COMPSCI 402 – Artificial Intelligence

Final Report

The Development and Outlook of AI Techniques in the field of <Graph Anomaly Detection from HCAI Perspective>

<Shouju Wang>
<sw599@duke.edu>

Abstract

Artificial intelligence is rapidly advancing and its impact on human lives is becoming increasingly profound. However, it is not without its complexities, particularly concerning issues of privacy and transparency. (Schmidt, 2020, 1) AI often disregards human-centered aspects, resulting in biased and unfair outcomes.To address these concerns, Human-centered AI (HCAI) (Ahmad 2023, 2) has emerged, focusing on how AI systems can effectively interact with human lives in a society comprising both artificial and human agents. A critical aspect of HCAI is translating qualitative statements into technical requirements for AI systems to ensure trustworthiness. (Bo 1-46)This paper emphasizes the need for Trustworthy AI systems from data acquisition to deployment and governance, with a focus on data preprocessing and anomaly detection. Anomaly detection plays a vital role in enhancing AI trustworthiness by identifying outliers and addressing adversarial inputs. The paper presents a comprehensive classification of current methods for anomaly detection. It also delves into Graph Anomaly Detection techniques, highlighting the role of Graph Convolutional Networks (GCNs) in addressing anomalies in attributed graphs.

1. Introduction

AI, like other significant technological breakthroughs in human history, has had a profound impact on our society. The widespread adoption of electricity during The Second Industrial Revolution is an example of such a development. Prior to its widespread use, production capability, social division of labor, and labor relations were not as advanced as they are today. Furthermore, cultural and commercial communication between different countries was absent compared to today's globalized world. With the onset of The Second Industrial Revolution, our working hours increased, labor relations became increasingly complex, and our culture, politics, economics, and law underwent significant changes. Today, we take electricity for granted as a necessary resource, just like air and water.

When we shift our focus to artificial intelligence, we see that it only has a history dating back to around 1960. (Michael and Andreas)Artificial intelligence, despite its brief existence, has provided substantial economic benefits to humanity and enhanced various aspects of life. Its advancement has greatly accelerated social development and inaugurated a new era of progress. However, people's feelings and perceptions towards AI are not always positive or straightforward, particularly in regard to issues of privacy and transparency in the AI process. (Bergdahl 3)

In fact, much AI-based technology ignores human-centered aspects and has produced biased and unfair outcomes. These include but are not limited to age, ethics, education, society class, gender, language, culture, emotions, personalities, and many others. (John 3)

Presented below are instances of bias and non-inclusive results that have been previously generated by artificial intelligence systems.

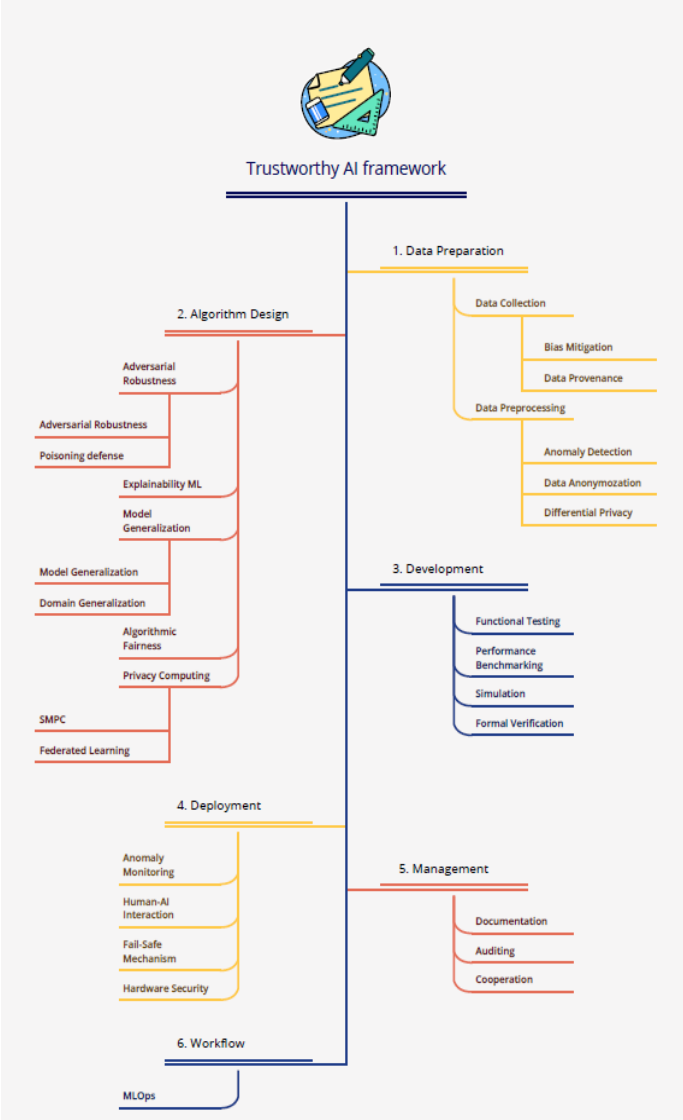
- Carnegie Mellon has reported that due to the underrepresentation of individuals from diverse backgrounds, including people of color and women, in high-paying IT positions, women are significantly less likely to receive jobs with high salary advertisements from online. (Kirsten)
- Within the realm of natural language processing, concerns regarding gender and dialect disparities have been raised, particularly in relation to YouTube's automatic captioning system. (Rachael)
- In the context of facial analysis tasks, machine learning algorithms have demonstrated the capability to make distinctions based on attributes such as race and gender. (Richa)

In order to optimize the application of AI systems to advance our society, Human-centered AI (HCAI) places its focus on the exploration of how contemporary and future AI systems can efficiently interact within a society that comprises both artificial and human agents. A pivotal component of HCAI is the precise transformation of abstract, high-level qualitative declarations into concrete, technical prerequisites for HCAI systems. This explicit recognition underscores the significance of this process.

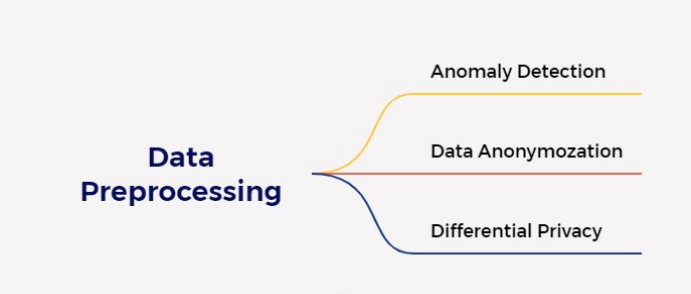
Requirements for Trustworthy AI should be "translated" into procedures and/or constraints on procedures, which should be anchored in the AI system's architecture. ("High-Level Expert Group on Artificial Intelligence. Ethics guidelines for trustworthy AI")

To tackle the issues highlighted, such as subtle attacks, bias against marginalized communities, and the absence of safeguards for user privacy, it becomes imperative to establish a Comprehensive AI System that encompasses every stage from data acquisition, model development, system creation and rollout, all the way to persistent monitoring and governance.

In the field of trustworthy AI, a mature framework has been developed to improve AI trustworthiness. (Bo 8)



In all the fields mentioned in the framework, Prior to inputting data into an AI process, data preprocessing plays a crucial role in eliminating inconsistent data contamination that could potentially disrupt the model's performance and in safeguarding sensitive information that may jeopardize user privacy.



Anomaly detection, often referred to as outlier detection, has been a prominent subject of research in the field of machine learning. This is primarily because machine learning models are highly sensitive to outliers, making data cleaning through anomaly detection a valuable method to enhance their performance.

Recent studies have illustrated the utility of anomaly detection in bolstering AI trustworthiness. Notably, fraudulent data can pose significant challenges to the robustness of systems, particularly in domains like banking and insurance. To combat this issue, several approaches have been introduced, with a focus on employing anomaly detection techniques.

Moreover, it is imperative to recognize and address adversarial inputs as critical safeguards against evasion and data contamination attacks. It's worth noting that the efficacy of detection in high-dimensional data, such as images, still presents limitations. The practice of thwarting adversarial attacks through data cleansing is frequently denoted as data sanitization.

In the subsequent section of this paper, Part 2, I will present a comprehensive classification of the current methods. Within each category, I will meticulously examine two exemplary papers to offer a more profound analysis. Finally, I will conclude by summarizing the key attributes of the prevailing methods within each category.

2. A brief survey of existing methods proposed in anomaly detection.

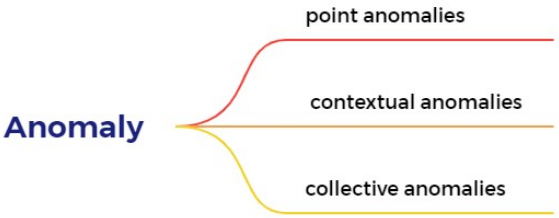
2.1 What is an anomaly

Within the realm of data mining and statistics, anomalies are commonly denoted as abnormalities, deviants, or outliers. These terms serve the purpose of characterizing data points that notably deviate from the prevailing majority within a dataset. Their significance lies in their pivotal role in the detection of unusual patterns or potential issues in datasets. Anomalies, in data, frequently unveil valuable insights and carry indispensable information. The process of anomaly detection assumes a critical role in decision-making systems spanning diverse

domains. The identification and comprehension of such anomalies facilitate the revelation of latent patterns, potential concerns, or opportunities concealed within the data. This procedure contributes to the enhancement of decision-making by fostering a more informed, data-driven approach, ultimately yielding improved outcomes across a broad spectrum of applications. (Srikanth)

2.2 Taxonomy of anomaly

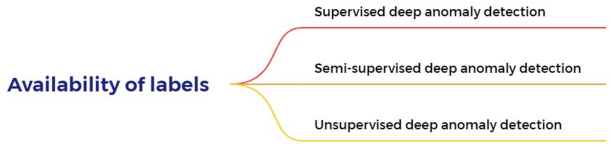
Anomalies exhibit diversity and can be classified into three overarching categories: point anomalies, contextual anomalies, and collective anomalies. (Raghavendra 9)



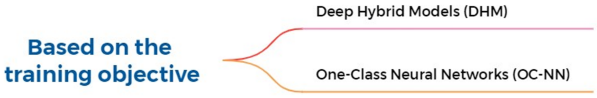
- The predominant body of literary work predominantly addresses point anomalies. Point anomalies frequently signify irregularities or deviations that occur randomly and may lack a specific interpretation.
- A contextual anomaly, alternatively termed a conditional anomaly, refers to a data instance that could be categorized as anomalous within a particular context. The identification of contextual anomalies involves the consideration of both contextual and behavioral features. Contextual features, typically encompassing time and space, are taken into account. Meanwhile, behavioral features may encompass patterns such as spending habits, the frequency of system log events, or any feature that characterizes normal behavior.
- Anomalies that manifest as unusual patterns when data points are considered collectively are referred to as collective or group anomalies. In such cases, each individual data point may appear normal when examined in isolation, but when observed as part of a group, they exhibit distinctive and uncommon characteristics.

2.3 Taxonomy of anomaly detection technique

1. **Availability of labels**



2. **From the training objective**



2.3 state of art subtopic : Graph Anomaly Detection

2.3.1 overview of graph anomaly detection

Anomaly detection is a crucial facet of data processing, aiming to discern uncommon patterns that stand out from the mainstream in a dataset (Tahereh). Conventional approaches in anomaly detection often involve representing real-world objects as feature vectors. For instance, social media posts might be encoded as bag-of-words (Sun), or web page images could be depicted using color histograms. These methodologies then identify outlier data points within this vector space (Zheng).

Although these methods have demonstrated their effectiveness in detecting abnormal data points within structured data formats, they often neglect the intricate interconnections between objects. (Leman). In reality, many objects are intricately linked, providing valuable additional insights for identifying anomalies. These complex structural relationships are frequently represented as graphs, where nodes or vertices represent actual entities, and edges signify the connections between these entities. The application of graph-based anomaly detection spans a wide array of fields, including social activities, e-commerce, and numerous others.

2.3.2 definition of different types of graph

According to the attributes of edges and nodes, we can categorize graphs into three different types. (Ma)

- Plain Graph definition: A static, undirected graph G is formally denoted as G = {V, E}. Here, V signifies the set of nodes, represented as V = {vi}n, and E represents the set of edges, which is denoted as E = {ei,j}. In this context, n corresponds to the total number of nodes, and ei,j = (vi, vj) signifies the presence of an edge connecting nodes vi and vj. The structural arrangement of the graph is encoded in the adjacency matrix A =

[ai,j]n×n, where ai,j assumes a value of 1 to signify a connection between nodes vi and vj and a value of 0 to indicate the absence of such a connection.

- Attributed Graph definition: A static attributed graph, symbolized as G = {V, E, X},comprises a collection of nodes (V), a network of edges (E), and a collection of attributes (X). Within this context, the graph's structure adheres to the definition outlined in Definition 1. The attribute matrix X, denoted as [xi]n×k, encompasses attribute vectors associated with the nodes. Each xi corresponds to the attribute vector linked with the node vi, while k denotes the dimension of these vectors. It's worth emphasizing that, going forward, the terms "attribute" and "feature" are employed interchangeably.
- Dynamic Graph definition:A dynamic graph, referred to as G(t), is characterized by a collection of elements {V(t), E(t), Xv(t), Xe(t)}. These elements undergo temporal changes, with V(t) signifying the nodes in the graph at a particular time step t, E(t) representing the associated edges at that specific time point, and Xv(t) and Xe(t) designating the node attribute matrix and edge attribute matrix, respectively, should they be present within the graph at time step t.

2.3.3 Different techniques on anomaly node detection

We can categorize different detection techniques according to the objectives. As mentioned before, a graph structure has node information and edge information. And a graph can also be categorized into three different types. Focusing on each type, we have anomaly detection on plain graph nodes, attributive graph nodes, and dynamic graph nodes. We also have techniques on the edges of each of three different types. Although some of the techniques can be utilized on each type, there are still some important differences when implementing concrete methods.

The table summarized techniques used for graph node anomaly detection.

Graph Type	Methoad	class	Evaluation
Static Plain Graph	DCI	NR	Anomaly Score
	NAC	RL	Anomaly Prediction
Static Attributed Graph	DOMINANT	GCN	Anomaly Score
	ALARM	GCN	Anomaly Score
	SpecAE	GCN	Density Estimation
	Fdgars	GCN	Anomaly Prediction
	GraphRfi	GCN	Anomaly Prediction
	ResGCN	GCN	Anomaly Score
	GraphUCB	RL	
	AnomalyDAE	GAT	Reconstruction Loss
	SemiGNN	GAT	Anomaly Prediction
	AEGIS	GAN	Anomaly Score
	REMAD	NR	Residual Analysis
	CARE-GNN	NR	Anomaly Prediction
	SEANO	NR	Anomaly Score
	OCGNN	NR	Location in Embedding Space
	GAL	NR	Anomaly Prediction
	CoLA	NR	Anomaly Score
	COMMANDER	NR	Anomaly Score
	FRAUDRE	NR	Anomaly Prediction
	Meta-GDN	NR	Anomaly Score
Dynamic Plain Graph	NetWalk	DNN	Anomaly Score
Dynamic Attributed Graph	MTHL	Non-DP	Anomaly Score
	OCAN	GAN	Anomaly Score

In the next section, I will deeply introduce Graph convolutional neural networks (GNC) to solve graph node anomaly detection problems.

2.3.4 GNC Based Techniques on anomaly node detection

Graph convolutional neural networks (GNC) is a widely used deep learning module to do tasks with graph or network data structure. (Thomas) They have gained significant attention recently, particularly in complex relationships related tasks, such as social network, recommendation system, biological networks and so on.

Typically, the process or the framework of GNC involve following key steps.

1. Data preparation
   - Graph Representation: The first step is to represent your data as a graph. This includes defining nodes (entities) and edges (relationships) between them. You also need to associate features with each node.
   - Adjacency Matrix: Create an adjacency matrix that describes the relationships between nodes. This matrix encodes which nodes are connected to each other.

2. Model Architecture:
   - Input Layer: The model starts with an input layer that takes the graph structure (adjacency matrix) and node features as input.
   - Graph Convolutional Layers: GCNs typically consist of multiple graph convolutional layers. Each layer aggregates information from neighboring nodes using the adjacency matrix and node features.
   - Activation Functions: Non-linear activation functions, like ReLU, are often applied after each convolutional layer.
   - Output Layer: The final layer is responsible for producing the desired output, depending on the task. This can be a classification layer for node classification tasks or a regression layer for other tasks.

3. Message Passing
   During the graph convolutional layers, a key operation is the message passing process. This involves propagating information from neighboring nodes to a given node based on the graph structure and the weights learned during training.

4. Training:
   - Cost Function: Establish a cost function designed to measure the disparity between the model's predictions and the actual ground truth. The selection of the cost function is contingent upon the nature of the task at hand, whether it be node classification or link prediction..
   - Optimization: Use an optimization algorithm (e.g., stochastic gradient descent) to update the model's weights and minimize the loss function. Training involves forward and backward passes to adjust the model's parameters.

5. Evaluation
   Following the training phase, assess the model's effectiveness using a validation set or, in the context of node classification, by analyzing labeled nodes that were not part of the training data. Commonly employed evaluation metrics encompass accuracy, F1 score, or mean squared error, chosen based on the specific nature of the task at hand.

6. Inference
   Once the model is trained and evaluated, you can use it for inference on new data or for making predictions on unseen parts of the graph.

7. Regularization
   It's often necessary to apply regularization techniques to prevent overfitting.

In the realm of detecting anomalies within an attributed graph, the DOMINANT approach stands out as a prominent method. This approach is structured into three integral components: the graph convolutional encoder, the structural reconstruction decoder, and the attribute reconstruction decoder. These elements work in unison within a neural network framework, which is meticulously trained to minimize the ensuing loss function:

$$L_{\{DOMINANT\}} = (1 - \alpha)R_S + \alpha R_A$$

In the context mentioned, α represents the coefficient, while R_S and R_A serve to assess reconstruction errors concerning the graph structure and node attributes.
Upon the completion of training, an anomaly score is subsequently distributed to each node based on its role in influencing the overall reconstruction flaw. This error is determined by the following calculation:

$$S_i = (1 - \alpha)||a_i - \hat{a}_i||_2 + \alpha||x_i - \hat{x}_i||_2,$$

In this context, node i is represented by structure vector a_i and attribute vector x_i, while their respective reconstructed vectors are denoted as $\widehat{a_i}$ and $\widehat{x_i}$. Following the computation of anomaly scores, the nodes are arranged in a descending order, and anomalies are determined by selecting the top-k nodes.

3. Discussion

3.1 Challenges in graph detection

Despite the extensive analysis and research in the field of anomaly detection across various domains and applications over recent years, the realm of graph anomaly detection presents a distinct set of challenges. These challenges stem from the intricate relationships among real-world entities, encompassing algorithmic complexities, unique features of graph data, and the relative immaturity of deep learning techniques tailored for graph data mining. Moreover, the predominant focus of current research primarily centers on detecting anomalous nodes, leaving the domains of anomalous edges and sub-graphs comparatively underexplored. This prevailing challenge has paved the way for an exciting avenue of research in the domain of Edge, Sub-graph, and Graph anomaly detection.

And we also live in a fast changing world, in which relationships and connections change quickly. Correspondingly, for the graph relation structure, the node and link are in a continuing dynamic state, which demand us to dynamically detect anomalies

What's more, it is a big data era, and the amount and dimensions of data is also a challenge we have to solve. Take social networks at an example,

3.2 Outlook and future research opportunities towards graph detection. (Ma 17)

1. **Less anomaly detection research in edge and sub-graph detection.**

   When looking at graph structures in real world, anomalies manifest as unconventional relationships among objects formed by atypical groups, or uncommon graphs, which are respectively termed anomalous edges, sub-graphs, and graphs. And the true challenge lies in the concrete application domain. When it comes to detecting anomalous edges, sub-graphs, and graphs, the proposed methods should possess the capability to harness the wealth of information embedded within graphs. This information serves to identify distinctive clues and characteristics that differentiate normal objects from outliers in certain applications. Traditionally,this involves the retrieval of characteristics at the edge, sub-graph, and graph levels, creating models to identify patterns based on these characteristics, and then evaluating data points that deviate from the norm.

2. **Dynamic Graphs Detection**

   The continually changing structure and attribute information in these contexts create inherent complexities for anomaly detection. These complexities lead to two main focal points in this endeavor. The initial point of emphasis is the need to incorporate spatial and temporal information within each snapshot of a dynamic graph, spanning multiple time intervals. The secondary point of interest revolves around unraveling the evolving features of nodes, edges, sub-graphs, and entire graphs, as well as their interplay with node and edge attributes over time. As these complexities are effectively tackled with well-established solutions, the potential for anomaly detection techniques to deliver enhanced, resilient outcomes becomes increasingly promising.

3. **Anomaly Detection in Huge Graphs**

   Addressing the scalability of anomaly detection methods for high-dimensional and large-scale data remains an ongoing and substantial challenge. Take, for example, the expansive networks found on platforms like Facebook and Twitter, boasting billions of users and friendship links, resulting in a substantial volume of data. The size of this data, both in terms of graph dimensions and node attributes, is exceptionally large. However, most existing approaches lack the capacity to effectively identify anomalies within such extensive datasets.

   This limitation stems from their reliance on transductive models, which necessitate processing the entire graph as input for subsequent analysis. As the network's scale increases, the computational time and memory requirements experience a dramatic surge, making existing techniques unviable for deployment in large-scale networks.

4. **Challenges in Detecting Adversarial Anomalies**

   The widespread use of online platforms makes them vulnerable to abuse by fraudsters, attackers and other malicious actors intent on engaging in harmful activities. While numerous anomaly detection systems have been deployed to safeguard legitimate entities, anomalies exhibit a remarkable capacity to shroud themselves, eluding identification—a phenomenon commonly referred to as 'cloaked anomalies.' These entities frequently assume the guise of commonplace objects. In the absence of detection methods capable of promptly and adeptly adjusting to this scenario, in other words, if they fail to swiftly and efficiently accommodate the evolving evasion strategies employed by potential attackers, these anomalies may go unnoticed and consequently pose a latent threat.

5. **Multi-faceted Anomaly Detection**

   In the realm of anomaly detection, we often confront a plethora of graph structures and intricate relationships. Simultaneously, there's a need to address additional objectives like community detection, node

classification, and relation prediction. The convergence of anomaly detection with these diverse tasks opens the door to the realm of multi-task learning, presenting a unique opportunity. This paradigm enables the concurrent management of a variety of tasks while facilitating the exchange of knowledge and insights among them, yielding advantages for all parties involved.

6. Enhancing the Interpretability of Graph Anomaly Detection

The significance of interpretability in the realm of anomaly detection techniques cannot be overstated. When these techniques are deployed in practical domains such as financial and insurance systems, the ability to furnish transparent and legally defensible evidence to substantiate anomaly detection outcomes becomes paramount. Regrettably, a notable constraint in many existing approaches is their limited capacity to offer such elucidating evidence.

Typically, the prevalent methods rely on top-k rankings and simplistic anomaly scoring functions as the primary metrics for identifying anomalies. While these metrics prove effective in classifying objects as anomalies or non-anomalies, their deficiency in delivering comprehensive explanations is evident. Moreover, considering the ongoing criticism of the low interpretability associated with deep learning techniques, it is imperative that forthcoming research in the field of graph anomaly detection using deep learning places a substantially stronger emphasis on enhancing interpretability."

7. Graph Anomaly Identification Strategies

In the field of unsupervised graph anomaly detection techniques, anomalies are typically recognized by a variety of methods, including residual analysis, reconstruction error, distance-based statistics, density-based statistics, graph scanning statistics, and single-class classification are techniques employed for anomaly detection. These methods share a common foundational principle: anomalies display data patterns distinct from those of regular objects. Consequently, anomalies often manifest as 1) generating more pronounced residual errors or inducing complexities during the reconstruction process, and 2) inhabiting sparsely populated areas distant from the majority of classes within the anomaly-aware feature space.

Notably, in the context of graph neural networks (GNNs) in anomaly detection, limited attention is currently being paid to innovative loss functions specifically designed for them.

While these strategies are highly effective in capturing the unique data characteristics of anomalies, they also have inherent limitations. Specifically, residual analysis, single-class classification, and reconstruction error strategies are particularly sensitive to noisy training data. The presence of noisy nodes, edges, or subgraphs tends to result in significant residuals, locations far from the origin or the center of the high-dimensional sphere, and high reconstruction errors. Meanwhile, the distance-based and density-based strategies are effective only when the anomalies and non-anomalies have a clear demarcation in the low-dimensional space. When the boundary between anomalies and non anomalies is less obvious, the effectiveness of detection is significantly weakened.

Therefore, strong future efforts are urgently needed to overcome these limitations and explore innovative strategies to recognize anomalies.

8. Systematic Benchmarking

Conducting systematic benchmarking stands as a pivotal component in the evaluation of graph anomaly detection techniques. Typically, a benchmarking framework encompasses benchmark datasets, baseline methods, evaluation metrics, and supplementary analytical tools. The choice of evaluation dataset and metrics becomes of utmost importance when gauging a technique's performance in comparison to other baselines, as it significantly influences each model's effectiveness.

However, the scarcity of publicly accessible datasets and baseline methods presents substantial obstacles to conducting thorough evaluations. While one of the primary objectives of our survey is to provide extensive resources for this purpose, including open-source implementations, datasets, and evaluation metrics, this endeavor should be regarded as a foundational step towards future systematic benchmarking endeavors. We wholeheartedly encourage heightened engagement from the anomaly detection community to address this crucial need.

Undoubtedly, a steadfast commitment to crafting enhanced benchmarking frameworks would prove instrumental in illuminating the strengths and weaknesses of various detection techniques. Such an effort would, in turn, contribute to maintaining an impartial and precise record of progress within this domain.

9. Comprehensive Anomaly Detection Framework

Anomalies in a graph can take different shapes, ranging from anomalous nodes, edges, or subgraphs within a single graph to an entire anomaly graph in a database.

These diverse anomalies usually exist simultaneously in real-world datasets. For example, in the realm of online social networking, individual fraudsters, strange relationships, or even entire fraudulent groups may be present at the same time.

Furthermore, the definition of anomalies can vary, distinguishing between community outliers and anomalous communities or differentiating attribute-based from structural anomalies. When it comes to implementing detection techniques in practical applications, the goal is to identify all types of anomalies while being mindful of resource and time efficiency.

While a direct approach involves integrating separate detection techniques for anomalous nodes, edges, and sub-graphs, this method becomes computationally unwieldy and impractical for extensive networks, such as those on platforms like Facebook and Twitter. This challenge arises from the need to load and process the same graph data repeatedly using different techniques.

To tackle this issue, the concept of unified frameworks emerges as a promising solution. These frameworks are designed to simultaneously detect various anomaly types, streamlining the process by capturing all necessary information for different detection techniques at once. Nonetheless, implementing such frameworks in the realm of deep learning entails significant effort and innovation, involving the design of neural network layers and learning strategies to fulfill this multifaceted requirement.

Works Cited

Ahmad, Khlood. “Requirements engineering framework for human-centered artificial intelligence software systems,.” *Applied Soft Computing,*, vol. Volume 143,, 2023, https://doi.org/10.1016/j.asoc.2023.110455.

Bergdahl, Jenna. “Self-determination and attitudes toward artificial intelligence: Cross-national and longitudinal perspectives,.” *Telematics and Informatics,*, vol. Volume 82,, 2023, https://doi.org/10.1016/j.tele.2023.102013.

Bo, Li. “Trustworthy AI: From principles to practices.” *ACM Computing Surveys*, vol. 55.9, 2023.

“High-Level Expert Group on Artificial Intelligence. Ethics guidelines for trustworthy AI.” 2019, https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai.

John, Grundy. “Impact of End User Human Aspects on Software Engineering.” Accessed 2021.

Kirsten, Lloyd. “Bias Amplification in Artificial Intelligence Systems.” 2018.

Leman, Akoglu. “Graph-based Anomaly Detection and Description: A Survey.” *Data Mining and Knowledge Discovery.*, vol. 29, 2014.

Ma, Xiaoxiao. “A Comprehensive Survey on Graph Anomaly Detection with Deep Learning.” *IEEE Transactions on Knowledge and Data Engineering.*, 2021.

Michael, Haenlein, and Kaplan Andreas. “A Brief History of Artificial Intelligence: On the Past, Present, and Future of Artificial Intelligence.” *California Management Review.* Accessed 2019.

Rachael, Tatman. “Gender and Dialect Bias in YouTube's Automatic Captions.” vol. 53-59, 2017.

Raghavendra, Chalapathy. “Deep Learning for Anomaly Detection: A Survey.” Accessed 2019.

Richa, Singh. “Anatomizing Bias in Facial Analysis.” *AAAI Conference on Artificial Intelligence*, 2021.

Schmidt, Albrecht. “Interactive Human Centered Artificial Intelligence: A Definition and Research Challenges.” *AVI '20: Proceedings of the International Conference on Advanced Visual Interfaces*, 2020, https://dl.acm.org/doi/10.1145/3399715.3400873.

Srikanth, Thudumu. “A comprehensive survey of anomaly detection techniques for high dimensional big data.” *Journal of Big Data.*, vol. 7, p. 2020.

Sun, Xiao. “Detecting Users’ Anomalous Emotion Using Social Media for Business Intelligence.” *Journal of Computational Science*, vol. 25, 2017.

Tahereh, Pourhabibi. “Fraud detection: A systematic literature review of graph-based anomaly detection approaches.” *Decision Support Systems*, vol. 133, 2020.

Thomas, Kipf. “Semi-Supervised Classification with Graph Convolutional Networks.” 2016.

Zheng, Wang. “Towards a Hierarchical Bayesian Model of Multi-View Anomaly Detection.” vol. 2392-2398, 2020.