-based graph queries in adversarial settings.

This study introduces the Graph Adaptive Neural Network (GANN), a novel defense mechanism designed to enhance the security of Graph Neural Networks (GNNs) by adapting their propagation dynamics based on structural insights. Unlike traditional defenses that reactively filter out adversarial perturbations, GANN proactively re-

structures propagation paths by leveraging trust and risk zones—adaptive partitions within the graph that modulate label propagation based on structural trustworthiness. By integrating these dynamic zones, GANN ensures stable and reliable propagation, even in the presence of adversarial manipulation, offering a fundamentally more resilient approach to graph security. By integrating these dynamic zones, GANN ensures stable and reliable propagation, even in the presence of adversarial manipulation, offering a fundamentally more resilient approach to graph security. Figure 4.1 illustrates the role of GANN in supporting secure graph queries under adversarial settings.

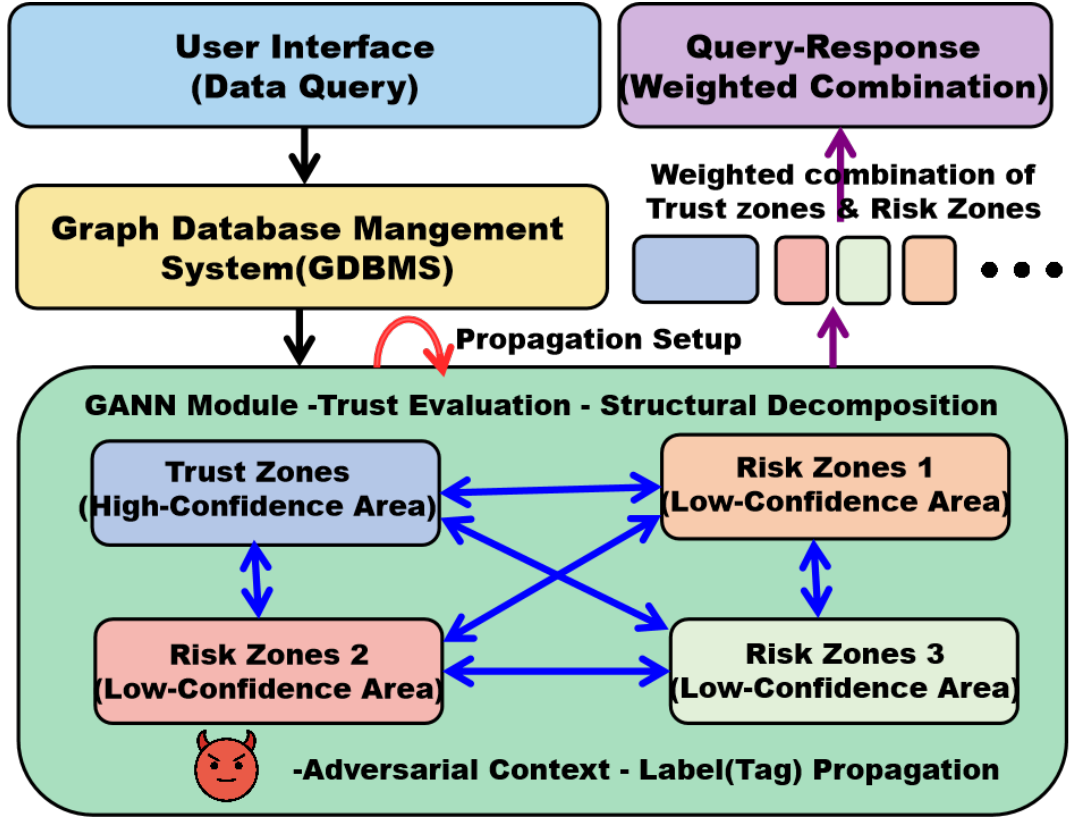


Figure 4.1: Role of GANN in secure graph queries under adversarial settings.

To further strengthen adversarial robustness, this study introduces Strong Tie Graphs (STiG)—a structural framework designed to characterize attacker preferences by dif-

differentiating between strong and weak ties in the graph topology. This differentiation allows GANN to identify and fortify key structural components that adversaries are most likely to exploit. By capturing the preferential behaviors of poisoning attacks and understanding their poisoning tendencies, STiG enhances the ability of GANN to preemptively mitigate adversarial risks.

By integrating structural graph analysis with adaptive defense mechanisms, this work provides a comprehensive solution for defending against structural poisoning attacks. The insights gained from this research contribute to a deeper understanding of adversarial strategies, enabling more robust trust-aware graph learning. Ultimately, GANN serves as a foundational advancement in securing graph intelligence, ensuring that graph-based queries remain stable and resilient in adversarial environments. This chapter is structured as follows:

- **Section II:** Presents two toy models that illustrate the intuition behind adaptive propagation
- **Section III:** Introduces the adaptive propagation mechanism and classifier definitions of GANN.
- **Section IV:** Introduces a defender-centric adversarial simulation framework under poisoning attacks.
- **Section V:** Describes how attacker preferences are evaluated over partitioned zones, with each zone assigned a trust confidence score via fuzzy theory.
- **Section VI:** Examines small and large trust zones, with a formal proof of provable trust guarantees.
- **Section VII:** Investigates how deviations in attacker preference reveal structural vulnerabilities and inform zone-specific mitigation strategies.

- **Section VIII:** Presents empirical evaluations of GANN’s effectiveness against poisoning attacks.
- **Section IX:** Concludes the chapter with key insights and future research directions.

4.2 Preliminary

To facilitate robust label propagation in graph-based learning models, we apply a transformation to the *adjacency matrix* A , converting it into a *normalized Laplacian adjacency matrix*:

$$\hat{A} = \tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}},$$

where $\tilde{A} = A + I_N$ represents the *self-looped adjacency matrix*, incorporating an identity matrix I_N to maintain self-connections. The matrix $\tilde{D}$ denotes the corresponding degree matrix. Node features $X \in \mathbb{R}^{N \times F}$ undergo iterative propagation using a message-passing framework, where a *projection function* Π ensures feature stability at each step. Ultimately, the aggregated features are transformed via a *Multilayer Perceptron (MLP)* to generate class predictions:

$$Z_{(i,:)} = f_{\text{MLP}}(X_{(i,:)}),$$

where $Z \in \mathbb{R}^{N \times C}$ represents the predicted class scores for N nodes across C possible categories.

Prototype Propagation Schemes: T-1ProP and T-2ProP

To illustrate distinct propagation behaviors, we introduce two simplified models—*Toy 1 Propagation and Prediction (T-1ProP)* and *Toy 2 Propagation and Prediction (T-2ProP)*—each demonstrating different update mechanisms.

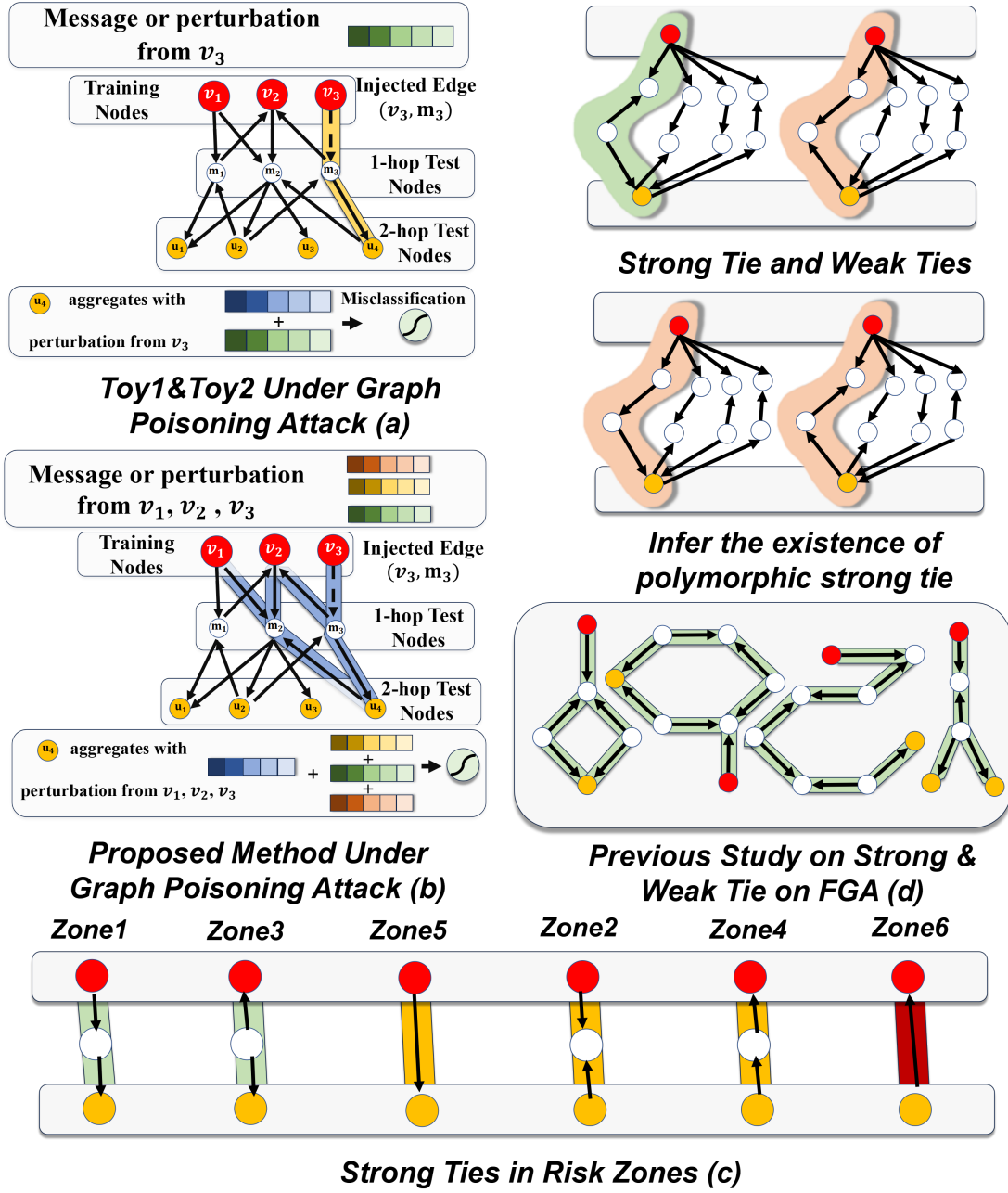


Figure 4.2: (a) Impact of poisoning attacks on Toy-1 & Toy-2. (b) Proposed method's response to attacks. (c) Trust confidence distribution in trust and risk zones. (d) Previous research on strong vs. weak ties in FGA, highlighting polymorphic strong ties.

In *T-1ProP*, features are processed separately for two directional propagation metrics, initialized as $\mathbf{m}_{i,\leftarrow}^{(0)} = \mathbf{m}_{i,\rightarrow}^{(0)} = X$. The iterative update equations for each metric are as follows:

$$\begin{aligned}\mathbf{m}_{i,\leftarrow}^{(k+1)} &= \text{AGG}_{\leftarrow}^{(k+1)} \left(\left\{ \mathbf{x}_j^{(k)}, \mathbf{x}_i^{(k)} : (j, i) \in E \right\} \right), \\ \mathbf{m}_{i,\rightarrow}^{(k+1)} &= \text{AGG}_{\rightarrow}^{(k+1)} \left(\left\{ \mathbf{x}_j^{(k)}, \mathbf{x}_i^{(k)} : (i, j) \in E \right\} \right),\end{aligned}$$

where AGG represents an aggregation function applied over neighboring node features. Feature stability is preserved through a projection function:

$$\mathbf{m}_{i,\rightarrow}^{(k)} = \Pi(\mathbf{m}_{i,\rightarrow}^{(k)}), \quad \mathbf{m}_{i,\leftarrow}^{(k)} = \Pi(\mathbf{m}_{i,\leftarrow}^{(k)}).$$

After k -step message aggregation, the final feature representation is computed as:

$$Z = f_{\text{MLP}}(\alpha \mathbf{m}_{i,\leftarrow}^{(k)} + \beta \mathbf{m}_{i,\rightarrow}^{(k)}),$$

where α and β are scalar coefficients regulating the contributions of both propagation directions.

In *T-2ProP*, the two directional metrics influence each other during updates, leading to an interdependent propagation process. The update rules remain structurally similar:

$$\begin{aligned}\mathbf{m}_{i,\leftarrow}^{(k+1)} &= \text{AGG}_{\leftarrow}^{(k+1)} \left(\left\{ \mathbf{x}_j^{(k)}, \mathbf{x}_i^{(k)} : (j, i) \in E \right\} \right), \\ \mathbf{m}_{i,\rightarrow}^{(k+1)} &= \text{AGG}_{\rightarrow}^{(k+1)} \left(\left\{ \mathbf{x}_j^{(k)}, \mathbf{x}_i^{(k)} : (i, j) \in E \right\} \right).\end{aligned}$$

However, in this model, the update steps incorporate cross-dependency:

$$\mathbf{m}_{i,\leftarrow}^{(k)} = \Pi(\mathbf{m}_{i,\rightarrow}^{(k)}), \quad \mathbf{m}_{i,\rightarrow}^{(k)} = \Pi(\mathbf{m}_{i,\leftarrow}^{(k)}).$$

After k -step message aggregation, the final feature representation is obtained through:

$$Z = f_{\text{MLP}}(\alpha \mathbf{m}_{i,\leftarrow}^{(k)} + \beta \mathbf{m}_{i,\rightarrow}^{(k)}).$$

These two toy models offer a foundational understanding of how directional propagation mechanisms influence trust prediction and classification tasks in graph-based

learning environments. To maintain numerical stability and ensure that the aggregated feature values remain within an appropriate range, a projection function Π is applied at each aggregation step. This function normalizes feature vectors to prevent extreme values and facilitate stable learning. Formally, Π is defined as:

$$\Pi(\mathbf{v}) = \frac{\mathbf{v}}{\|\mathbf{v}\| + \epsilon},$$

where $\mathbf{v}$ represents the aggregated feature vector, $\|\mathbf{v}\|$ denotes its norm, and ϵ is a small constant introduced to avoid division by zero. Figure 4.2 illustrates the impact of poisoning attacks on the Toy-1 and Toy-2 models, alongside the proposed approach. The figure highlights trust confidence across partitioned zones, providing a comparative analysis with previous research on strong and weak ties. This visualization underscores the advantages of incorporating structural-aware propagation mechanisms in mitigating adversarial effects while preserving trust reliability within graph-based systems.

4.3 Graph Adaptive Neural Network (GANN)

In this section, we introduce the Graph Adaptive Neural Network (GANN), an advanced framework that extends feature propagation to directed graphs by independently handling information from both in-neighbors and out-neighbors. The model integrates multi-directional propagation while employing a projection function Π to ensure numerical stability and effective feature aggregation throughout the iterative process.

4.3.1 Adaptive Propagation Strategy in GANN

For each node $i \in V$, GANN updates its feature representation iteratively by aggregating messages from its in- and out-neighbors while adaptively balancing their

contributions. The aggregation steps at iteration k are defined as follows:

$$\begin{aligned}\mathbf{m}_{i,\leftarrow}^{(k)} &= \text{AGG}_{\leftarrow}^{(k)} \left(\left\{ \mathbf{x}_j^{(k-1)}, \mathbf{x}_i^{(k-1)} : (j, i) \in E \right\} \right) \\ \mathbf{m}_{i,(\alpha,\beta)}^{(k)} &= \text{AGG}_{(\alpha,\beta)}^{(k)} \left(\alpha \mathbf{m}_{i,\leftarrow}^{(k)} + \beta \mathbf{m}_{i,\rightarrow}^{(k)} \right) \\ \mathbf{m}_{i,\rightarrow}^{(k)} &= \text{AGG}_{\rightarrow}^{(k)} \left(\left\{ \mathbf{x}_j^{(k-1)}, \mathbf{x}_i^{(k-1)} : (i, j) \in E \right\} \right),\end{aligned}$$

where $\text{AGG}_{\leftarrow}^{(k)}$ aggregates messages from in-neighbors (represented by $\hat{A}^T$), $\text{AGG}_{\rightarrow}^{(k)}$ aggregates messages from out-neighbors (represented by $\hat{A}$), and $\text{AGG}_{(\alpha,\beta)}^{(k)}$ adaptively combines both, weighted by parameters α and β that modulate the influence of incoming and outgoing edges, respectively. To ensure stability in feature propagation, the following projection updates are applied:

$$\mathbf{m}_{i,\leftarrow}^{(k)} = \Pi(\mathbf{m}_{i,(\alpha,\beta)}^{(k)}), \quad \mathbf{m}_{i,(\alpha,\beta)}^{(k)} = \Pi(\alpha \mathbf{m}_{i,\leftarrow}^{(k)} + \beta \mathbf{m}_{i,\rightarrow}^{(k)}), \quad \mathbf{m}_{i,\rightarrow}^{(k)} = \Pi(\mathbf{m}_{i,(\alpha,\beta)}^{(k)}).$$

Here, the projection function Π prevents numerical instability by keeping feature magnitudes within a controlled range.

After computing the aggregated messages, the node feature representation is updated using the combination function $\text{COM}^{(k)}$:

$$\mathbf{x}_i^{(k)} = \text{COM}^{(k)} \left(\mathbf{x}_i^{(k-1)}, \mathbf{m}_{i,\leftarrow}^{(k)}, \mathbf{m}_{i,(\alpha,\beta)}^{(k)}, \mathbf{m}_{i,\rightarrow}^{(k)} \right).$$

This function integrates information from previous node states and aggregated messages, often through weighted summation or concatenation, ensuring that local dependencies and global structural patterns are effectively captured. After k iterations of propagation, the final node embeddings are processed by a Multilayer Perceptron (MLP) for classification:

$$Z = f_{\text{MLP}} \left(\alpha \cdot \mathbf{m}_{i,\leftarrow}^{(k)} + \beta \cdot \mathbf{m}_{i,\rightarrow}^{(k)} \right).$$

This approach allows GANN to enhance trust-aware learning by dynamically adjusting propagation paths, ensuring robustness against adversarial attacks and effectively distinguishing between trusted and vulnerable regions within the graph.

4.3.2 Designing of GANN’s Adaptive Classifier

The Graph Adaptive Neural Network (GANN) introduces a classifier family $\mathcal{F}_{GANN}(\alpha, \beta)$, characterized by the adaptive propagation balance terms $\alpha, \beta \in [0, 1]$ with the constraint $\alpha + \beta = 1$. This formulation enables dynamic control over the influence of inbound and outbound information during feature aggregation. The propagation process, denoted as $\text{COM}^{(k)}$, iteratively refines node representations through structured aggregation. The final classification function is defined as:

$$\mathcal{F}_{GANN}(\alpha, \beta)(\mathbf{X}, \mathbf{A})_i = f_{\text{MLP}} \left(\alpha \cdot \mathbf{m}_{i,\leftarrow}^{(k)} + \beta \cdot \mathbf{m}_{i,\rightarrow}^{(k)} \right),$$

where $\mathbf{m}_{i,\leftarrow}^{(k)}$ and $\mathbf{m}_{i,\rightarrow}^{(k)}$ represent the aggregated feature embeddings obtained from incoming and outgoing edges, respectively. By integrating the adaptive propagation mechanism $\text{COM}^{(k)}$ with the expressive capability of a Multilayer Perceptron (MLP), $\mathcal{F}_{GANN}(\alpha, \beta)$ serves as a versatile and resilient classifier. This approach ensures robustness against adversarial attacks while allowing flexible adaptation to varying graph structures, making it particularly effective in mitigating structural perturbations in adversarial environments.

4.4 Defender-Centric Adversarial Simulation

Adversarial simulation plays a crucial role in graph-based security, enabling defenders to assess and enhance model robustness against adversarial threats. In a controlled *node classification task*, an attacker aims to degrade classification accuracy by injecting *poisoned samples* into the graph structure. The simulated adversary is assumed to have access to both the graph G and the classifier $f(w)$, operating within a white-box threat model where the graph structure and training set are fully observable. To maintain structural integrity, the simulation enforces constraints that prevent disruptive perturbations, such as node isolation. Notably, defenders conduct evaluations

based solely on the observed graph A^ζ , without reliance on an unaltered reference graph.

Building on insights gained through adversarial simulations, defenders employ strategic adaptations within the GANN framework to counteract adversarial manipulations. Two primary defense mechanisms are outlined: 1. **Static Strategy**: This approach maintains a fixed classifier $f(w) = \mathcal{F}_{GANN}(\hat{\alpha}, \hat{\beta})$ before and after the simulated attack. The defender optimizes the propagation parameters $(\hat{\alpha}, \hat{\beta})$ based on the observed impact of adversarial modifications, ensuring the model’s resilience through preemptively selected propagation schemes. By optimizing robustness in advance, this strategy minimizes the need for real-time adjustments while preserving model stability. 2. **Dynamic Strategy**: This adaptive approach introduces real-time countermeasures by reconfiguring the propagation scheme in response to detected adversarial perturbations. The defender dynamically updates the propagation parameters from $(\hat{\alpha}, \hat{\beta})$ to a refined configuration $(\hat{\alpha}', \hat{\beta}')$, effectively neutralizing poisoning attempts. The resulting adaptive classifier is expressed as: $f(w') = \mathcal{F}_{GANN}(\hat{\alpha}', \hat{\beta}')$ where $\mathcal{F}_{GANN}(\hat{\alpha}', \hat{\beta}')$ represents the reconfigured model, optimized to mitigate the impact of adversarial influence.

By leveraging defender-centric adversarial simulation, the GANN framework provides a rigorous and adaptable defense mechanism, empowering defenders to anticipate, detect, and mitigate adversarial threats effectively. This ensures the robustness and reliability of graph-based systems in dynamic, high-risk environments.

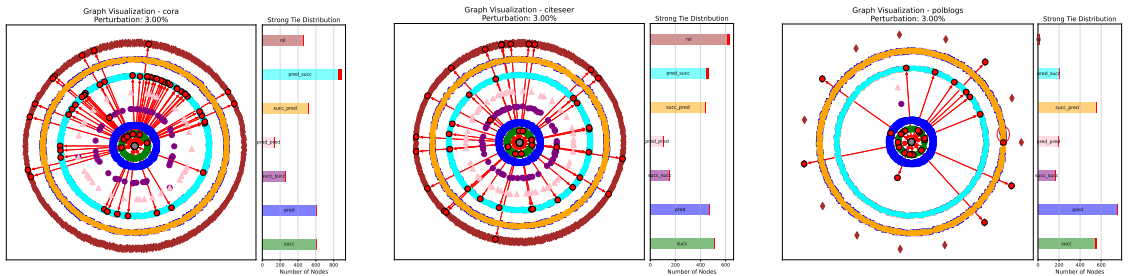


Figure 4.3: Graph visualization of strong ties in GANN under varying dataset.

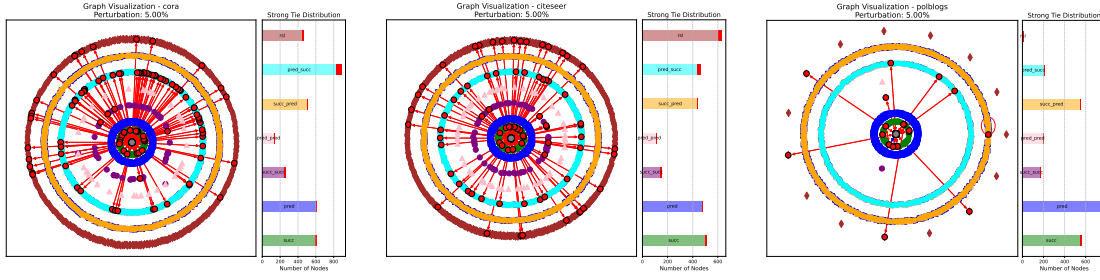


Figure 4.4: Strong tie visualization in GANN under poisoning attacks ($\alpha = 0.0$, $\beta = 1.0$) at 5% perturbation.

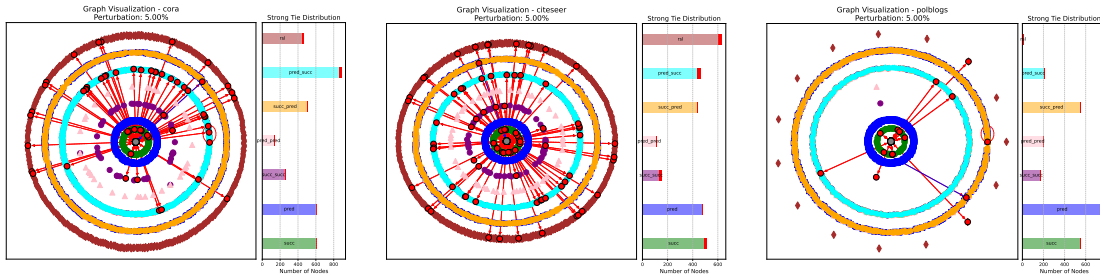


Figure 4.5: Strong tie visualization in GANN under poisoning attacks ($\alpha = 1.0$, $\beta = 0.0$) at 5% perturbation.

4.5 Tie-Sensitive Robustness Evaluation Under Adversarial Perturbation

This section investigates the impact of poisoning attacks across different graph regions by segmenting the network into neighborhood-based zones derived from the training set. The goal is to evaluate *zone-specific attacker preferences* within this training-driven partitioning.

4.5.1 Mathematical Formulation of Simulated Attacks

A poisoning attack strategically introduces malicious samples into the graph to degrade classification accuracy. The attack model is formulated as a bilevel optimization

problem:

$$\begin{aligned} \min_{D_P \subseteq \Phi} \mathbf{E} \left[l(f(x, w^*), y) \right] \\ \text{s.t. } w^* = \arg \min_{f(w) \in \mathcal{F}_{GAN}(\alpha, \beta)} L(D_C \cup D_P, w). \end{aligned} \quad (4.1)$$

where D_C represents the clean dataset containing nodes, edges, labels, and features, while D_P denotes the poisoning samples. The perturbation space is denoted by Φ , and $f(x, w)$ is the classifier parameterized by w .

The outer optimization seeks to minimize test accuracy, while the inner optimization refines model parameters to minimize empirical loss over the manipulated dataset $D_C \cup D_P$. This attack scenario can be interpreted as a *combinatorial search problem* over the perturbation space $(\frac{|\Phi|}{|D_P|})$.

4.5.2 Selection of Test Nodes Based on Tie-Sensitive Zones

To systematically analyze adversarial impact, nodes are categorized into zones based on their connectivity to training nodes $\mathcal{V}_{train} \subset D_C$. A test node $v_{test} \in \mathcal{V}_{test} \subset D$ is classified based on its relation to $v_{train} \in \mathcal{V}_{train}$ as follows:

$$v_{test} = \mathcal{N}_{selector} \left(\prod_{i=1}^l \mathcal{O}_i \right) (v_{train}),$$

where: $\mathcal{O}_i \in \{\mathcal{S}, \mathcal{P}\}$ denotes a directed operation, either the successor ($\mathcal{S}$) or predecessor ($\mathcal{P}$) operator. l represents the hop distance. $\prod_{i=1}^l \mathcal{O}_i$ signifies the sequential composition of l directional operators, maintaining the graph's structure. $\mathcal{N}_{selector}$ filters nodes based on specific neighborhood attributes. For $l = 1$, two primary zones emerge:

$$\mathcal{N}_1(v_{train}) = \{\mathcal{S}(v_{train}), \mathcal{P}(v_{train})\}.$$

Extending to $l = 2$, four additional zones appear:

$$\mathcal{N}_2(v_{train}) = \{\mathcal{S} \circ \mathcal{S}(v_{train}), \mathcal{S} \circ \mathcal{P}(v_{train}), \mathcal{P} \circ \mathcal{S}(v_{train}), \mathcal{P} \circ \mathcal{P}(v_{train})\}.$$

Any test node not falling into these defined regions is allocated to a residual zone:

$$Z_{\text{residual}} = \{v \in \mathcal{V}_{\text{test}} : v \notin \mathcal{N}_1(v_{\text{train}}) \cup \mathcal{N}_2(v_{\text{train}})\}.$$

Figures 4.3, 4.4, and 4.5 depict the zones distribution of strong ties within GANN under adversarial perturbations, offering insights into attack preference.

4.5.3 Fuzzy-Theoretic Evaluation of Trust Confidence Across Node Types

A fuzzy set framework is employed to quantify the trust confidence of each zone, measuring its resilience to adversarial perturbations. Formally, a fuzzy set is represented as $(\mathcal{N}, t)$, where: $\mathcal{N}$ comprises the designated test zones. $t : \mathcal{N} \rightarrow [0, 1]$ is the trust membership function, mapping each zone to a confidence level reflecting its susceptibility to perturbations. The support set of the fuzzy representation is given by:

$$\text{Supp}(\mathcal{N}, t) = \{\eta \in \mathcal{N} \mid t(\eta) > 0\}.$$

Algorithm 4 outlines the iterative computation and normalization of the trust membership function using the *Softmax* operator. It outputs a final decision score Q , which aggregates trust across all zones in a weighted manner. To understand how trust confidence reacts to small adversarial changes, we define its limiting response as:

$$t(A^\zeta, \eta) = \lim_{\|A'_\epsilon - A^\zeta\|_0 \rightarrow 0} t(A'_\epsilon, \eta),$$

where A^ζ is the observed graph and A'_ϵ is a minimally perturbed version of it. This definition captures the sensitivity of trust evaluation to even slight structural modifications, ensuring the framework remains robust under adversarial influence.

Figure 4.6 illustrates the average classification accuracy across six distinct zones, each characterized by specific neighborhood relationships:

$$\mathcal{N}(v_{train}) = \{\mathcal{S} \circ \mathcal{S}(v_{train}), \mathcal{S} \circ \mathcal{P}(v_{train}), \mathcal{P} \circ \mathcal{S}(v_{train}), \mathcal{P} \circ \mathcal{P}(v_{train}), \mathcal{S}(v_{train}), \mathcal{P}(v_{train})\}.$$

This assessment highlights the influence of various neighborhood structures on classification performance under adversarial conditions, providing insights into which regions of the graph are more vulnerable to poisoning attacks. Further refinement of this analysis is shown in Figure 4.7, where overlapping nodes from high-performing and low-performing zones are excluded. This step isolates the most susceptible areas, allowing for a more precise identification of zones requiring enhanced defensive mechanisms. Simulation results indicate that strong ties tend to cluster within specific zones, suggesting the presence of preferential adversarial behavior.

4.6 Formal Analysis of Trust Zone Confidence

In this section, we provide a theoretical analysis of provable trust zones, establishing a formal foundation for their role in ensuring graph robustness. Additionally, we introduce a provable trust-aware adaptive deviation framework tailored for poisoning graphs, designed to enhance resilience against adversarial perturbations.

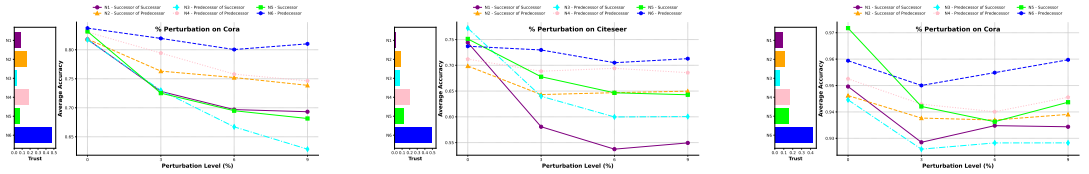


Figure 4.6: Node Labeling Accuracy Trends for Zones N1–N6 under Varying Perturbation Intensities

Algorithm 4: Setup Trust Zones and Query Decisions

Input: A^ζ : Observed adjacency matrix;

A' : Poisoned adjacency matrix;

$\mathcal{F}_{GANN}(\alpha, \beta)$: GANN classifier ensemble;

Φ : Perturbation space; Test zones $\mathcal{N}$, Zone $\eta \in \mathcal{N}$;

Temperature parameter τ ; $\mathbf{X}(\eta)$: Feature vector associated with η .

Output: Trust confidence $t(\eta)$ for each $\eta \in \mathcal{N}$;

Decision Q based on weighted combination of trust confidence.

Initialize: Set $t(\eta) \leftarrow 0$ for all $\eta \in \mathcal{N}$;

Simulation: Run iterative propagation on graphs under adversarial conditions ;

for $f \in \mathcal{F}_{GANN}(\alpha, \beta)$ **do**

for $A' \in \Phi, \eta \in \mathcal{N}$ **do**

 Compute perturbation degree:

$\Delta f(\eta) \leftarrow f(A^\zeta, \eta) - f(A', \eta)$;

 Update trust membership function:

$t(\eta) \leftarrow \frac{f(A^\zeta, \eta)}{\Delta f(\eta) + 1}$;

Normalize trust scores using Softmax with temperature τ :

$t(\eta) \leftarrow \frac{\exp(t(\eta)/\tau)}{\sum_{\eta' \in \mathcal{N}} \exp(t(\eta')/\tau)} \quad \forall \eta \in \mathcal{N}$;

Compute decision Q for query node v_q :

$Q = \sum_{\eta \in \mathcal{N}(v_q)} t(\eta) \cdot \mathbf{X}(\eta)$;

return Trust confidence $t(\eta)$ for all $\eta \in \mathcal{N}$, Decision Q .

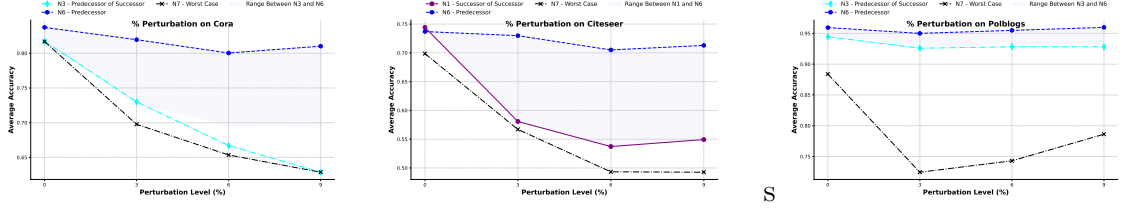


Figure 4.7: Zone-Based Accuracy Evaluation: N7 Extracted as Worst-Case from N1–N6 by Removing Stable Overlapping Nodes

4.6.1 Granular Analysis of Trust Zones

The test zones can be represented as subsets of k edges selected from the edge set E , denoted as $\binom{|E|}{k}$, where E consists of all edges in A^ζ . Each test subspace is formed by sampling k edges uniformly at random without replacement from $|E|$. When $\binom{|E|}{k}$ is relatively small (e.g., in the order of hundreds), an exhaustive enumeration of all possible test zones is computationally feasible. In this setting, the proportion of test zones that contain at least one poisoned edge is given by $1 - \frac{\binom{|E|-\Delta}{k}}{\binom{|E|}{k}}$. In the worst-case scenario, the trust confidence shifts from t_a to t_b , where these Δ adversarially injected edges reduce the highest trust confidence t_a by a factor of $1 - \frac{\binom{|E|-\Delta}{k}}{\binom{|E|}{k}}$ while simultaneously increasing the second highest trust confidence t_b by the same amount. Formally, this yields the following trust confidence transformations:

$$t'_a = t_a - \left(1 - \frac{\binom{|E|-\Delta}{k}}{\binom{|E|}{k}}\right), \quad t'_b = t_b + \left(1 - \frac{\binom{|E|-\Delta}{k}}{\binom{|E|}{k}}\right). \quad (4.2)$$

The trust confidence within small test zones satisfies the inequality:

$$t'_a > t'_b \iff t_a - t_b > 2 - 2 \frac{\binom{|E|-\Delta}{k}}{\binom{|E|}{k}}. \quad (4.3)$$

For cases where $\binom{|E|}{k}$ is large, the graph can be partitioned into N subgraphs, $G_1, G_2, \dots, G_N$, ensuring that the number of edge combinations within each subgraph remains computationally manageable. Given this partitioning, we derive a lower bound $\underline{t}_a$ for t_a while estimating an upper bound $\bar{t}_b = 1 - \underline{t}_a \geq 1 - t_a \geq t_b$ for t_b . By substituting

$\underline{t}_a$ and $\bar{t}_b$ into Inequality (4), we obtain the refined trust confidence inequality:

$$\underline{t}_a - \bar{t}_b > 2 - 2 \frac{\binom{|E|-\Delta}{k}}{\binom{|E|}{k}}. \quad (5)$$

4.6.2 Theoretical Guarantees of Trust Zone Confidence

In the Graph Adaptive Neural Network (GANN) framework, the classifier set is denoted as $\mathcal{F}_{GANN}(\alpha, \beta)$, where each instance f represents an adaptive ensemble global model tailored for trust zone evaluation. Given a test zone η , trust confidence levels are assessed, with t_a and t_b representing the highest and second-highest trust confidence scores, respectively. The bounds $\underline{t}_a$ and $\bar{t}_b$ provide a lower estimate for t_a and an upper estimate for t_b . Similarly to the provable security in [104], the following theorem establishes the conditions for the provable trust zones.

Theorem 4. *Given any classifier $f \in \mathcal{F}$ within the GANN model and a test zone η , the trust confidence values satisfy the following condition:*

$$\max_{i \neq a} t_i = t_b \leq \bar{t}_b \leq \underline{t}_a \leq t_a. \quad (4.4)$$

A zone a is considered provably trustworthy when the maximum perturbation budget Δ does not exceed a predefined threshold, such that:

$$t(A', \eta) = t(A^\zeta, \eta) = t_a, \quad \forall A', \quad \|A^\zeta - A'\|_0 \leq \Delta. \quad (4.5)$$

To demonstrate the tightness of the derived trust level, we analyze the edge structure of the graph. Let E and E' represent the sets of edges in the observed graph A^ζ and the perturbed graph A' , respectively. The subspace of k -edge samples is represented as $\binom{|E|}{k}$, with subsets sampled uniformly without replacement. The space of all possible k -edge subsets from E and E' is denoted as $O_E = \{S(E, k)\}$ and $O_{E'} = \{S(E', k)\}$, respectively. The intersection space $O_o = O_E \cap O_{E'}$ contains all possible k -edge subsets that remain unchanged between E and E' . Figure 4.8 illustrates these spaces.

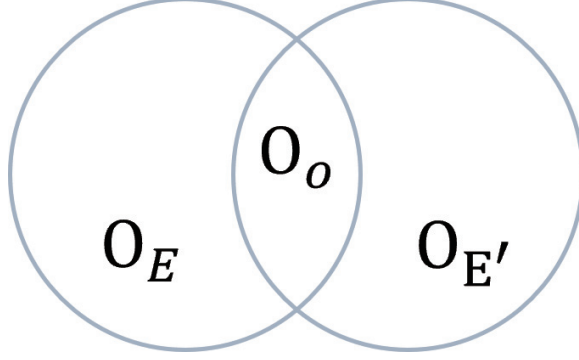


Figure 4.8: Construction of Subspaces O_E , $O_{E'}$, and O_o .

Proof. For any random variable $\mathbf{X}$ denoting a sampled subset $S(E, k)$, the probability distributions are defined as:

$$\Pr(\mathbf{X} = \eta) = \begin{cases} \frac{1}{\binom{|E|}{k}}, & \text{if } \eta \in O_E \\ 0, & \text{otherwise.} \end{cases} \quad (4.6)$$

$$\Pr(\mathbf{Y} = \eta) = \begin{cases} \frac{1}{\binom{|E'|}{k}}, & \text{if } \eta \in O_{E'} \\ 0, & \text{otherwise.} \end{cases} \quad (4.7)$$

The trust score for each test zone is computed based on the probability distribution of sampled edges. The probability of assigning trust confidence t_a is given by:

$$t_a = \Pr(t(\mathbf{X}, \eta) = a) \quad (4.8)$$

$$= \Pr(t(\mathbf{X}, \eta) = a \mid \mathbf{X} \in O_o) \cdot \Pr(\mathbf{X} \in O_o) \quad (4.9)$$

$$+ \Pr(t(\mathbf{X}, \eta) = a \mid \mathbf{X} \in (O_E - O_o)) \cdot \Pr(\mathbf{X} \in (O_E - O_o)). \quad (4.10)$$

A similar formulation holds for the perturbed graph A' :

$$t'_a = \Pr(t(\mathbf{Y}, \eta) = a) \quad (4.11)$$

$$= \Pr(t(\mathbf{Y}, \eta) = a \mid \mathbf{Y} \in O_o) \cdot \Pr(\mathbf{Y} \in O_o) \quad (4.12)$$

$$+ \Pr(t(\mathbf{Y}, \eta) = a \mid \mathbf{Y} \in (O_{E'} - O_o)) \cdot \Pr(\mathbf{Y} \in (O_{E'} - O_o)). \quad (4.13)$$

Chapter 4. Adversarial Robustness in Node Labeling via Strong-Tie Adaptive Defense

Given that the probability of unchanged edge samples remains the same in both O_E and $O_{E'}$, we derive:

$$\Pr(\mathbf{X} \in O_o) = \Pr(\mathbf{Y} \in O_o) = \frac{\binom{|E|-\Delta}{k}}{\binom{|E|}{k}}. \quad (4.14)$$

By subtracting the perturbed and unperturbed trust probabilities, we derive:

$$t'_a - t_a \geq t_a - t_i - 2 \left[1 - \frac{\binom{|E|-\Delta}{k}}{\binom{|E|}{k}} \right], \quad (4.15)$$

which ensures that trust confidence remains valid under the specified perturbation budget. ■

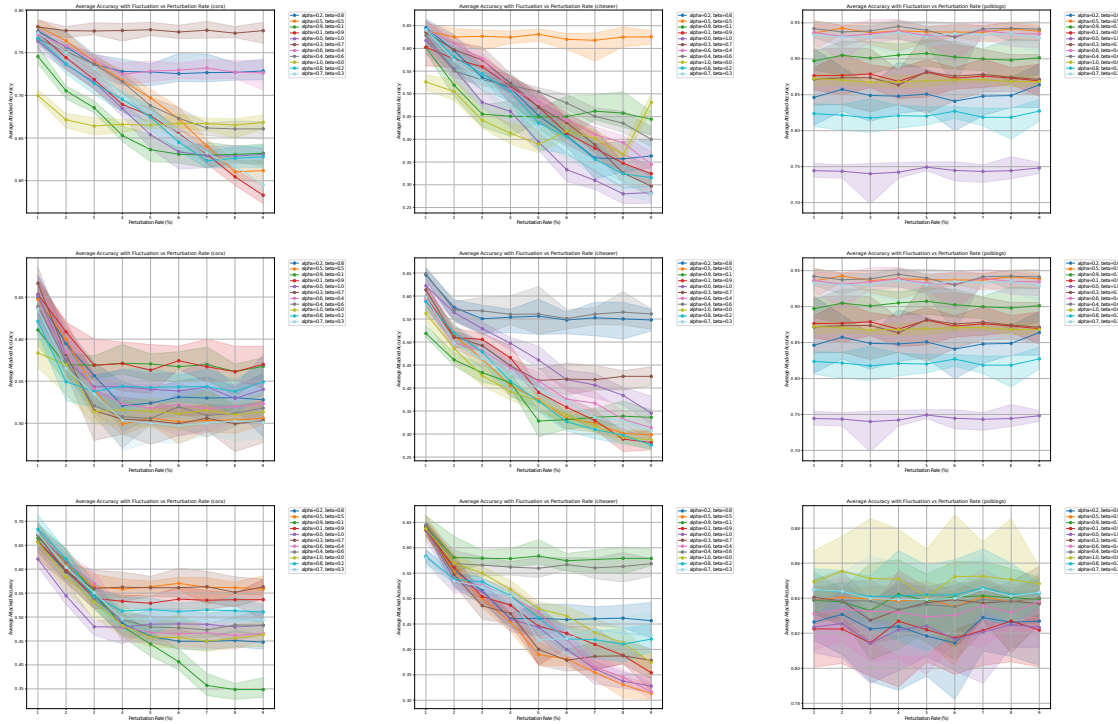


Figure 4.9: Accuracy variation with perturbation rates for GANN, Toy-1, and Toy-2.

4.7. Structure-Guided Adaptive Defense for Trust-Aware Poisoning Mitigation

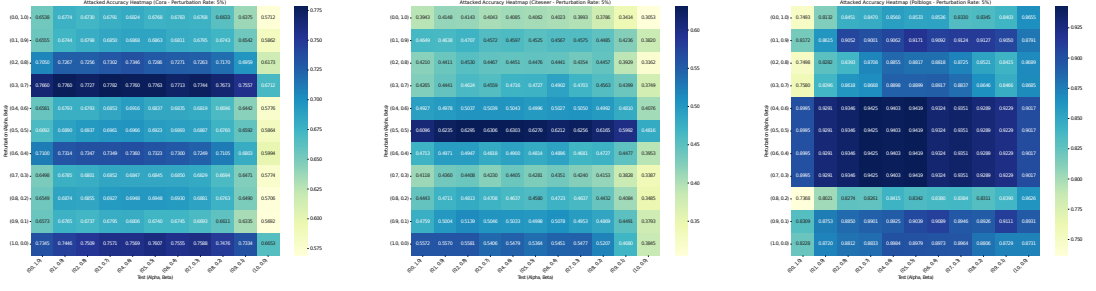


Figure 4.10: Heatmap of attacked accuracy across different (α, β) configurations at 5% perturbation.

4.7 Structure-Guided Adaptive Defense for Trust-Aware Poisoning Mitigation

Expanding upon the trust zones identified via the fuzzy set framework in GANN, we proposed an adaptive mechanism for mitigating poisoning attacks. The core idea is to strategically modulate propagation between trust and risk zones to minimize adversarial influence. Ideally, message passing remains restricted within trust zones to reinforce robustness; however, due to inherent structural overlaps with risk zones, achieving complete isolation is often impractical. Therefore, we introduce a trust-aware adaptive propagation method that adapts propagation routes based on trust and risk evaluations, effectively reducing or counteracting poisoning effects.

The variation in trust confidence across zones reflects the underlying preferences of the attacker. Zones with consistently high trust confidence are less likely to be targeted, whereas those with fluctuating or lower confidence are more susceptible to adversarial influence. From a system optimization perspective, the attacker’s strategy tends to converge toward structurally weaker zones, making their preferences relatively stable over time. In non-i.i.d. graphs where trust and risk zones may overlap, understanding attacker preference becomes essential. To quantify how adversaries exploit different structural contexts, we define χ as a preference-aware parameter shaped by the local

neighborhood structure. By aggregating these sensitivity scores, χ captures structural dependencies that influence the adversary’s optimal strategy. This provides a principled way to analyze which substructures are more likely to be targeted, enabling more informed and adaptive defenses.

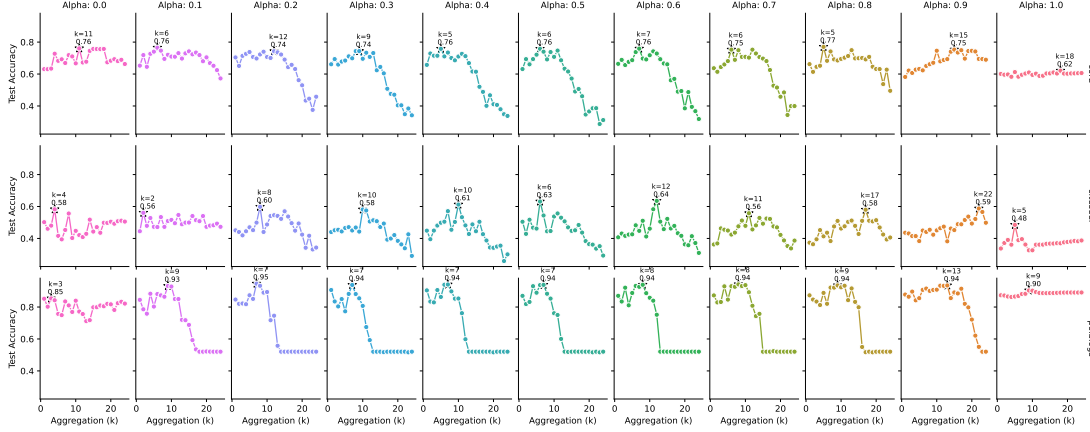


Figure 4.11: Analysis of aggregation depth (k) impact on poisoned graphs.

4.7.1 Theoretical Deviation Analysis under Adaptive Trust Propagation

To assess the impact of adversarial perturbations, we introduce a formal deviation measure between clean and poisoned samples. This measure facilitates understanding how different neighborhood structures contribute to adversarial susceptibility, aiding in the development of propagation strategies that mitigate attack influence.

Theorem 5. Suppose a set of clean samples D_C of size n_c follows a non-i.i.d. distribution $\mathcal{N}(\mu, \Sigma)$. A set of poisoning samples D_P of size $n_p = \epsilon \cdot n_c$ is introduced, where the distance between the centers is given by $\|\mu_p - \mu\|_2 = d = \chi \cdot \sigma$. Here, χ represents an attacker preference-aware parameter. The deviation between clean and

poisoned samples can be expressed as:

$$\mathbf{E}_{i \in D_P}[q^2(x_i)] - \mathbf{E}_{i \in D_C}[q^2(x_i)] \geq \left(\frac{1-\epsilon}{1+\epsilon} \cdot \chi^2 - \left(\frac{1}{\epsilon} + 1 \right) \right) \sigma^2. \quad (4.16)$$

This theorem quantifies the deviation between poisoned samples (size n_p) and clean samples (size n_c). χ is defined as a *preference-aware parameter* that measures the alignment between the attacker’s structural targeting and the system’s adaptive propagation. We decompose it as:

$$\chi^2 = \chi_p \cdot \chi_a \cdot \cos \theta,$$

where χ_p is the strength of the attacker’s preference, χ_a is the strength of the system’s propagation response, and θ is the angle between the two in structural space. When $\theta = 90^\circ$, the two vectors are orthogonal and $\chi^2 = 0$, meaning the attack has no effect—indicating ideal robustness through preference suppression. We consider two regimes: 1. Positive Deviation: If $\chi^2 > \frac{(1+\epsilon)^2}{\epsilon(1-\epsilon)}$, poisoned samples deviate significantly from clean ones, revealing strong attacker alignment and insufficient propagation correction. 2. Negative Deviation: If χ^2 is small, poisoned samples resemble clean samples, indicating that adaptive propagation has successfully weakened attacker preference. Ideally, maintaining $\chi^2 \leq \frac{(1+\epsilon)^2}{\epsilon(1-\epsilon)}$ ensures robustness. Assuming clean samples D_C are drawn from $\mathcal{N}(\mu, \Sigma)$, and following methodologies in [138, 139], we introduce this deviation score as an extension of previously established *poisoning scores*.

Method	Cora (Test Accuracy %)		
	25%	50%	75%
Robust-GCN	56.43 $\pm$ 0.64	36.68 $\pm$ 1.19	25.79 $\pm$ 3.17
GCN-Jaccard	65.72 $\pm$ 1.04	46.29 $\pm$ 2.26	32.55 $\pm$ 1.34
GCN-SVD	58.28 $\pm$ 0.99	36.54 $\pm$ 1.78	25.13 $\pm$ 3.22
Pro-GNN	72.55 $\pm$ 1.66	44.44 $\pm$ 2.81	36.17 $\pm$ 0.74
GNNGuard	54.46 $\pm$ 1.44	34.45 $\pm$ 2.47	25.73 $\pm$ 3.35
Elastic GNN	57.51 $\pm$ 1.67	43.83 $\pm$ 2.01	29.49 $\pm$ 1.69
STABLE	76.58 $\pm$ 0.87	71.07 $\pm$ 0.71	63.14 $\pm$ 1.19
EvenNet	76.54 $\pm$ 0.37	71.40 $\pm$ 0.91	67.09 $\pm$ 0.58
GCN-GARNET	74.57 $\pm$ 0.69	70.61 $\pm$ 0.96	68.07 $\pm$ 0.79
HANG-quad	67.45 $\pm$ 0.95	65.63 $\pm$ 0.60	63.07 $\pm$ 1.14
GADC (IV)	76.75 $\pm$ 1.46	70.24 $\pm$ 0.70	66.19 $\pm$ 0.40
GANN	77.88 $\pm$ 0.46	78.42 $\pm$ 0.44	78.41 $\pm$ 0.49

Table 4.1: Test accuracy (%) under non-adaptive adversarial attack conditions on Cora.

Proof. Let D_C , D_P , and $D_C \cup D_P$ denote the clean, poisoned, and combined samples, respectively, with centers μ_c , μ_p , and $\mu_{c \cup p}$. Given $n_p = \epsilon \cdot n_c$, the center displacement can be written as:

$$d = \|\mu_c - \mu_p\| = \left(1 + \frac{1}{\epsilon}\right) \|\mu_c - \mu_{c \cup p}\| = (1 + \epsilon) \|\mu_p - \mu_{c \cup p}\|. \quad (4.17)$$

Using the top-right singular vector of the centered data matrix of $D_C \cup D_P$, the *deviation score* for the combined set with any unit vector $\mathcal{V}$ is:

$$\mathbf{E}_{i \in D_C \cup D_P}[(\mathcal{V} \cdot (x - \mu_{C \cup P}))^2] = \mathbf{E}_{i \in D_C \cup D_P}[(\mathcal{V} \cdot (x - \mu_p) + \mathcal{V} \cdot (\mu_p - \mu_{C \cup P}))^2]. \quad (4.18)$$

Expanding and simplifying:

$$= \mathbf{E}_{i \in D_C \cup D_P}[(\mathcal{V} \cdot (x - \mu_p))^2] - (\mathcal{V} \cdot (\mu_p - \mu_{C \cup P}))^2. \quad (4.19)$$

Using the unit vector $\mathcal{V}' = \frac{\mu_p - \mu_{C \cup P}}{\|\mu_p - \mu_{C \cup P}\|}$:

$$\mathbf{E}_{i \in D_C \cup D_P}[(\mathcal{V}' \cdot (x - \mu_{C \cup P}))^2] \geq \mathbf{E}_{i \in D_C \cup D_P}[(\mathcal{V}' \cdot (x - \mu_p))^2] - \|\mu_p - \mu_{C \cup P}\|^2. \quad (4.20)$$

For the deviation score:

$$\mathbf{E}_{i \in D_C \cup D_P}[(\mathcal{V}' \cdot (x - \mu_p))^2] \geq \frac{1}{1 + \epsilon} \mathbf{E}_{i \in D_C}[(\mathcal{V}' \cdot (x - \mu_c))^2] = \frac{1}{1 + \epsilon} \|\mu_c - \mu_p\|^2. \quad (4.21)$$

Thus, the average *deviation score* for $D_C \cup D_P$:

$$\mathbf{E}_{i \in D_C \cup D_P}[(\mathcal{V}' \cdot (x - \mu_{C \cup P}))^2] \geq \frac{1}{(1 + \epsilon)^2} d^2. \quad (4.22)$$

The deviation score for clean samples:

$$\mathbf{E}_{i \in D_C}[(\mathcal{V}' \cdot (x - \mu_p))^2] \leq \sigma^2 + \left(\frac{\epsilon}{1 + \epsilon} d \right)^2, \quad (4.23)$$

and for poisoned samples:

$$\mathbf{E}_{i \in D_P}[(\mathcal{V}' \cdot (x - \mu_p))^2] \geq \frac{1}{(1 + \epsilon)^2} d^2 - \frac{1}{\epsilon} \sigma^2. \quad (4.24)$$

Thus, the *deviation score* between poisoned and clean samples is:

$$\mathbf{E}_{i \in D_P} - \mathbf{E}_{i \in D_C} \geq \left(\frac{1 - \epsilon}{1 + \epsilon} \chi^2 - \left(\frac{1}{\epsilon} + 1 \right) \right) \sigma^2. \quad (4.25)$$

This completes the proof for Theorem 5, establishing a formal measure of deviation between clean and poisoned distributions based on neighborhood sensitivity. ■

4.7.2 Adaptive Mitigation via Trust Zone–Attack Preference Alignment

Theorem 5 establishes fundamental conditions for implementing an effective trust-aware adaptive mitigation strategy. By leveraging neighbor types to assign dynamic trustworthiness scores, the adaptive propagation mechanism facilitates the identification of regions within the graph that exhibit a high concentration of adversarially introduced perturbations. This approach ensures that trust zones maintain a high degree of resilience against adversarial influences while preserving the structural integrity of the graph.

When negative deviation is observed, adaptive propagation serves as a countermeasure to mitigate the adverse effects of poisoning attacks. This ensures that the propagation process does not amplify adversarial disruptions but instead neutralizes them, allowing for a more stable and reliable representation of the node labeling. By adaptive propagation based on trust zone distributions, the framework effectively minimizes the impact of perturbations while maintaining consistency in graph-based learning models.

Furthermore, the adaptive propagation process continuously refines itself based on observed deviations, allowing for an evolving defense mechanism that enhances robustness over time. This ensures that trust zones remain stable and less susceptible to adversarial manipulation, thereby strengthening the overall security of the graph model. The application of this trust-aware adaptive mitigation framework provides a structured and theoretically grounded approach to preserving the fidelity of graph-based learning systems under adversarial conditions.

Method	Citeseer (Test Accuracy %)		
	25%	50%	75%
GCN	56.21 $\pm$ 1.26	42.41 $\pm$ 2.83	32.00 $\pm$ 3.08
GAT	59.80 $\pm$ 1.08	45.78 $\pm$ 2.06	36.37 $\pm$ 2.68
APPNP	62.65 $\pm$ 1.01	55.50 $\pm$ 1.75	44.41 $\pm$ 1.51
S ² GC	58.29 $\pm$ 0.00	41.23 $\pm$ 0.00	29.72 $\pm$ 0.22
NAGphormer	65.28 $\pm$ 0.72	54.99 $\pm$ 1.47	44.00 $\pm$ 2.56
Robust-GCN	56.62 $\pm$ 0.65	41.20 $\pm$ 0.85	29.81 $\pm$ 1.06
GCN-Jaccard	60.30 $\pm$ 1.10	47.72 $\pm$ 1.65	38.68 $\pm$ 2.66
GCN-SVD	66.97 $\pm$ 0.83	59.30 $\pm$ 0.89	40.87 $\pm$ 1.50
Pro-GNN	66.88 $\pm$ 1.42	50.78 $\pm$ 2.06	33.74 $\pm$ 1.81
GNNGuard	55.91 $\pm$ 1.71	41.19 $\pm$ 1.84	31.78 $\pm$ 1.63
Elastic GNN	63.99 $\pm$ 1.02	52.99 $\pm$ 1.11	43.56 $\pm$ 1.75
GANN	68.49 $\pm$ 0.56	67.63 $\pm$ 0.32	66.01 $\pm$ 0.58

Table 4.2: Test accuracy (%) under non-adaptive adversarial attack conditions on Citeseer.

Method	Pubmed (Test Accuracy %)		
	25%	50%	75%
GCN	43.16 $\pm$ 4.83	35.91 $\pm$ 3.91	35.33 $\pm$ 4.31
GAT	31.19 $\pm$ 2.95	29.33 $\pm$ 2.46	26.71 $\pm$ 1.95
APPNP	38.78 $\pm$ 3.27	39.96 $\pm$ 0.00	39.96 $\pm$ 0.00
S ² GC	32.09 $\pm$ 2.14	39.96 $\pm$ 0.02	39.96 $\pm$ 0.00
NAGphormer	71.43 $\pm$ 1.71	47.60 $\pm$ 3.33	34.71 $\pm$ 4.32
Robust-GCN	32.85 $\pm$ 1.52	35.19 $\pm$ 2.10	35.03 $\pm$ 2.31
GCN-Jaccard	43.78 $\pm$ 4.74	35.89 $\pm$ 3.92	35.31 $\pm$ 4.31
GCN-SVD	78.73 $\pm$ 0.41	55.00 $\pm$ 1.03	36.28 $\pm$ 2.47
GNNGuard	44.51 $\pm$ 2.88	37.56 $\pm$ 4.28	37.45 $\pm$ 4.92
Elastic GNN	54.56 $\pm$ 0.52	39.94 $\pm$ 0.00	39.94 $\pm$ 0.00
STABLE	35.73 $\pm$ 4.37	37.66 $\pm$ 5.81	37.05 $\pm$ 6.42
GANN	82.67 $\pm$ 0.22	82.35 $\pm$ 0.20	81.90 $\pm$ 0.17

Table 4.3: Test accuracy (%) under non-adaptive adversarial attack conditions on Pubmed.

In adaptive propagation, partitioned zones represent regions of the graph with varying degrees of labeling reliability, classified according to the relationship with the training set. Through trust-weighted path, GANN prioritizes secure propagation paths while attenuating the influence of risky connections. To formalize this propagation mech-

anism, we define the trust-weighted path operator $\mathcal{P}_k(\alpha, \beta, t(\eta))$, which adaptively modulates propagation based on zone-specific trust confidence scores. This operator consists of a polynomial expansion of adjacency transformations, where each term is weighted by the trustworthiness of its corresponding zone:

$$\mathcal{P}_k(\alpha, \beta, t(\eta)) = \sum_{j=0}^k \sum_{\pi \in \mathcal{S}(j, k-j)} \alpha^j \beta^{k-j} \left(\prod_{i \in \pi} \mathcal{M}_i \right) \cdot t(\eta), \quad (4.26)$$

Table 4.4: Dataset characteristics, including statistics for Citeseer, Cora, Polblogs, and PubMed.

Dataset	Citeseer	Cora	Polblogs	Pubmed
Classes	6	7	2	3
Features	3,703	1,433	1,490	500
Nodes	2,110	2,708	1,222	19,717
Edges	3,668	5,429	16,714	44,324

where α and β are parameters that regulate within each traversal path. By modulating α and β , the operator dynamically adjusts directional preference during propagation across the graph. $\mathcal{S}(j, k-j)$ represents the set of all unique arrangements of j forward steps (denoted as $\hat{A}$) and $k-j$ backward steps (denoted as $\hat{A}^T$), ensuring that the non-commutative properties of $\hat{A}$ and $\hat{A}^T$ are maintained.

The term $\prod_{i \in \pi} \mathcal{M}_i$ denotes the sequence of adjacency transformations corresponding to each unique propagation path in $\mathcal{S}(j, k-j)$, where $\mathcal{M}_i = \hat{A}$ for forward propagation and $\mathcal{M}_i = \hat{A}^T$ for reverse propagation. Each term in $\mathcal{P}_k$ represents a k -step walk across the graph, with $\hat{A}$ guiding forward movements and $\hat{A}^T$ encoding reverse movements, thereby preserving the structural constraints inherent in directed propagation paths. Incorporating the trust-weighted path operator into the adversarial objective,

we define a robust optimization problem:

$$\begin{aligned} \min_{D_P \subseteq \Phi} \mathbf{E} \left[l(f(\mathcal{P}_k(\alpha, \beta, t(\eta)) \cdot X_{GANN}, w), y) \right] \\ \text{s.t. } w^* = \arg \min_w L(D_C \cup D_P, w, \mathcal{P}_k(\alpha, \beta, t(\eta))). \end{aligned}$$

In this setting, the trust-weighted path operator $\mathcal{P}_k(\alpha, \beta, t(\eta))$ dynamically modulates the propagation process by amplifying information flow in high-trust zones while attenuating it in regions deemed susceptible to adversarial influence. The resulting formulation, $\mathcal{P}_k(\alpha, \beta, t(\eta)) \cdot X_{GANN}$, represents the classifier within GANN, structured to leverage adaptive trust-aware propagation mechanisms. This approach ensures an optimal balance between attacker preference and adaptive propagation, effectively mitigating adversarial perturbations while preserving the integrity of learned representations.

Table 4.5: Test accuracy (%) under poisoning attacks, comparing GANN against Dir-GNN and Magnet under adaptive adversarial settings.

Dataset	Cora	Citeseer	Polblogs
Dir-GNN	71.45 $\pm$ 0.59	52.25 $\pm$ 1.74	88.01 $\pm$ 1.92
Magnet	76.39 $\pm$ 0.67	58.23 $\pm$ 2.67	89.92 $\pm$ 2.65
GANN	77.43 $\pm$ 0.65	63.10 $\pm$ 0.96	93.75 $\pm$ 0.67

4.8 Experimental Results

Baseline Methods: We evaluate our approach against a range of existing models under both *non-adaptive* and *adaptive* attack scenarios. For non-adaptive attacks, we consider two standard GNN architectures: GCN [140] and GAT [141]; two graph diffusion convolutional models: APPNP [142] and S²GC [143]; and a graph transformer model, NAGphormer [144]. Additionally, we benchmark against

several adversarial defense strategies, including Robust-GCN[76], GCN-Jaccard[71], GCN-SVD[145], Pro-GNN[146], GNNGuard[147], Elastic GNN[148], STABLE[149], EvenNet[150], GCN-GARNET[151], HANG-quad[152], and GADC(IV)[153]. For *adaptive* attacks, we employ Dir-GNN [154] and Magnet [155], as detailed in Table 4.5.

Experimental Setup: As presented in Table 4.1,4.2,4.3, we generate adversarial graph structures using DeepRobust, applying perturbations at levels of 0.25, 0.5, and 0.75 on the Cora, Citeseer, and Pubmed datasets under *non-adaptive attack* settings. We maintain dataset splits consistent with prior work. Each experiment is repeated 10 times, and we report the mean test accuracy alongside its standard deviation. In GANN, a two-layer MLP with 16 hidden units follows the feature aggregation stage. The adaptive propagation parameters are tuned using the validation dataset. For *adaptive attacks* targeting directed neural networks, we allocate a minimal but effective perturbation budget to assess model robustness.

Computational Complexity: While GANN incorporates an iterative aggregation mechanism, we mitigate excessive computational overhead by decoupling aggregation from training. Unlike APPNP, which recomputes aggregation at each training step, GANN performs aggregation only once, significantly reducing redundant computations. Running 100 epochs of GANN on the Pubmed dataset requires approximately 59.72 seconds, demonstrating the computational efficiency of our approach.

For empirical evaluation, we use real-world citation networks, including Citeseer, Cora, Pubmed, and Polblogs. Dataset statistics are presented in Table 4.4. Figures 4.9 and 4.10 illustrate the impact of varying alpha-beta configurations on classification accuracy and robustness, displaying fluctuations across perturbation rates and a heatmap of model performance under adversarial settings. Figure 4.11 analyzes the effect of aggregation depth k on poisoned graphs, identifying the optimal k for maximizing accuracy under attack conditions.

4.9 Summary

This study presents the Graph Adaptive Neural Network (GANN), a robust framework for defending graph-based systems against structural poisoning attacks. Starting from graph partitioning, we identify attacker preference by analyzing how adversarial perturbations disproportionately target specific zones. We then define *strong ties* as the structural connections between sensitive zones and the training set—key pathways that adversaries exploit to influence label propagation. GANN adaptively regulates these strong ties to balance two opposing forces: attacker preference and propagation fidelity. By tuning propagation parameters in GANN, the framework neutralizes targeted perturbations and maintains stable label prediction. This dynamic adjustment mechanism makes poisoning samples ineffective, as their influence is absorbed or redirected before affecting the training distribution. Through preference-aware propagation control and zone-level structural awareness, GANN significantly enhances model resilience. Empirical results confirm that GANN outperforms existing defenses by improving label stability under attack, validating the effectiveness of attacker preference suppression as a new direction for graph-based robustness.

Chapter 5

Leveraging Strong-Tie Structures for LLM-Driven Annotation and GNN Label Stability

5.1 Motivation

Graph-based models, particularly Graph Neural Networks (GNNs), heavily rely on label-rich supervision, yet acquiring high-quality labeled data remains a fundamental challenge due to the cost and complexity of manual annotation. Existing structure-based annotation methods, such as unsupervised propagation or clustering, often suffer from poor stability and limited generalization, resulting in biased or unreliable node representations.

Despite their success in zero-shot and few-shot learning, LLMs face fundamental limitations when applied to graph annotation tasks, primarily due to the gap between textual reasoning and structural information. Key challenges include: 1. Structural Sensitivity: LLMs primarily operate on text-based embeddings, lacking an innate ability to capture graph topologies. 2. Semantic Drift: Annotation performance may deteriorate due to contextual inconsistencies when aggregating information across

multiple hops. 3. Scalability: Naive prompt engineering for each node does not scale well due to computational overhead, necessitating an efficient graph-based selection strategy. To tackle these challenges, we develop a homophily-aware LLM annotation framework that systematically integrates structural and textual cues, enabling scalable and robust graph annotation.

While LLM-based annotation has demonstrated its potential, the integration of structural information into this process remains underexplored. Contrary to the prevailing notion that structural context contributes only marginally to annotation quality, we propose a novel approach that harnesses crowdsourced subgraph patterns. In this work, we introduce *Crowdsourced Homophily Ties-Based Graph Annotation via Large Language Models (CSA-LLM)*, a framework that challenges conventional annotation methodologies by incorporating structural context in prompt design. CSA-LLM integrates both individual node information and neighborhood relationships - capturing direct and indirect connections - to refine the reliability of the annotation. By modeling homophily relationships as strong ties—where nodes with similar labels tend to form tightly connected subgraphs [42]—CSA-LLM improves the ability of large language models to generate annotations that are semantically accurate and structurally coherent.

This approach is systematically evaluated on the *Cora* and *Citeseer* datasets [156], which serve as well-established benchmarks in graph-based learning. These datasets combine textual attributes with structural graph properties, making them ideal testbeds for assessing the performance of CSA-LLM. Experimental results demonstrate that CSA-LLM significantly improves annotation accuracy by effectively incorporating structural dependencies into the annotation process.

Beyond improving annotation quality, we introduce a *novel filtering mechanism* [1, 157, 158] that prioritizes structurally significant nodes based on graph metrics such as PageRank, degree centrality, and embedding density. By focusing annotation efforts on these pivotal nodes, CSA-LLM optimizes both efficiency and accuracy. This demonstrates that LLMs can not only generate annotations at scale but also adap-

tively refine annotations based on structural importance, making label-free graph annotation more robust and scalable.

The remainder of this chapter is structured as follows:

- **Section II** introduces key preliminaries on GNNs and text-attributed graphs (TAGs).
- **Section III** details the CSA-LLM methodology, highlighting the integration of homophily ties into LLM-based annotation and the implementation of our novel filtering mechanism.
- **Section IV** provides a theoretical analysis of the role of homophily ties in graph learning.
- **Section V** presents the experimental setup, evaluation metrics, and results.
- **Section VI** concludes the paper by summarizing our contributions and discussing potential directions for future research.

5.2 Preliminaries

In this section, we formally introduce the core concepts and notation that form the foundation of our study, which focuses on directed Text-Attributed Graphs (TAGs). A TAG is defined as $G_T = (V, A, T, X)$, where $V = \{v_1, v_2, \dots, v_n\}$ denotes the set of n nodes in the graph, with each node v_i representing a distinct entity. Each node v_i is associated with a textual attribute t_i , and the collection of all node attributes forms the set $T = \{t_1, t_2, \dots, t_n\}$, which encapsulates semantic information essential for understanding the contextual role of nodes within the graph. These textual attributes are transformed into numerical representations $X = \{x_1, x_2, \dots, x_n\}$ using SentenceBERT [159], a transformer-based model designed to generate high-quality sentence embeddings tailored for natural language processing (NLP) applications.

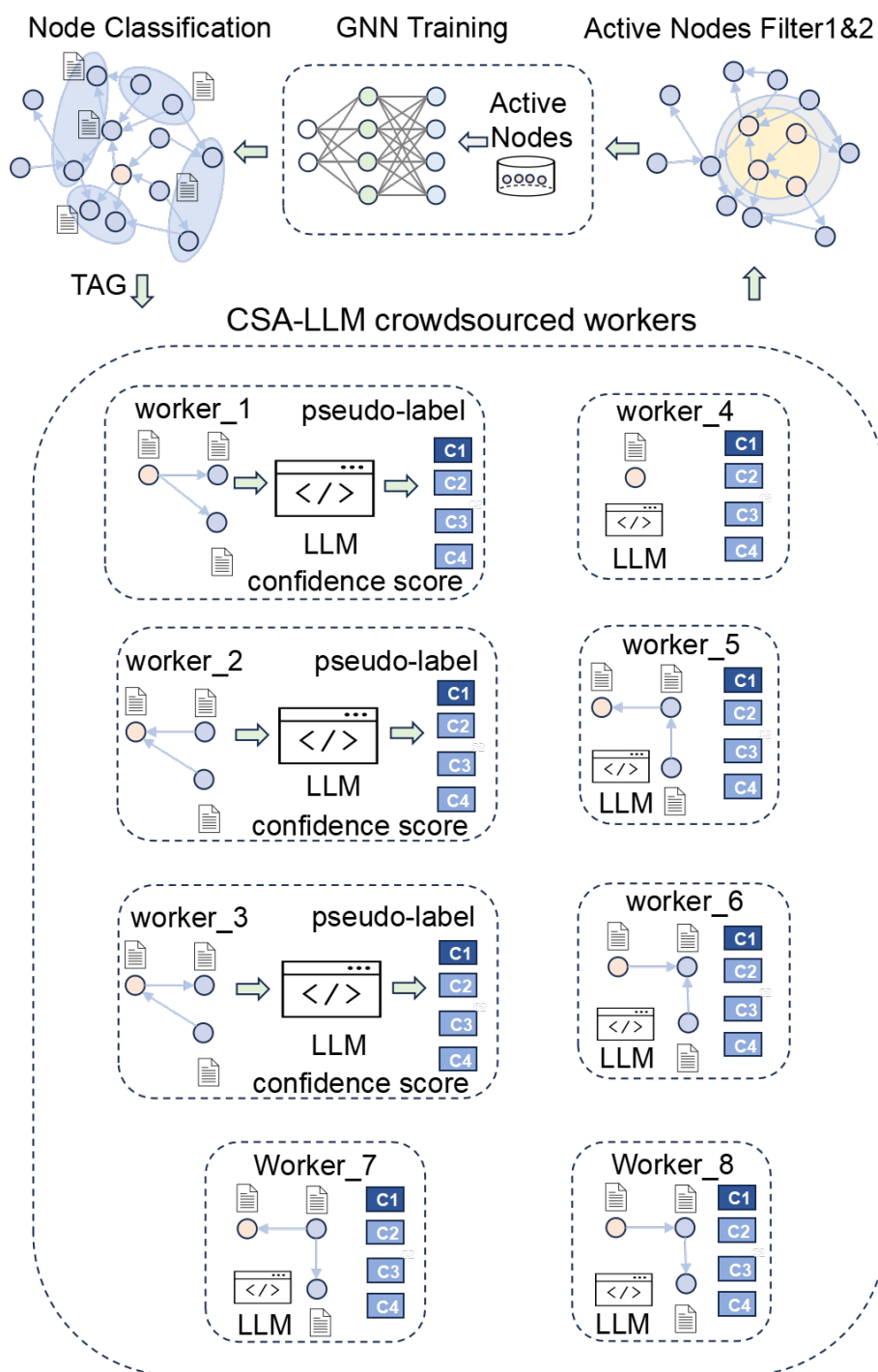


Figure 5.1: Crowdsourced annotation using LLMs to support GNN-based Node Classification.

The structural dependencies between nodes are encoded in the adjacency matrix $A \in \{0, 1\}^{n \times n}$, where an entry $A[i, j] = 1$ indicates a directed edge from node v_i to node v_j , while $A[j, i] = 1$ signifies a directed edge from v_j to node v_i . Given the inherently directional nature of edges in TAGs, the adjacency matrix A is typically asymmetric, accurately capturing the underlying connectivity patterns within the graph.

5.2.1 Structural Neighbor Typologies in Directed Graphs

A precise understanding of the structural relationships within a directed graph is crucial for improving the expressiveness and effectiveness of graph-based learning models. We define several key relational structures that characterize the connectivity of nodes in a directed setting:

- **Predecessors ($\text{pred}(v_i)$):** The set of nodes that have directed edges pointing to node v_i , formally defined as:

$$\text{pred}(v_i) = \{v_j \mid A[j, i] = 1\}. \quad (5.1)$$

- **Successors ($\text{succ}(v_i)$):** The set of nodes to which node v_i has outgoing edges, given by:

$$\text{succ}(v_i) = \{v_j \mid A[i, j] = 1\}. \quad (5.2)$$

- **Predecessors of Successors ($\text{pred}(\text{succ}(v_i))$):** The set of nodes that serve as predecessors to the successors of node v_i , defined as:

$$\text{pred}(\text{succ}(v_i)) = \{v_k \mid v_k \in \text{pred}(v_j), v_j \in \text{succ}(v_i)\}. \quad (5.3)$$

- **Successors of Predecessors ($\text{succ}(\text{pred}(v_i))$):** The set of nodes that act as successors to the predecessors of node v_i , expressed as:

$$\text{succ}(\text{pred}(v_i)) = \{v_k \mid v_k \in \text{succ}(v_j), v_j \in \text{pred}(v_i)\}. \quad (5.4)$$

- Second-Order Predecessors ($\text{pred}(\text{pred}(v_i))$): The set of nodes that are predecessors of node v_i 's predecessors, given by:

$$\text{pred}(\text{pred}(v_i)) = \{v_k \mid v_k \in \text{pred}(v_j), v_j \in \text{pred}(v_i)\}. \quad (5.5)$$

- Second-Order Successors ($\text{succ}(\text{succ}(v_i))$): The set of nodes that are successors of node v_i 's successors, formulated as:

$$\text{succ}(\text{succ}(v_i)) = \{v_k \mid v_k \in \text{succ}(v_j), v_j \in \text{succ}(v_i)\}. \quad (5.6)$$

These relational structures—such as hierarchical and transitive patterns in directed graphs—offer valuable contextual cues that reflect the semantic organization of the network. By systematically encoding these structural dependencies into the prompts, we enable the model to access richer contextual information during annotation. This integration not only enhances the confidence of label predictions but also improves the consistency and interpretability of the generated annotations.

5.3 Proposed Method

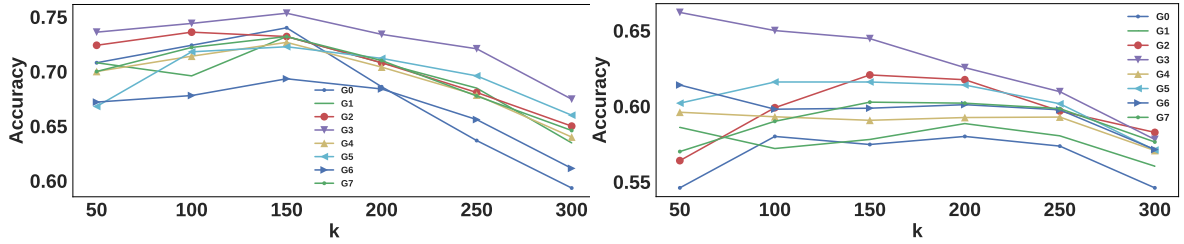


Figure 5.2: Performance evaluation of LLM as an annotation tool.

5.3.1 Strong-Tie-Guided Prompt Design for Crowdsourced LLM Annotation

In this section, we introduce a novel framework that combines structural homophily modeling with semantic reasoning via LLMs to address the annotation bottleneck

caused by structural-agnostic learning in graph-based systems. Our approach is formalized as a tuple $\langle \mathcal{G}, \Psi, \Phi \rangle$, where $\mathcal{G}$ denotes a directed text-attributed graph (TAG), Ψ represents the LLM-based annotator, and Φ serves as the homophily-aware aggregation operator. The central innovation lies in leveraging multi-scale structural contexts through directed neighborhood expansion, ensuring that both local interactions and semi-global patterns are preserved in the annotation process.

Extracting Contextual Strong-Tie Structures from Graphs To construct structurally meaningful prompt inputs, we formalize a compositional neighborhood extraction process for each node $v_i \in V$. We define a family of composite neighborhoods based on combinations of predecessor and successor mappings:

$$\mathcal{N}_i^{\text{comp}} = \bigcup_{\phi \in \mathcal{P}} \phi(v_i) \quad (5.7)$$

where $\mathcal{P}$ denotes the set of allowed path compositions, such as $\text{succ}(\text{pred}(v_i))$, $\text{pred}(\text{succ}(v_i))$, or deeper combinations like $\text{pred}^2(\text{succ}(v_i))$. These composite operators enable the construction of context-aware subgraphs that embed directional and structural semantics beyond simple 1-hop neighborhoods.

Building upon the composite neighborhood definitions, we generate a set of candidate subgraphs $\{\mathcal{G}_i^{(k)}\}_{k=1}^K$ by aggregating selected neighborhood patterns. Specifically, we define:

$$\mathcal{G}_i^k = \bigcup_{m \in \mathcal{M}_k} \mathcal{N}_i^{\text{comp}}, \quad \mathcal{M}_k \subseteq \mathbb{P}(\{1, 2\}) \quad (5.8)$$

where $\mathbb{P}(\cdot)$ denotes the power set over permitted hop configurations. This formulation flexibly combines 1-hop and 2-hop structural views to preserve key homophily and strong-tie characteristics. By leveraging such subgraph configurations, we ensure that prompt construction remains sensitive to the structural coherence of node relationships—providing a robust foundation for reliable LLM-based annotation.

Aggregating Homophily-Weighted Contexts Along Strong-Tie Paths Each subgraph $\mathcal{G}_i^k$ is assigned a context-aware textual representation, integrating node attributes via an attention-weighted aggregation scheme:

$$T(\mathcal{G}_i^k) = \sum_{v_j \in \mathcal{G}_i^k} \alpha_{ij} T_{v_j} \quad (5.9)$$

where α_{ij} is an attention coefficient that prioritizes structurally and semantically relevant nodes, computed as:

$$\alpha_{ij} = \frac{\exp(\mathbf{W}_a [T_{v_i} \oplus T_{v_j}])}{\sum_{v_k \in \mathcal{G}_i^k} \exp(\mathbf{W}_a [T_{v_i} \oplus T_{v_k}])} \quad (5.10)$$

Here, $\mathbf{W}_a$ is a heuristically or adaptively defined attention kernel, sensitive to both structural alignment (e.g., strong-tie memberships) and semantic similarity between nodes. Rather than relying solely on global learnable weights, this adaptive mechanism adjusts to the local topological patterns and attribute affinities within each subgraph, ensuring that the aggregated context faithfully captures the homophily-driven strong-tie structure. This formulation enables LLMs to process graph-derived textual inputs with built-in structural guidance, effectively grounding the annotation in both semantic proximity and graph connectivity.

Constructing Strong-Tie-Informed Prompts for LLM Annotation To effectively utilize LLMs, we synthesize structural and semantic cues into prompt templates:

$$q_i^k = \mathcal{P}_{\text{struct}}(E(\mathcal{G}_i^k)) \oplus \mathcal{P}_{\text{semantic}}(T(\mathcal{G}_i^k)) \quad (5.11)$$

where $E(\cdot)$ extracts edge directionality patterns, and $\mathcal{P}_{\{\cdot\}}$ are pre-defined templates that transform graph-based signals into structured textual input. The LLM annotator Ψ then produces probabilistic class predictions:

$$p(y_i | q_i^k) = \text{Softmax}(\Psi(q_i^k)), \quad y_i \in \mathcal{C} \quad (5.12)$$

Table 5.1: Example of a complete zero-shot annotation prompt for the CORA dataset [1]

Input: The content of the paper is **text from raw data** , which is cited by the paper(s) that **text from raw data**. The following categories are available: **list of categories**.

Task: Identify the category of this paper. Provide your seven most probable guesses along with a confidence score (0 to 100) for each, ensuring that the total confidence sums to 100. Format your response as follows: [{"answer": <your_first_answer>, "confidence": <confidence_for_first_answer>}, ...]

Output: [{"answer": <LLM's_first_answer>, "confidence": <LLM's confidence_for_first_answer>}, ...]

Through our proposed crowdsourced annotation framework, LLMs effectively integrate both textual and structural contexts to generate high-quality labels. By leveraging multi-level subgraph configurations, our approach captures a diverse range of contextual perspectives, mitigating potential biases that may arise in individual annotations. This structured methodology enhances annotation robustness, ensuring greater consistency and reliability in label inference.

Table 5.1 presents an example of our zero-shot annotation prompt, illustrating the structured query formulation used to guide LLM-based inference. Additionally, Figure 5.2 compares annotation accuracy across multiple graph topologies, demonstrating how different structural configurations impact the effectiveness of LLM-powered label generation. This evaluation underscores the adaptability of our framework in capturing graph-based contextual nuances while maintaining competitive performance.

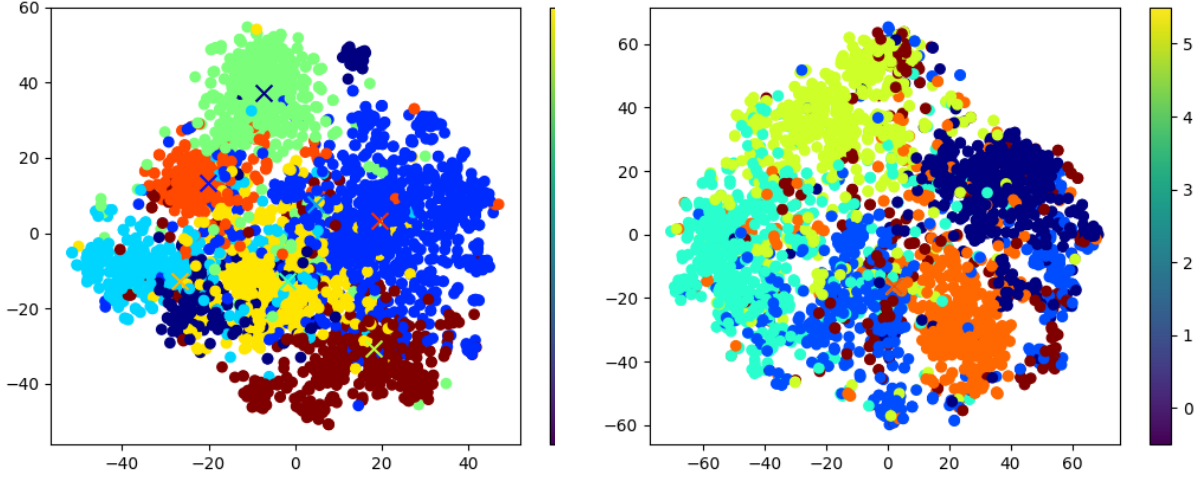


Figure 5.3: Clustering of nodes based on C-Density metric.

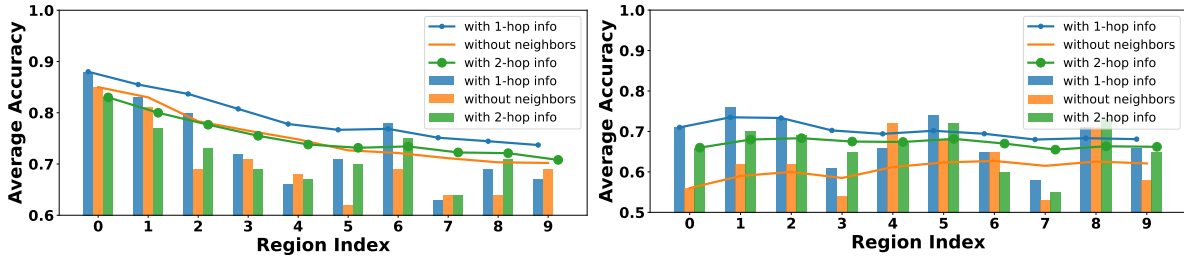


Figure 5.4: Accuracy trends with respect to C-Density distance and confidence score.

5.3.2 Two-Stage Information Bottleneck for Active Node Selection

Our node selection framework addresses the annotation efficiency challenge through a principled information-theoretic approach, formalizing active learning as a sequential information bottleneck optimization problem. The two-stage filtering process maximizes mutual information $I(\mathbf{y}_S; \mathbf{y}_U)$ between selected nodes S and unlabeled nodes U while minimizing annotation cost.

Stage I: Multi-Objective Preselection Based on Graph and Feature Centrality

The initial filter score synthesizes three complementary node centrality measures:

$$S_1(v_i) = \gamma\pi(v_i) + \lambda\psi(v_i) + \theta\delta(v_i) \quad (5.13)$$

where:

- $\pi(v_i)$: PageRank centrality [160], capturing global structural importance.
- $\psi(v_i)$: Cluster-centric density, measuring local consistency:

$$\psi(v_i) = [1 + \|\mathbf{x}_{v_i} - \mathbf{c}_{v_i}\|_2]^{-1}, \quad \mathbf{c}_{v_i} = \arg \min_{\mathbf{c}_k \in \mathcal{C}} \|\mathbf{x}_{v_i} - \mathbf{c}_k\|_2 \quad (5.14)$$

where $\mathcal{C}$ represents K-means cluster centers [161].

- $\delta(v_i)$: Degree centrality, encoding local connectivity.

The coefficients $\gamma, \lambda, \theta \in [0, 1]$ control modality weights (subject to $\gamma + \lambda + \theta = 1$). We select the candidate set $S^{(1)}$ based on an empirical percentile threshold, ensuring stability. For example:

$$S^{(1)} = \{v_i \mid S_1(v_i) \geq \xi_{0.7}\}, \quad \text{where } \xi_{0.7} \text{ is empirically tuned for robustness.} \quad (5.15)$$

Stage II: Entropy-Based Refinement for Informative Node Selection

The second stage optimizes information gain through entropy dynamics analysis:

$$S_2(v_i) = \underbrace{H(\tilde{\mathbf{y}}_{S^{(1)}}) - H(\tilde{\mathbf{y}}_{S^{(1)} \setminus \{v_i\}})}_{\Delta H(v_i)} + \underbrace{\mathbb{E}_{q_i \sim Q(v_i)}[p(\tilde{y}_i | q_i)]}_{\text{Confidence}(v_i)} \quad (5.16)$$

where:

- $\Delta H(v_i)$ quantifies the node's information influence via entropy reduction.

Algorithm 5: Active Nodes Filter Strategy

Input: Graph $G = (V, A)$, node features $\mathbf{X}$, annotation confidences $\mathbf{C}$,

hyperparameters γ, λ, η , number of nodes to select K

Output: Selected nodes for training

- 1: Compute PageRank scores $\mathbf{P}$, C-Density scores $\mathbf{D}$, and Degree scores $\mathbf{Deg}$
- 2: Compute first-stage filtering scores as:

$$S_1(v_i) = \gamma \cdot \mathbf{P}(v_i) + \lambda \cdot \mathbf{D}(v_i) + (1 - \gamma - \lambda) \cdot \mathbf{Deg}(v_i)$$

for all $v_i \in V$

- 3: Select top K nodes based on $S_1(v_i)$
- 4: Compute second-stage filtering scores:

$$S_2(v_i) = \text{COE}(v_i) + \text{Confidence}(v_i)$$

- 5: Select top $K \cdot \eta$ nodes based on $S_2(v_i)$
 - 6: **return** Selected nodes
-

- $\text{Confidence}(v_i)$ measures LLM annotation certainty through Monte Carlo dropout [162].

The final selected set S^* maximizes expected information gain:

$$S^* = \arg \max_{|S|=K} \sum_{v_i \in S} S_2(v_i) \quad \text{s.t.} \quad S \subseteq S^{(1)} \quad (5.17)$$

Our selection strategy satisfies two key properties. First, it ensures submodularity, as mutual information exhibits diminishing returns, allowing a $(1 - 1/e)$ -approximation to the optimal solution [163]. Second, it guarantees robustness, since the two-stage design tolerates ϵ -approximation errors in Stage I while bounding the final regret. The complete procedure for the active node filtering strategy is detailed in Algorithm 5. Figure 5.3 illustrates the clustering of nodes based on the proposed C-Density metric, which captures both structural compactness and semantic coherence. Figure 5.4 further shows how annotation accuracy varies with respect to C-Density distance and confidence score, validating the effectiveness of our selection mechanism.

5.4 Strong-Tie-Guided Homophily Propagation: A Theoretical Perspective

This section presents a rigorous theoretical analysis of homophily dominance in graph-based label propagation models. We demonstrate that under reasonable assumptions about conditional independence and transition probabilities, homophily dominance is preserved across multiple propagation steps.

5.4.1 Transition Matrix and Label Transition Probabilities

We consider a directed graph $G = (\mathcal{V}, \mathcal{E})$ where each node $v \in \mathcal{V}$ is assigned a label y_v from a finite set of labels $\mathcal{Y}$. The label propagation process can be formalized using a transition matrix $\mathbf{Q}$, which models the probability distribution of label transitions

between neighboring nodes. Although the graph is directed, the label transition matrix $\mathbf{Q}$ remains symmetric for analytical convenience. This is justified by the fact that label propagation is modeled based on probabilistic co-occurrence rather than edge directionality. The directionality is already encoded in the neighborhood structure (i.e., $\text{Pred}(v)$, $\text{Succ}(v)$), while the transition probabilities reflect how labels diffuse through the graph. The transition matrix is defined as:

$$\mathbf{Q} = (\alpha - \beta)\mathbf{I} + \beta\mathbf{J}, \quad (5.18)$$

where $\mathbf{I}$ is the identity matrix representing self-retention of labels, and $\mathbf{J}$ is the all-ones matrix capturing uniform label mixing. The parameter $\alpha \in [0, 1]$ represents the probability that a node retains its original label during propagation, while $\beta = \frac{1-\alpha}{|\mathcal{Y}|-1}$ defines the probability of transitioning to a different class, ensuring each row of $\mathbf{Q}$ forms a valid probability distribution. To illustrate, consider a multi-class setting with $|\mathcal{Y}| = 3$. The matrix $\mathbf{Q}$ takes the following form:

$$\mathbf{Q} = \begin{bmatrix} \alpha & \frac{1-\alpha}{2} & \frac{1-\alpha}{2} \\ \frac{1-\alpha}{2} & \alpha & \frac{1-\alpha}{2} \\ \frac{1-\alpha}{2} & \frac{1-\alpha}{2} & \alpha \end{bmatrix}.$$

In general, for $|\mathcal{Y}| = n$, the transition matrix $\mathbf{Q} \in \mathbb{R}^{n \times n}$ has the form:

$$\mathbf{Q} = \begin{bmatrix} \alpha & \beta & \cdots & \beta \\ \beta & \alpha & \cdots & \beta \\ \vdots & \vdots & \ddots & \vdots \\ \beta & \beta & \cdots & \alpha \end{bmatrix},$$

where each off-diagonal entry is $\beta = \frac{1-\alpha}{n-1}$, and each row sums to 1. This structure ensures that the diagonal dominance induced by α captures homophily, while the off-diagonal uniformity models symmetric label mixing across different classes.

5.4.2 Homophily Label Transition under 1-Hop Strong-Tie

In semi-supervised learning on graphs, the reliability of label propagation is closely tied to the preservation of homophily across neighborhoods. Strong ties—edges formed through structurally and semantically aligned neighbors—are hypothesized to preserve label consistency more effectively. Inspired by prior work on homophily-aware propagation [42], we examine the conditions under which homophily is maintained in 1-hop strong-tie neighborhoods, based on a probabilistic label transition matrix. The following theorem provides a sufficient condition for ensuring homophily dominance in this context.

Table 5.2: Comparative analysis of model performance across baselines and filtering techniques on Cora and Citeseer datasets.

	Cora	Citeseer
Zero-shot	67.00 $\pm$ 1.41	65.50 $\pm$ 3.53
Zero-shot with hops	71.75 $\pm$ 0.35	62.00 $\pm$ 1.41
Degree	68.67 $\pm$ 0.30	60.23 $\pm$ 0.54
Degree-W	69.86 $\pm$ 0.35	60.47 $\pm$ 0.49
DA-Degree	72.86 $\pm$ 0.27	60.33 $\pm$ 0.54
PS-Degree-W	70.92 $\pm$ 0.28	62.36 $\pm$ 0.69
DA-Degree-W	73.01 $\pm$ 0.24	61.29 $\pm$ 0.47
Pagerank	70.31 $\pm$ 0.42	61.21 $\pm$ 0.11
Pagerank-W	71.50 $\pm$ 0.44	61.97 $\pm$ 0.19
DA-Pagerank	74.34 $\pm$ 0.41	60.44 $\pm$ 0.40
PS-Pagerank-W	74.81 $\pm$ 0.37	63.27 $\pm$ 0.44
DA-Pagerank-W	75.62 $\pm$ 0.39	61.25 $\pm$ 0.45
Zero-shot-CSA	76.08 $\pm$ 0.50	67.40 $\pm$ 0.57

Theorem 6 (Homophily Dominance in 1-Hop Strong-Tie Neighborhoods). *Let node v have ground-truth label $y_v = i$, and let $w \in N_1(v)$ be a 1-hop neighbor connected via a*

strong tie. Assume that label transition behavior is governed by a symmetric transition matrix $\mathbf{Q} \in \mathbb{R}^{|\mathcal{Y}| \times |\mathcal{Y}|}$, where $\alpha \in [0, 1]$ denotes the probability of label preservation:

$$[\mathbf{Q}]_{i,i} = \alpha, \quad [\mathbf{Q}]_{i,j} = \frac{1 - \alpha}{|\mathcal{Y}| - 1}, \quad \forall j \neq i.$$

Then, the 1-hop neighborhood $N_1(v)$ is homophily-dominant in expectation if and only if

$$\alpha \geq \frac{1}{|\mathcal{Y}|}.$$

Proof. Given node v with label $y_v = i$, the probability that its neighbor $w \in N_1(v)$ retains the same label is:

$$P(y_w = i \mid y_v = i) = \alpha.$$

For any other label $j \neq i$, the transition probability is:

$$P(y_w = j \mid y_v = i) = \frac{1 - \alpha}{|\mathcal{Y}| - 1}.$$

To ensure homophily dominance in the 1-hop neighborhood, the probability of label agreement must be at least as large as the probability of disagreement:

$$\alpha \geq \frac{1 - \alpha}{|\mathcal{Y}| - 1}.$$

Rewriting the inequality:

$$\alpha(|\mathcal{Y}| - 1) \geq 1 - \alpha \quad \Rightarrow \quad \alpha|\mathcal{Y}| \geq 1 \quad \Rightarrow \quad \alpha \geq \frac{1}{|\mathcal{Y}|}.$$

Thus, for any $|\mathcal{Y}| \geq 2$, homophily dominance holds if and only if $\alpha \geq \frac{1}{|\mathcal{Y}|}$. ■

This result, aligned with the insights in [42], formally confirms that strong-tie connections within 1-hop neighborhoods preserve homophily as long as the self-transition probability α exceeds a dataset-dependent threshold. The threshold $\frac{1}{|\mathcal{Y}|}$ provides a quantitative criterion for evaluating when local propagation is likely to preserve label consistency.

5.4.3 Homophily Label Transition under 2-Hop Strong-Tie

Building upon the 1-hop analysis, we now consider homophily preservation in 2-hop neighborhoods. This setting captures the indirect propagation behavior that is critical in strong-tie guided semi-supervised learning frameworks, particularly under noisy or sparse supervision. Following the probabilistic label transition formulation in [42], we derive sufficient conditions for 2-hop homophily dominance.

Theorem 7 (Homophily Dominance in 2-Hop Strong-Tie Neighborhoods). *Let node v have ground-truth label $y_v = i$, and let $w \in N_2(v)$ be a 2-hop neighbor. Assume that label propagation follows a symmetric transition matrix $\mathbf{Q} \in \mathbb{R}^{|\mathcal{Y}| \times |\mathcal{Y}|}$, where $[\mathbf{Q}]_{i,i} = \alpha$ and $[\mathbf{Q}]_{i,j} = \beta = \frac{1-\alpha}{|\mathcal{Y}|-1}$ for $j \neq i$. Then, homophily dominance is guaranteed at 2 hops, i.e.,*

$$[\mathbf{Q}^2]_{i,i} \geq [\mathbf{Q}^2]_{i,j}, \quad \forall j \neq i.$$

Proof. The probability that node $w \in N_2(v)$ retains the label $y_v = i$ is:

$$P(y_w = k \mid y_v = i) = \sum_{j \in \mathcal{Y}} P(y_w = k \mid y_u = j) P(y_u = j \mid y_v = i),$$

which translates into:

$$P(y_w = k \mid y_v = i) = [\mathbf{Q}^2]_{i,k}.$$

We now explicitly compute the values:

$$\begin{aligned} [\mathbf{Q}^2]_{i,i} &= \alpha^2 + (|\mathcal{Y}| - 1)\beta^2, \\ [\mathbf{Q}^2]_{i,j} &= 2\alpha\beta + (|\mathcal{Y}| - 2)\beta^2, \quad j \neq i. \end{aligned}$$

Taking the difference:

$$[\mathbf{Q}^2]_{i,i} - [\mathbf{Q}^2]_{i,j} = (\alpha - \beta)^2.$$

Since $\alpha > \beta$ for any $\alpha > \frac{1}{|\mathcal{Y}|}$, it follows that $(\alpha - \beta)^2 > 0$. Therefore, homophily dominance holds in expectation for all 2-hop neighbors. $\blacksquare$

This result confirms that label consistency remains stronger along 2-hop strong-tie paths than across divergent labels. Compared to 1-hop propagation, 2-hop neighborhoods provide a richer context for enforcing homophily-aware propagation, which

supports more stable semi-supervised learning. These insights are instrumental in designing robust annotation systems that leverage structural priors, as emphasized in [42].

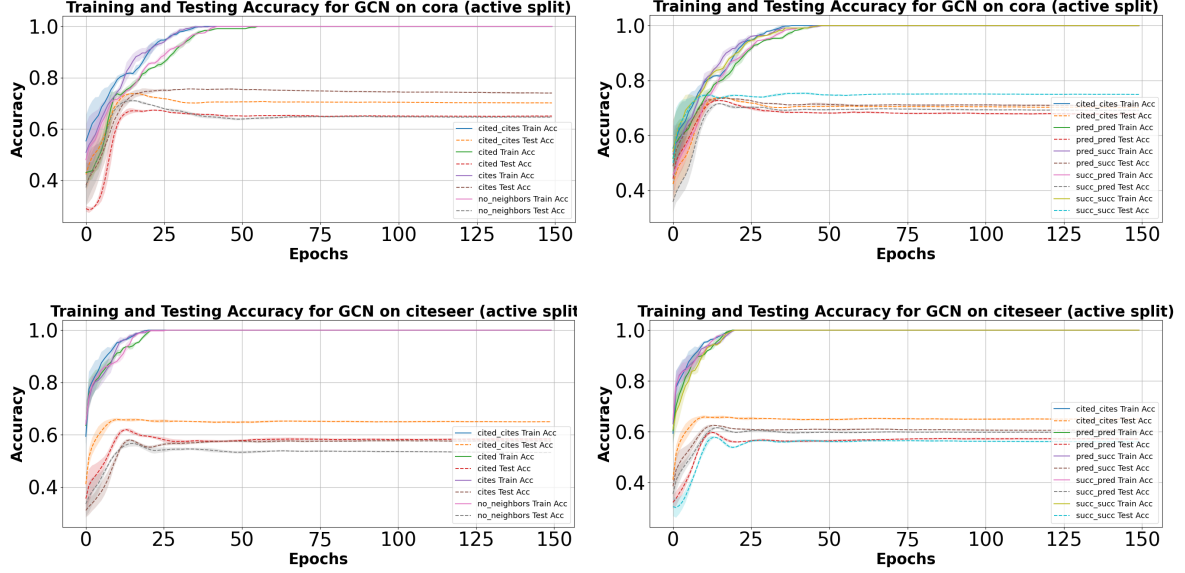


Figure 5.5: Training vs. testing accuracy for GCN under CSA-LLM framework.

5.4.4 Homophily Label Transition under Multi-Hop Strong-Tie

To generalize the homophily dominance property to multi-hop propagation, we analyze the label transition behavior using the spectral decomposition of the transition matrix $\mathbf{Q}$. This enables a closed-form expression for $\mathbf{Q}^h$, capturing the influence of repeated label propagation over h steps.

Theorem 8 (Homophily Dominance in Multi-Hop Strong-Tie Propagation). *Let $\mathbf{Q} \in \mathbb{R}^{|\mathcal{Y}| \times |\mathcal{Y}|}$ denote the label transition matrix, where diagonal entries are α and off-diagonal entries are $\beta = \frac{1-\alpha}{|\mathcal{Y}|-1}$. Then for any $h \geq 1$, the h -step transition matrix satisfies:*

$$[\mathbf{Q}^h]_{i,i} > [\mathbf{Q}^h]_{i,j}, \quad \forall j \neq i.$$

Table 5.3: Hyperparameters and Test Accuracy for Cora and Citeseer Datasets

Dataset	Hyperparameters	Test Acc
CORA $\eta = 0.15$	$\gamma = 0.00, \lambda = 0.80$	75.71 ± 0.68
	$\gamma = 0.01, \lambda = 0.79$	74.96 ± 0.42
	$\gamma = 0.02, \lambda = 0.78$	76.08 ± 0.50
	$\gamma = 0.03, \lambda = 0.77$	74.49 ± 0.20
	$\gamma = 0.04, \lambda = 0.76$	74.99 ± 0.29
CITeseer $\eta = 0.20$	$\gamma = 0.07, \lambda = 0.73$	65.40 ± 0.49
	$\gamma = 0.08, \lambda = 0.72$	66.28 ± 0.65
	$\gamma = 0.09, \lambda = 0.71$	67.40 ± 0.57
	$\gamma = 0.10, \lambda = 0.70$	66.32 ± 0.56
	$\gamma = 0.11, \lambda = 0.69$	64.01 ± 0.57

Proof. By spectral decomposition, the h -step transition matrix can be expressed as:

$$\mathbf{Q}^h = (\alpha - \beta)^h \mathbf{I} + \frac{1 - (\alpha - \beta)^h}{|\mathcal{Y}|} \mathbf{J},$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{J}$ is the all-ones matrix. For diagonal elements $i = j$, we have:

$$[\mathbf{Q}^h]_{i,i} = (\alpha - \beta)^h + \frac{1 - (\alpha - \beta)^h}{|\mathcal{Y}|}.$$

For off-diagonal elements $i \neq j$, we have:

$$[\mathbf{Q}^h]_{i,j} = \frac{1 - (\alpha - \beta)^h}{|\mathcal{Y}|}.$$

Taking the difference:

$$[\mathbf{Q}^h]_{i,i} - [\mathbf{Q}^h]_{i,j} = (\alpha - \beta)^h > 0,$$

since $\alpha > \beta$ by assumption. Therefore, homophily dominance is preserved at all propagation steps $h \geq 1$. ■

This result confirms that label propagation across multi-hop strong-tie paths consistently favors homophilic label retention, thereby reinforcing classification consistency

in iterative message-passing frameworks. It also provides a theoretical guarantee for designing deeper, yet stable, graph-based learning architectures.

5.5 Evaluation

We evaluate the effectiveness of CSA-LLM by benchmarking it against multiple baseline methods introduced in [115] and various filtering strategies discussed in [1]. The comparative results, summarized in Table 5.2, demonstrate that CSA-LLM consistently outperforms existing approaches across both the Cora and Citeseer datasets. This superior performance highlights the robustness of our proposed framework in leveraging structural and semantic information for label-free graph annotation.

To further analyze the influence of hyperparameter configurations on the predictive accuracy of GNN models, we conduct a comprehensive study on different parameter settings. Table 5.3 provides an overview of the hyperparameters utilized in our experiments, underscoring the critical role of parameter optimization in graph-based learning tasks. Our findings reveal that careful tuning of hyperparameters significantly enhances model performance, emphasizing the importance of selecting optimal configurations to maximize learning efficiency.

Furthermore, we assess the learning dynamics of GNN models by tracking both training and testing accuracy across multiple epochs. Figure 5.5 presents the accuracy progression over time, offering insights into the model’s convergence behavior, learning stability, and generalization capability. This empirical evaluation provides a deeper understanding of how CSA-LLM adapts to graph-based learning scenarios and maintains stable performance throughout the training process.

In addition to evaluating predictive accuracy, we consider the cost-efficiency of our crowdsourced annotation approach. By leveraging ChatGPT 3.5 for label inference, we successfully maintain an annotation budget of approximately two dollars per dataset. This demonstrates the feasibility of integrating LLM-powered annotation methods into large-scale graph learning applications while ensuring cost-effectiveness

and scalability.

5.6 Summary

CSA-LLM enhances graph annotation by integrating strong-tie structures into LLM-based labeling, improving annotation accuracy and GNN performance. By leveraging homophily-driven relationships, it ensures structural consistency, reducing inconsistencies in label prediction and classification. Additionally, CSA-LLM refines annotation prioritization through structural filtering, optimizing efficiency and reliability.

Our findings highlight that structural properties play a crucial role in annotation stability. Specifically, modeling strong ties helps bridge local semantic similarity with global topological context, allowing LLMs to produce more coherent and contextually grounded annotations. This integration of structure into prompt design significantly reduces noise in weak supervision, particularly in low-label or structurally sparse settings. As previously discussed, the computational cost for CSA-LLM is minimal, with a testing cost of only \$2 on the evaluation data. The primary computational cost arises from the data of active node neighbors, which is manageable in the proposed framework.

Although developed for standard attributed graphs, this framework offers broader applicability. Its emphasis on structurally-aware prompts and homophily-based inference naturally extends to heterogeneous graphs with multi-relational edges, where edge semantics must be embedded into annotation decisions. Moreover, domains such as accounting systems—which often resemble directed interaction graphs—may benefit from similar structure-guided labeling strategies.

Overall, CSA-LLM bridges human-in-the-loop annotation with automated learning, establishing a scalable and resilient foundation for improving the trustworthiness of graph-based learning models.

Chapter 6

Conclusion

Ensuring node labeling stability is fundamental to graph-based learning systems, especially in tasks like trust evaluation, fraud detection, and decentralized platforms. In unsupervised learning, where labels are derived from structural patterns, even slight perturbations can significantly disrupt trust estimation. Our analysis shows that adversarial agents often exploit strong-tie paths to manipulate these systems. Building on this insight, we extend the investigation to semi-supervised learning and find that label propagation is similarly vulnerable. To counter this, we propose an adaptive defense framework that leverages strong-tie semantics to adjust propagation and improve robustness. Furthermore, the strong-tie problem can be viewed as a constrained shortest path problem. This problem can be reduced and solved using machine learning methods, or approximated by heuristic algorithms. This ensures that even on extremely large graphs, strong ties can still be effectively mined. Finally, we explore LLM-assisted annotation. While scalable, LLM-based methods may produce unstable labels without structural guidance. By embedding strong-tie constraints into prompt design, we improve annotation consistency and model reliability across graph learning tasks.

Exposing Structural Vulnerabilities in Node Labeling through Strong-Tie Exploitation

In trust-centric learning, unsupervised models such as the Fairness and Goodness Algorithm (FGA) and REV2 generate node-level scores based solely on structural interactions. We show that these propagation processes introduce biases that adversaries can exploit. Specifically, indirect Sybil attacks—though limited in access—can significantly disrupt trust prediction by targeting key structural formations. Our findings reveal that adversarial behaviors tend to favor strong-tie pathways, creating a structural vulnerability wherein small, indirect perturbations produce outsized shifts in trust scores. To better understand and quantify this effect, we conduct formal and empirical studies showing that attackers can manipulate trust predictions by modifying key edges along preferential paths. This reveals a core vulnerability in many trust scoring systems: they depend heavily on structural patterns, but lack the robustness to defend against such targeted manipulations.

Defending Against Structural Poisoning via Strong-Tie-Aware Adaptive Graph Learning

To counter structural poisoning attacks, we propose the Graph Adaptive Neural Network (GANN), a defense architecture that incorporates both propagation adaptability and structural validation. GANN utilizes an adaptive propagation mechanism over directed edges, allowing robust node classification. The classifier dynamically adjusts its propagation scheme in response to perturbations, strengthening resilience against adversarial injection. GANN first identifies strong-tie relationships between the training set and structurally sensitive nodes, forming dynamic trust zones that reflect the underlying propagation patterns. These zones enable a more fine-grained analysis of adversarial behavior, revealing attacker preferences for certain tie types

during simulation. Based on these insights, GANN deploys an adaptive propagation mechanism that dynamically adjusts message flow, redirecting it away from high-risk paths. This reconfiguration of strong-tie structures helps mitigate the influence of adversarial perturbations and improves the stability of node labeling under attack.

Improving LLM-Guided Annotation Stability with Strong-Tie Structural Priors

Automated annotation using LLMs offers a scalable alternative to manual labeling, especially in domains where labels are sparse or expensive. However, LLM-driven annotation lacks structural grounding, making it prone to label noise and inconsistency. This dissertation introduces a strong-tie-guided prompt engineering approach, designed to align textual annotation with graph topology. The proposed framework consists of two major components: a structurally-informed prompt design and an active node selection mechanism. First, by extracting directed neighborhoods and encoding strong-tie context into prompt design, the framework guides LLMs to generate labels that align with underlying graph structure. Second, a two-stage node selection strategy identifies structurally informative nodes to serve as training anchors, improving annotation efficiency under limited supervision. Experimental results demonstrate that this approach significantly enhances both the accuracy of LLM-generated annotations and the downstream performance of GNNs.

Summary of Contributions

This dissertation presents a comprehensive investigation into the vulnerabilities and defenses of node labeling in directed graph learning systems. The contributions are organized around three core dimensions:

- **Strong-Tie Exploitation in Node Labeling for Unsupervised Trust**

Prediction: We formally analyze how indirect Sybil attacks exploit propagation bias in trust-based models such as FGA and REV2. Our findings show that adversaries leverage strong-tie structures and preferential paths to distort node-level trust scores, even with minimal edge modifications. This highlights the structural fragility of unsupervised and even semi-supervised trust estimation under adversarial conditions.

- **Strong-Tie-Aware Defense via Graph Adaptive Neural Network:** GANN defends against adversarial attacks by modeling the relationship between training nodes and vulnerable graph partitions. It identifies trust zones based on structural risk, and adaptively adjusts label propagation for low-confidence nodes. With fuzzy logic refinement, GANN weakens attacker influence and improves labeling stability under perturbations.
- **Strong-Tie-Guided Prompt Engineering for Stable LLM-Based Graph Annotation:** To improve labeling accuracy and reduce annotation variance, we develop a prompt engineering framework that incorporates strong-tie priors into LLM input generation. This approach enhances alignment between textual representations and structural semantics, enabling more stable and reliable graph annotation at scale.

Collectively, these contributions chart a path toward robust and interpretable trust prediction in adversarial graph environments, with far-reaching implications for the security and integrity of social, financial, and decentralized ecosystems.

References

- [1] Z. Chen, H. Mao, H. Wen, H. Han, W. Jin, H. Zhang, H. Liu, and J. Tang, “Label-free node classification on graphs with large language models (llms),” *arXiv preprint arXiv:2310.04668*, 2023.
- [2] X. Li, L. Sun, M. Ling, and Y. Peng, “A survey of graph neural network based recommendation in social networks,” *Neurocomputing*, vol. 549, p. 126441, 2023.
- [3] X.-M. Zhang, L. Liang, L. Liu, and M.-J. Tang, “Graph neural networks and their current applications in bioinformatics,” *Frontiers in genetics*, vol. 12, p. 690049, 2021.
- [4] P. Bansal and A. Sharma, “Large language models as annotators: Enhancing generalization of nlp models at minimal cost,” *arXiv preprint arXiv:2306.15766*, 2023.
- [5] B. Ding, C. Qin, L. Liu, Y. K. Chia, S. Joty, B. Li, and L. Bing, “Is gpt-3 a good data annotator?,” *arXiv preprint arXiv:2212.10450*, 2022.
- [6] K. Guo, Z. Liu, Z. Chen, H. Wen, W. Jin, J. Tang, and Y. Chang, “Learning on graphs with large language models (llms): A deep dive into model robustness,” *arXiv preprint arXiv:2407.12068*, 2024.
- [7] X. He, Z. Lin, Y. Gong, A. Jin, H. Zhang, C. Lin, J. Jiao, S. M. Yiu, N. Duan, W. Chen, *et al.*, “Annollm: Making large language models to be better crowd-sourced annotators,” *arXiv preprint arXiv:2303.16854*, 2023.

-
- [8] J. P. Cruz and Y. Kaji, “The bitcoin network as platform for transorganizational attribute authentication,” in *The Third International Conference on Building and Exploring Web Based Environments (WEB)*, At Rome, Italy, 2015.
 - [9] J. Leskovec, D. Huttenlocher, and J. Kleinberg, “Signed networks in social media,” in *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1361–1370, 2010.
 - [10] S. Kumar, F. Spezzano, V. Subrahmanian, and C. Faloutsos, “Edge weight prediction in weighted signed networks,” in *2016 IEEE 16th international conference on data mining (ICDM)*, pp. 221–230, IEEE, 2016.
 - [11] H. Zhang, C. Xu, and J. Zhang, “Exploiting trust and distrust information to combat sybil attack in online social networks,” in *Trust Management VIII: 8th IFIP WG 11.11 International Conference, IFIPTM 2014, Singapore, July 7-10, 2014. Proceedings 8*, pp. 77–92, Springer, 2014.
 - [12] C. R. Jones, D. Ferreday, and V. Hodgson, “Networked learning a relational approach: weak and strong ties,” *Journal of computer assisted learning*, vol. 24, no. 2, pp. 90–102, 2008.
 - [13] Z. Gu, J. Li, and L. Chen, “Are all edges necessary? a unified framework for graph purification,” *arXiv preprint arXiv:2211.05184*, 2022.
 - [14] C. Xiong, D. Johnson, and J. J. Corso, “Spectral active clustering via purification of the k-nearest neighbor graph,” in *Proceedings of European conference on data mining*, vol. 1, p. 3, 2012.
 - [15] X. Dong, D. Thanou, L. Toni, M. Bronstein, and P. Frossard, “Graph signal processing for machine learning: A review and new perspectives,” *IEEE Signal processing magazine*, vol. 37, no. 6, pp. 117–127, 2020.
 - [16] Y. Bu, Y. Zhu, L. Geng, and K. Zhou, “Uncovering strong ties: a study of indirect sybil attack on signed social network,” in *ICASSP 2024-2024 IEEE*

- International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4535–4539, IEEE, 2024.
- [17] Y. Bu, Y. Zhu, L. Geng, and K. Zhou, “Unleashing the power of indirect attacks against trust prediction via preferential path,” *Knowledge and Information Systems*, pp. 1–28, 2025.
- [18] W. Sherchan, S. Nepal, and C. Paris, “A survey of trust in social networks,” *ACM Computing Surveys (CSUR)*, vol. 45, no. 4, pp. 1–33, 2013.
- [19] V. Buskens, *Social networks and trust*. Kluwer Academic Publishers, 2002.
- [20] S. Adali, R. Escriva, M. K. Goldberg, M. Hayvanovych, M. Magdon-Ismael, B. K. Szymanski, W. A. Wallace, and G. Williams, “Measuring behavioral trust in social networks,” in *2010 IEEE international conference on intelligence and security informatics*, pp. 150–152, IEEE, 2010.
- [21] K. Zolfaghar and A. Aghaie, “Mining trust and distrust relationships in social web applications,” in *Proceedings of the 2010 IEEE 6th international conference on intelligent computer communication and processing*, pp. 73–80, IEEE, 2010.
- [22] J. Tang, Y. Chang, C. Aggarwal, and H. Liu, “A survey of signed network mining in social media,” *ACM Computing Surveys (CSUR)*, vol. 49, no. 3, pp. 1–37, 2016.
- [23] T. Lizurej, T. Michalak, and S. Dziembowski, “On manipulating weight predictions in signed weighted networks,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, pp. 5222–5229, 2023.
- [24] N. Vallarano, C. J. Tessone, and T. Squartini, “Bitcoin transaction networks: an overview of recent results,” *Frontiers in Physics*, vol. 8, p. 286, 2020.

-
- [25] D. S. Evans, “Economic aspects of bitcoin and other decentralized public-ledger currency platforms,” *University of Chicago Coase-Sandor Institute for Law & Economics Research Paper*, no. 685, 2014.
- [26] D. A. Wijaya, “Extending asset management system functionality in bitcoin platform,” in *2016 International Conference on Computer, Control, Informatics and its Applications (IC3INA)*, pp. 97–101, IEEE, 2016.
- [27] A. Mohaisen and J. Kim, “The sybil attacks and defenses: a survey,” *arXiv preprint arXiv:1312.6349*, 2013.
- [28] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman, “Sybilguard: defending against sybil attacks via social networks,” in *Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*, pp. 267–278, 2006.
- [29] R. Gunturu, “Survey of sybil attacks in social networks,” *arXiv preprint arXiv:1504.05522*, 2015.
- [30] J. R. Douceur, “The sybil attack,” in *International workshop on peer-to-peer systems*, pp. 251–260, Springer, 2002.
- [31] M. T. Godziszewski, T. P. Michalak, M. Waniek, T. Rahwan, K. Zhou, and Y. Zhu, “Attacking similarity-based sign prediction,” in *2021 IEEE International Conference on Data Mining (ICDM)*, pp. 1072–1077, IEEE, 2021.
- [32] Y. Zhu, T. Michalak, X. Luo, X. Zhang, and K. Zhou, “Towards secrecy-aware attacks against trust prediction in signed social networks,” *IEEE Transactions on Information Forensics and Security*, 2024.
- [33] B. Wang and N. Z. Gong, “Attacking graph-based classification via manipulating the graph structure,” in *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pp. 2023–2040, 2019.

- [34] D. Zügner, A. Akbarnejad, and S. Günnemann, “Adversarial attacks on graph neural networks,” 2019.
- [35] Y. Zhu, Y. Lai, K. Zhao, X. Luo, M. Yuan, J. Ren, and K. Zhou, “Binarized attack: Structural poisoning attacks to graph-based anomaly detection,” in *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pp. 14–26, IEEE, 2022.
- [36] H. Dai, H. Li, T. Tian, X. Huang, L. Wang, J. Zhu, and L. Song, “Adversarial attack on graph structured data,” in *International conference on machine learning*, pp. 1115–1124, PMLR, 2018.
- [37] L. Sun, Y. Dou, C. Yang, K. Zhang, J. Wang, P. S. Yu, L. He, and B. Li, “Adversarial attack and defense on graph data: A survey,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 8, pp. 7693–7711, 2022.
- [38] R. West, H. S. Paskov, J. Leskovec, and C. Potts, “Exploiting social network structure for person-to-person sentiment analysis,” *Transactions of the Association for Computational Linguistics*, vol. 2, pp. 297–310, 2014.
- [39] A. Shafahi, M. Najibi, Z. Xu, J. Dickerson, L. S. Davis, and T. Goldstein, “Universal adversarial training,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 5636–5643, 2020.
- [40] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. March, and V. Lempitsky, “Domain-adversarial training of neural networks,” *Journal of machine learning research*, vol. 17, no. 59, pp. 1–35, 2016.
- [41] M. Gu, G. Yang, S. Zhou, N. Ma, J. Chen, Q. Tan, M. Liu, and J. Bu, “Homophily-enhanced structure learning for graph clustering,” in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pp. 577–586, 2023.

-
- [42] J. Zhu, Y. Yan, L. Zhao, M. Heimann, L. Akoglu, and D. Koutra, “Beyond homophily in graph neural networks: Current limitations and effective designs,” *Advances in neural information processing systems*, vol. 33, pp. 7793–7804, 2020.
 - [43] F. Liu, X. Li, Y. Ding, H. Zhao, X. Liu, Y. Ma, and B. Tang, “A social network-based trust-aware propagation model for p2p systems,” *Knowledge-Based Systems*, vol. 41, pp. 8–15, 2013.
 - [44] R. Zhang and Y. Mao, “Trust prediction via belief propagation,” *ACM Transactions on Information Systems (TOIS)*, vol. 32, no. 3, pp. 1–27, 2014.
 - [45] G. Jethava and U. P. Rao, “Exploring security and trust mechanisms in online social networks: An extensive review,” *Computers & Security*, p. 103790, 2024.
 - [46] S. Kumar, B. Hooi, D. Makhija, M. Kumar, C. Faloutsos, and V. Subrahmanian, “Rev2: Fraudulent user prediction in rating platforms,” in *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pp. 333–341, 2018.
 - [47] J. Tang, Y. Chang, C. Aggarwal, and H. Liu, “A survey of signed network mining in social media,” *ACM Computing Surveys (CSUR)*, vol. 49, no. 3, pp. 1–37, 2016.
 - [48] G. Jethava and U. P. Rao, “A novel trust prediction approach for online social networks based on multifaceted feature similarity,” *Cluster Computing*, vol. 25, no. 6, pp. 3829–3843, 2022.
 - [49] G. Jethava and U. P. Rao, “An interaction-based and graph-based hybrid approach to evaluate trust in online social networks (osns),” *Arabian Journal for Science and Engineering*, vol. 47, no. 8, pp. 9615–9628, 2022.
 - [50] P. Berkhin, “A survey on pagerank computing,” *Internet mathematics*, vol. 2, no. 1, pp. 73–120, 2005.

- [51] J. M. Kleinberg, “Authoritative sources in a hyperlinked environment,” *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999.
- [52] A. Mishra and A. Bhattacharya, “Finding the bias and prestige of nodes in networks based on trust scores,” in *Proceedings of the 20th international conference on World wide web*, pp. 567–576, 2011.
- [53] R.-H. Li, J. Xu Yu, X. Huang, and H. Cheng, “Robust reputation-based ranking on bipartite rating networks,” in *Proceedings of the 2012 SIAM international conference on data mining*, pp. 612–623, SIAM, 2012.
- [54] L. Akoglu, R. Chandy, and C. Faloutsos, “Opinion fraud detection in online reviews by network effects,” in *Proceedings of the international AAAI conference on web and social media*, vol. 7, pp. 2–11, 2013.
- [55] M. Shahriari and M. Jalili, “Ranking nodes in signed social networks,” *Social network analysis and mining*, vol. 4, pp. 1–12, 2014.
- [56] V. A. Traag, Y. E. Nesterov, and P. Van Dooren, “Exponential ranking: taking into account negative links,” in *International Conference on Social Informatics*, pp. 192–202, Springer, 2010.
- [57] Z. Wu, C. C. Aggarwal, and J. Sun, “The troll-trust model for ranking in signed networks,” in *Proceedings of the Ninth ACM international conference on Web Search and Data Mining*, pp. 447–456, 2016.
- [58] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins, “Propagation of trust and distrust,” in *Proceedings of the 13th international conference on World Wide Web*, pp. 403–412, 2004.
- [59] W. Lee, Y.-C. Lee, D. Lee, and S.-W. Kim, “Look before you leap: Confirming edge signs in random walk with restart for personalized node ranking in signed networks,” in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 143–152, 2021.

-
- [60] M. Khan, G. B. Mello, P. Engelstad, L. Habib, and A. Yazidi, “Hits-gnn: A simplified propagation scheme for graph neural networks,” in *2022 IEEE International Conference on Big Data (Big Data)*, pp. 5514–5519, IEEE, 2022.
- [61] M. Khan, G. B. Mello, L. Habib, P. Engelstad, and A. Yazidi, “Hits-based propagation paradigm for graph neural networks,” *ACM Transactions on Knowledge Discovery from Data*, vol. 18, no. 4, pp. 1–23, 2024.
- [62] S.-H. Yang, A. J. Smola, B. Long, H. Zha, and Y. Chang, “Friend or frenemy? predicting signed ties in social networks,” in *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pp. 555–564, 2012.
- [63] Y. Wang, X. Wang, J. Tang, W. Zuo, and G. Cai, “Modeling status theory in trust prediction,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, 2015.
- [64] Y. Jing, H. Wang, K. Shao, and X. Huo, “Relation representation learning via signed graph mutual information maximization for trust prediction,” *Symmetry*, vol. 13, no. 1, p. 115, 2021.
- [65] S. Knapskog *et al.*, “A metric for trusted systems,” in *Proceedings of the 21st National Security Conference*, pp. 16–29, Citeseer, 1998.
- [66] M. De Cock and P. P. Da Silva, “A many valued representation and propagation of trust and distrust,” in *Fuzzy Logic and Applications: 6th International Workshop, WILF 2005, Crema, Italy, September 15-17, 2005, Revised Selected Papers 6*, pp. 114–120, Springer, 2006.
- [67] P. Victor, C. Cornelis, M. De Cock, and P. Pinheiro da Silva, “Towards a provenance-preserving trust model in agent networks,” in *WWW2006 Conference Proceedings, Special Interest Tracks, Posters and Workshops*, 2006.

- [68] S. Malik, V. Dedeoglu, S. S. Kanhere, and R. Jurdak, “Trustchain: Trust management in blockchain and iot supported supply chains,” in *2019 IEEE international conference on blockchain (Blockchain)*, pp. 184–193, IEEE, 2019.
- [69] H. Dai, H. Li, T. Tian, X. Huang, L. Wang, J. Zhu, and L. Song, “Adversarial attack on graph structured data,” in *International conference on machine learning*, pp. 1115–1124, PMLR, 2018.
- [70] B. Wu, Z. Zhu, L. Liu, Q. Liu, Z. He, and S. Lyu, “Attacks in adversarial machine learning: A systematic survey from the life-cycle perspective,” *arXiv preprint arXiv:2302.09457*, 2023.
- [71] H. Wu, C. Wang, Y. Tyshetskiy, A. Docherty, K. Lu, and L. Zhu, “Adversarial examples on graph data: Deep insights into attack and defense,” *arXiv preprint arXiv:1903.01610*, 2019.
- [72] J. Zhang and C. Li, “Adversarial examples: Opportunities and challenges,” *IEEE transactions on neural networks and learning systems*, vol. 31, no. 7, pp. 2578–2593, 2019.
- [73] C. Xiao, B. Li, J.-Y. Zhu, W. He, M. Liu, and D. Song, “Generating adversarial examples with adversarial networks,” *arXiv preprint arXiv:1801.02610*, 2018.
- [74] P. Otte, M. de Vos, and J. Pouwelse, “Trustchain: A sybil-resistant scalable blockchain,” *Future Generation Computer Systems*, vol. 107, pp. 770–780, 2020.
- [75] R. Bapna, A. Gupta, S. Rice, and A. Sundararajan, “Trust and the strength of ties in online social networks,” *MIS quarterly*, vol. 41, no. 1, pp. 115–130, 2017.
- [76] D. Zügner and S. Günnemann, “Certifiable robustness and robust training for graph convolutional networks,” in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 246–256, 2019.

-
- [77] G. Liu, X. Huang, and X. Yi, “Adversarial label poisoning attack on graph neural networks via label propagation,” in *European Conference on Computer Vision*, pp. 227–243, Springer, 2022.
- [78] H. Han, X. Liu, F. Shi, M. Torkamani, C. Aggarwal, and J. Tang, “Towards label position bias in graph neural networks,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [79] M. Fang, G. Yang, N. Z. Gong, and J. Liu, “Poisoning attacks to graph-based recommender systems,” in *Proceedings of the 34th annual computer security applications conference*, pp. 381–392, 2018.
- [80] E. Dai, T. Zhao, H. Zhu, J. Xu, Z. Guo, H. Liu, J. Tang, and S. Wang, “A comprehensive survey on trustworthy graph neural networks: Privacy, robustness, fairness, and explainability,” *arXiv preprint arXiv:2204.08570*, 2022.
- [81] J. Steinhardt, P. W. W. Koh, and P. S. Liang, “Certified defenses for data poisoning attacks,” *Advances in neural information processing systems*, vol. 30, 2017.
- [82] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards deep learning models resistant to adversarial attacks,” *arXiv preprint arXiv:1706.06083*, 2017.
- [83] L. E. Celis, A. Mehrotra, and N. Vishnoi, “Fair classification with adversarial perturbations,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 8158–8171, 2021.
- [84] C. Liu, H. Huang, Y. Xing, and X. Zuo, “Query-based and unnoticeable graph injection attack from neighborhood perspective,” *arXiv preprint arXiv:2502.01936*, 2025.

- [85] C. Zhang, S. Zhang, J. J. Yu, and S. Yu, “Sam: Query-efficient adversarial attacks against graph neural networks,” *ACM Transactions on Privacy and Security*, vol. 26, no. 4, pp. 1–19, 2023.
- [86] T. Xie, B. Wang, and C.-C. J. Kuo, “Graphhop: An enhanced label propagation method for node classification,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 11, pp. 9287–9301, 2022.
- [87] Q. Huang, H. He, A. Singh, S.-N. Lim, and A. R. Benson, “Combining label propagation and simple models out-performs graph neural networks,” *arXiv preprint arXiv:2010.13993*, 2020.
- [88] A. K. Dayı, O. E. Akgün, S. Gil, M. Yemini, and A. Nedić, “Fast distributed optimization over directed graphs under malicious attacks using trust,” *arXiv preprint arXiv:2407.06541*, 2024.
- [89] M. Zhao, J. Xi, L. Wang, K. Xia, and Y. Zheng, “Completely distributed secure consensus for multiagent systems with a general directed graph under interaction link attacks,” *IEEE Systems Journal*, 2024.
- [90] Y. Sun, Z. Liu, B. Hooi, Y. Yang, R. Fathony, J. Chen, and B. He, “Multi-label node classification with label influence propagation,” in *The Thirteenth International Conference on Learning Representations*.
- [91] Z. Shi, J. Wang, F. Lu, H. Chen, D. Lian, Z. Wang, J. Ye, and F. Wu, “Label deconvolution for node representation learning on large-scale attributed graphs against learning bias,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.
- [92] Y. Dong, S. Wang, Y. Wang, T. Derr, and J. Li, “On structural explanation of bias in graph neural networks,” in *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 316–326, 2022.

-
- [93] Y. Dong, N. Liu, B. Jalaian, and J. Li, “Edits: Modeling and mitigating data bias for graph neural networks,” in *Proceedings of the ACM web conference 2022*, pp. 1259–1269, 2022.
- [94] C. Piro, C. Shields, and B. N. Levine, “Detecting the sybil attack in mobile ad hoc networks,” in *2006 Securecomm and Workshops*, pp. 1–11, IEEE, 2006.
- [95] X. Tang, H. Yao, Y. Sun, Y. Wang, J. Tang, C. Aggarwal, P. Mitra, and S. Wang, “Investigating and mitigating degree-related biases in graph convolutional networks,” in *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, pp. 1435–1444, 2020.
- [96] G. Gluch and R. Urbanke, “Query complexity of adversarial attacks,” in *International Conference on Machine Learning*, pp. 3723–3733, PMLR, 2021.
- [97] D. Lin, J. Wu, Q. Fu, Z. Zheng, and T. Chen, “Riskprop: Account risk rating on ethereum via de-anonymous score and network propagation,” *IEEE Transactions on Dependable and Secure Computing*, 2024.
- [98] Q. Fu, D. Lin, J. Wu, and Z. Zheng, “A general framework for account risk rating on ethereum: toward safer blockchain technology,” *IEEE Transactions on Computational Social Systems*, vol. 11, no. 2, pp. 1865–1875, 2023.
- [99] Y.-x. Xie, L.-x. Ji, L.-s. Li, Z. Guo, and T. Baker, “An adaptive defense mechanism to prevent advanced persistent threats,” *Connection Science*, vol. 33, no. 2, pp. 359–379, 2021.
- [100] X. Zhang and P. Bao, “Adaptive reliable defense graph for multi-channel robust gcN,” *IEEE Transactions on Knowledge and Data Engineering*, 2025.
- [101] R. Andersen, F. Chung, and K. Lang, “Local partitioning for directed graphs using pagerank,” in *International Workshop on Algorithms and Models for the Web-Graph*, pp. 166–178, Springer, 2007.

- [102] M. Charikar, K. Makarychev, and Y. Makarychev, “Directed metrics and directed graph partitioning problems,” in *SODA*, vol. 6, pp. 51–60, 2006.
- [103] I. A. Kader, “Path partition in directed graph—modeling and optimization,” *New Trends in Mathematical Sciences*, vol. 1, no. 1, pp. 74–84, 2013.
- [104] X. Cao, Z. Zhang, J. Jia, and N. Z. Gong, “Flcert: Provably secure federated learning against poisoning attacks,” *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 3691–3705, 2022.
- [105] X. Cao, J. Jia, and N. Z. Gong, “Provably secure federated learning against malicious clients,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, pp. 6885–6893, 2021.
- [106] C. Xie, M. Chen, P.-Y. Chen, and B. Li, “Crfl: Certifiably robust federated learning against backdoor attacks,” in *International Conference on Machine Learning*, pp. 11372–11382, PMLR, 2021.
- [107] B. Wang, J. Jia, X. Cao, and N. Z. Gong, “Certified robustness of graph neural networks against adversarial structural perturbation,” in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pp. 1645–1653, 2021.
- [108] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and S. Y. Philip, “A comprehensive survey on graph neural networks,” *IEEE transactions on neural networks and learning systems*, vol. 32, no. 1, pp. 4–24, 2020.
- [109] S. Wu, F. Sun, W. Zhang, X. Xie, and B. Cui, “Graph neural networks in recommender systems: a survey,” *ACM Computing Surveys*, vol. 55, no. 5, pp. 1–37, 2022.
- [110] J. Zhang, M. Wu, Z. Sun, and C. Zhou, “Learning from crowds using graph neural networks with attention mechanism,” *IEEE Transactions on Big Data*, 2024.

-
- [111] Q. Dong, L. Li, D. Dai, C. Zheng, J. Ma, R. Li, H. Xia, J. Xu, Z. Wu, T. Liu, *et al.*, “A survey on in-context learning,” *arXiv preprint arXiv:2301.00234*, 2022.
 - [112] F. Gilardi, M. Alizadeh, and M. Kubli, “Chatgpt outperforms crowd workers for text-annotation tasks,” *Proceedings of the National Academy of Sciences*, vol. 120, no. 30, p. e2305016120, 2023.
 - [113] F. Demrozi, C. Turetta, F. A. Machot, G. Pravadelli, and P. H. Kindt, “A comprehensive review of automated data annotation techniques in human activity recognition,” *arXiv preprint arXiv:2307.05988*, 2023.
 - [114] G. Wang, M. Du, N. Liu, N. Zou, and X. Hu, “Mitigating algorithmic bias with limited annotations,” in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 241–258, Springer, 2023.
 - [115] Z. Chen, H. Mao, H. Li, W. Jin, H. Wen, X. Wei, S. Wang, D. Yin, W. Fan, H. Liu, *et al.*, “Exploring the potential of large language models (llms) in learning on graphs,” *ACM SIGKDD Explorations Newsletter*, vol. 25, no. 2, pp. 42–61, 2024.
 - [116] N. Pangakis, S. Wolken, and N. Fasching, “Automated annotation with generative ai requires validation,” *arXiv preprint arXiv:2306.00176*, 2023.
 - [117] L. Wang, C. Ma, X. Feng, Z. Zhang, H. Yang, J. Zhang, Z. Chen, J. Tang, X. Chen, Y. Lin, *et al.*, “A survey on large language model based autonomous agents,” *Frontiers of Computer Science*, vol. 18, no. 6, p. 186345, 2024.
 - [118] J. Guo, L. Du, H. Liu, M. Zhou, X. He, and S. Han, “Gpt4graph: Can large language models understand graph structured data? an empirical evaluation and benchmarking,” *arXiv preprint arXiv:2305.15066*, 2023.
 - [119] J. Huang, X. Zhang, Q. Mei, and J. Ma, “Can llms effectively leverage graph structural information: when and why,” *arXiv preprint arXiv:2309.16595*, 2023.

- [120] H. Wang, S. Feng, T. He, Z. Tan, X. Han, and Y. Tsvetkov, “Can language models solve graph problems in natural language?,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [121] R. Ye, C. Zhang, R. Wang, S. Xu, Y. Zhang, *et al.*, “Natural language is all a graph needs,” *arXiv preprint arXiv:2308.07134*, vol. 4, no. 5, p. 7, 2023.
- [122] Y. Bu, Y. Zhu, and K. Zhou, “Crowdsourced homophily ties based graph annotation via large language model,” in *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, IEEE, 2025.
- [123] R. Li, S. Zhang, B. Wan, and X. He, “Bipartite graph network with adaptive message passing for unbiased scene graph generation,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11109–11119, 2021.
- [124] X. Shen, S. Pan, K.-S. Choi, and X. Zhou, “Domain-adaptive message passing graph neural network,” *Neural Networks*, vol. 164, pp. 439–454, 2023.
- [125] A. Iscen, G. Tolias, Y. Avrithis, and O. Chum, “Label propagation for deep semi-supervised learning,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5070–5079, 2019.
- [126] K. Zhou, A. Martin, Q. Pan, and Z. Liu, “Selp: Semi-supervised evidential label propagation algorithm for graph data clustering,” *International Journal of Approximate Reasoning*, vol. 92, pp. 139–154, 2018.
- [127] J. Chen, Y. Wu, X. Xu, Y. Chen, H. Zheng, and Q. Xuan, “Fast gradient attack on network embedding,” *arXiv preprint arXiv:1809.02797*, 2018.
- [128] D. Zügner and S. Günnemann, “Adversarial attacks on graph neural networks via meta learning,” *arXiv preprint arXiv:1902.08412*, 2019.

-
- [129] J. Zhou, Y. Lai, J. Ren, and K. Zhou, “Black-box attacks against signed graph analysis via balance poisoning,” in *2024 International Conference on Computing, Networking and Communications (ICNC)*, pp. 530–535, IEEE, 2024.
 - [130] M. S. Granovetter, “The strength of weak ties,” *American journal of sociology*, vol. 78, no. 6, pp. 1360–1380, 1973.
 - [131] L. Zhang and W. Tu, “Six degrees of separation in online society,” 2009.
 - [132] B. Recht, M. Fazel, and P. A. Parrilo, “Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization,” *SIAM review*, vol. 52, no. 3, pp. 471–501, 2010.
 - [133] M. Taylor, J. Guiver, S. Robertson, and T. Minka, “Softrank: optimizing non-smooth rank metrics,” in *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pp. 77–86, 2008.
 - [134] Z. Xi, R. Pang, S. Ji, and T. Wang, “Graph backdoor,” in *30th USENIX Security Symposium (USENIX Security 21)*, pp. 1523–1540, 2021.
 - [135] Z. Zhang, J. Jia, B. Wang, and N. Z. Gong, “Backdoor attacks to graph neural networks,” in *Proceedings of the 26th ACM Symposium on Access Control Models and Technologies*, pp. 15–26, 2021.
 - [136] J. Xu, M. Xue, and S. Picek, “Explainability-based backdoor attacks against graph neural networks,” in *Proceedings of the 3rd ACM workshop on wireless security and machine learning*, pp. 31–36, 2021.
 - [137] S. Yang, B. G. Doan, P. Montague, O. De Vel, T. Abraham, S. Camtepe, D. C. Ranasinghe, and S. S. Kanhere, “Transferable graph backdoor attack,” in *Proceedings of the 25th international symposium on research in attacks, intrusions and defenses*, pp. 321–332, 2022.

- [138] H. Xu, X. Liu, Y. Wan, and J. Tang, “Towards fair classification against poisoning attacks,” *arXiv preprint arXiv:2210.09503*, 2022.
- [139] P. Elinas, E. V. Bonilla, and L. Tiao, “Variational inference for graph convolutional networks in the absence of graph data and adversarial settings,” *Advances in neural information processing systems*, vol. 33, pp. 18648–18660, 2020.
- [140] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” *arXiv preprint arXiv:1609.02907*, 2016.
- [141] P. Casanova, A. R. P. Lio, and Y. Bengio, “Graph attention networks,” *ICLR. Petar Velickovic Guillem Cucurull Arantxa Casanova Adriana Romero Pietro Liò and Yoshua Bengio*, 2018.
- [142] J. Gasteiger, A. Bojchevski, and S. Günnemann, “Predict then propagate: Graph neural networks meet personalized pagerank,” *arXiv preprint arXiv:1810.05997*, 2018.
- [143] H. Zhu and P. Koniusz, “Simple spectral graph convolution,” in *International conference on learning representations*, 2021.
- [144] J. Chen, K. Gao, G. Li, and K. He, “Nagphormer: A tokenized graph transformer for node classification in large graphs,” *arXiv preprint arXiv:2206.04910*, 2022.
- [145] N. Entezari, S. A. Al-Sayouri, A. Darvishzadeh, and E. E. Papalexakis, “All you need is low (rank) defending against adversarial attacks on graphs,” in *Proceedings of the 13th international conference on web search and data mining*, pp. 169–177, 2020.
- [146] W. Jin, Y. Ma, X. Liu, X. Tang, S. Wang, and J. Tang, “Graph structure learning for robust graph neural networks,” in *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 66–74, 2020.

-
- [147] X. Zhang and M. Zitnik, “Gnn-guard: Defending graph neural networks against adversarial attacks,” *Advances in neural information processing systems*, vol. 33, pp. 9263–9275, 2020.
- [148] X. Liu, W. Jin, Y. Ma, Y. Li, H. Liu, Y. Wang, M. Yan, and J. Tang, “Elastic graph neural networks,” in *International Conference on Machine Learning*, pp. 6837–6849, PMLR, 2021.
- [149] K. Li, Y. Liu, X. Ao, J. Chi, J. Feng, H. Yang, and Q. He, “Reliable representations make a stronger defender: Unsupervised structure refinement for robust gnn,” in *Proceedings of the 28th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 925–935, 2022.
- [150] R. Lei, Z. Wang, Y. Li, B. Ding, and Z. Wei, “Evennet: Ignoring odd-hop neighbors improves robustness of graph neural networks,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 4694–4706, 2022.
- [151] C. Deng, X. Li, Z. Feng, and Z. Zhang, “Garnet: Reduced-rank topology learning for robust and scalable graph neural networks,” in *Learning on Graphs Conference*, pp. 3–1, PMLR, 2022.
- [152] K. Zhao, Q. Kang, Y. Song, R. She, S. Wang, and W. P. Tay, “Adversarial robustness in graph neural networks: A hamiltonian approach,” *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- [153] S. Liu, J. Chen, T. Fu, L. Lin, M. Zitnik, and D. Wu, “Graph adversarial diffusion convolution,” *arXiv preprint arXiv:2406.02059*, 2024.
- [154] E. Rossi, B. Charpentier, F. Di Giovanni, F. Frasca, S. Günnemann, and M. M. Bronstein, “Edge directionality improves learning on heterophilic graphs,” in *Learning on Graphs Conference*, pp. 25–1, PMLR, 2024.

- [155] X. Zhang, Y. He, N. Brugnone, M. Perlmutter, and M. Hirn, “Magnet: A neural network for directed graphs,” *Advances in neural information processing systems*, vol. 34, pp. 27003–27015, 2021.
- [156] C. L. Giles, K. D. Bollacker, and S. Lawrence, “Citeseer: An automatic citation indexing system,” in *Proceedings of the third ACM conference on Digital libraries*, pp. 89–98, 1998.
- [157] J. Ma, Z. Ma, J. Chai, and Q. Mei, “Partition-based active learning for graph neural networks,” *arXiv preprint arXiv:2201.09391*, 2022.
- [158] W. Zhang, Y. Wang, Z. You, M. Cao, P. Huang, J. Shan, Z. Yang, and B. Cui, “Rim: Reliable influence-based active learning on graphs,” *Advances in Neural Information Processing Systems*, vol. 34, pp. 27978–27990, 2021.
- [159] N. Reimers, “Sentence-bert: Sentence embeddings using siamese bert-networks,” *arXiv preprint arXiv:1908.10084*, 2019.
- [160] S. Brin and L. Page, “The anatomy of a large-scale hypertextual web search engine,” *Computer networks and ISDN systems*, vol. 30, no. 1-7, pp. 107–117, 1998.
- [161] S. Lloyd, “Least squares quantization in pcm,” *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.
- [162] Y. Gal and Z. Ghahramani, “Dropout as a bayesian approximation: Representing model uncertainty in deep learning,” *ICML*, 2016.
- [163] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, “An analysis of approximations for maximizing submodular set functions,” *Mathematical programming*, vol. 14, pp. 265–294, 1978.