Enhancing Graph Clustering in Dynamic Networks with Distributed Online Life-Long Learning

Hariprasauth Ramamoorthy[a], Rajkumar Vaidyanathan[b] and Suresh Sundaram[c]
Indian Institute of Science, Bengaluru, Karnataka, India
{hariprasauth, vssuresh}@iisc.ac.in, rajvaidyanathan@gmail.com

Keywords: Graph-Based Clustering, Distributed Online Life-Long Learning (DOL3), Social Network Analysis, Trust and Reputation, Multi-Agent System, Trust, Dynamic Network.

Abstract: Trust and reputation assessment are critical in dynamic environments like recommendation systems, biological network and social networks. Malicious agents tend to collude to manipulate the reputation for selfish reasons. However, traditional methods struggle to adapt to the evolving relationships and interactions within these networks. This paper introduces a novel approach that integrates Distributed Online Life-Long Learning (DOL3) with graph clustering to address the challenge of collusion. By enabling agents to continuously learn and update their clustering models, our approach enhances the system's ability to detect malicious agents, maintain trust, and ensure the integrity of reputation scores. We present a detailed mathematical formulation of our algorithm, incorporating local clustering models, distributed consensus, and model adaptation. Experimental results on the Cora dataset demonstrate the superior performance of our approach compared to existing methods, particularly in terms of accuracy (by 11.8%) and adaptability to dynamic and complex network scenarios. The accuracy is measured using Normalized Mutual Information (NMI), a robust metric for comparing predicted and actual cluster assignments. Our findings highlight the effectiveness of DOL3-enhanced graph clustering in addressing the challenges of trust and reputation assessment in dynamic environments.

1 INTRODUCTION

Graph clustering is a fundamental task in network analysis, with applications in various domains such as social networks, biological networks, and recommendation systems. The goal of graph clustering is to partition a graph into groups of nodes (clusters) that are densely connected within themselves but sparsely connected to other clusters. Traditional graph clustering algorithms often assume static networks in which the relationships between nodes remain constant over time. However, in many real-world scenarios, networks are dynamic and evolve due to changes in node attributes, edge weights, or the addition/removal of nodes. These dynamic changes can significantly impact the accuracy and relevance of the clustering results (Sievers, 2020).

To address the limitations of traditional graph clustering algorithms, this paper proposes a novel approach that integrates Distributed Online Life-Long Learning (DOL3) with graph clustering. DOL3 enables agents to continuously learn and adapt their models in a distributed manner, making them well suited for dynamic environments. The contributions of this paper are as follows:

- DOL3-Based Graph Clustering Framework: We introduce a framework that integrates the novel DOL3 with graph clustering, allowing agents to adapt their clustering models to changing network dynamics.
- Mathematical Formulation: We provide a detailed mathematical formulation of the proposed algorithm, including the local clustering models, distributed consensus mechanism, and the model adaptation process.
- Experimental Evaluation: We conducted experiments on synthetic and real-world datasets to evaluate the performance of our approach and compare it with existing methods.

In the context of multi-agent systems, trust refers to the belief that an agent will act in a way that is beneficial to another agent, even in the absence of direct monitoring or enforcement. Reputation is a collective

[a] https://orcid.org/0009-0004-4922-0319
[b] https://orcid.org/0009-0002-2577-0813
[c] https://orcid.org/0000-0001-6275-0921

Ramamoorthy, H., Vaidyanathan, R. and Sundaram, S.
Enhancing Graph Clustering in Dynamic Networks with Distributed Online Life-Long Learning.
DOI: 10.5220/0013167100003890

In *Proceedings of the 17th International Conference on Agents and Artificial Intelligence (ICAART 2025) - Volume 1*, pages 80-90
ISBN: 978-989-758-737-5; ISSN: 2184-433X

assessment of an agent's trustworthiness based on the opinions and experiences of other agents. Some of the examples of dynamic network include social networks, biological networks, e-commerce recommendation systems, etc. (Govindaraj et al., 2021). It is often used as a proxy for trust, as it reflects the consensus view of an agent's behavior. Similar online learning approaches have also been used in the area of robotics (Gupta and Sundaram, 2023). This paper proposes a novel approach that integrates Distributed Online Life-Long Learning (DOL3) with graph clustering to address the challenges of trust and reputation assessment in dynamic networks. DOL3 empowers agents to continuously learn and update their clustering models, enhancing the system's adaptability and robustness. By combining DOL3 with graph clustering, we aim to improve the accuracy and efficiency of identifying malicious agents and maintaining the integrity of reputation scores. Our proposed algorithm offers several key advantages:

- Adaptability: DOL3 enables agents to continuously learn and adapt their clustering models to changing network dynamics, making the system more resilient to evolving conditions.
- Accuracy: By combining graph clustering with DOL3, we achieve improved accuracy in identifying malicious agents and assessing trustworthiness.
- Efficiency: Our approach is computationally efficient, making it suitable for real-world applications with large-scale networks.

The remainder of this paper is organized as follows: Section 2 provides a background on trust and reputation systems, graph clustering, and DOL3. Section 3 presents the proposed DOL3-enhanced graph clustering algorithm. Section 4 describes the experimental setup and evaluation methodology. Section 5 presents the experimental results, comparing our approach to existing methods. Finally, Section 6 concludes the paper with a summary of our findings and potential future directions.

2 RELATED WORK

There are models and studies in the area of Graph Clustering, Colluding Agents and the assessment of Trust and Reputation. This section explains the key related works in these areas and their underlying principles that drove the motivation behind this paper. Graph clustering, a fundamental task in data mining and machine learning, involves partitioning a graph into subsets of nodes, such that nodes within the same subset are more similar to each other than those in different subsets. This section also outlines the integration of graph clustering with the trust assessment.

2.1 Graph Clustering and Learning

Graph clustering, also known as community detection, aims to identify groups of nodes (agents) that are densely connected within themselves but sparsely connected to other groups. Various algorithms have been proposed for graph clustering, including:

- Modularity-based methods: These methods optimize a modularity score to identify communities (Ghosh et al., 2019). Examples include Louvain Modularity Optimization (Seifikar et al., 2020) and Girvan-Newman algorithm (Despalatović et al., 2014).
- Spectral clustering: This method uses the eigenvectors of the graph Laplacian matrix to embed nodes in a lower-dimensional space and then apply clustering algorithms (Liu and Han, 2018).
- Hierarchical clustering: This approach creates a hierarchy of clusters by iteratively merging or splitting existing clusters (Bonald, 2018).
- Deep learning-based methods: Recent advances in deep learning have led to the development of graph neural networks (GNNs) for graph clustering tasks. GNNs can learn representations of nodes and edges that capture the underlying structure of the graph (Wu et al., 2020).
- K-means clustering: While not specifically designed for graph clustering, K-means can be applied to the node embeddings obtained from graph clustering algorithms to further refine the clusters (Galluccio et al., 2012).

Network representation learning, also known as network embedding, aims to learn low-dimensional representations for nodes in a graph that capture the underlying structure and relationships between nodes. These representations can be used for various tasks, including node classification, link prediction, and community detection (Wang et al., 2022).

2.2 Trust and Reputation Assessment

Trust and reputation are fundamental concepts in multi-agent systems, particularly in dynamic environments where agents interact, collaborate, and make decisions based on their perceptions of each other's trustworthiness. Trust and reputation are essential for effective collaboration and cooperation among agents. (Drawel et al., 2022) explains the art of propagating trust in a scalable manner among the agents. This lays the foundation on how the reputation could be built over time through information exchange. (Barbosa et al., 2023) significantly describes the importance of trust sharing in the negotiation phase of agents to mitigate the malicious agents in the network. (Khalid et al., 2021) introduces algorithms to effectively validate the trust credibility of agents as seen through the observers in the social network.

2.3 Colluding Agents in Multi-Agent Systems

Colluding agents in multi-agent systems pose significant challenges to trust and reputation systems. The colluding agents form groups and communities to manipulate the trust scores of each other thereby corrupting the overall reputation scores in the social network. Some of the possible ways to detect such colluding agents include identifying abnormal patterns in agent behavior (Johnson and Sokol, 2020), analyzing the structure of agent relationships to detect cliques or communities that might be engaged in collusion (Mohanty, 2020) and employing machine learning algorithms to learn patterns of collusion and identify potential culprits (Rodríguez et al., 2022). Detecting collusion in dynamic environments can be challenging, as the patterns of collusion may evolve over time.

2.4 Integration of Graph Clustering and Trust

Several studies have explored the integration of graph clustering with trust and reputation systems. (Rossetti and Cazabet, 2018) reviews the idea of modeling the evolution of dynamic network as nodes and edges representing the agents and their interactions. These studies often focus on static environments or do not explicitly address the challenges of dynamic networks that evolves over time. Our proposed approach, which integrates DOL3 with graph clustering, offers a novel solution to these challenges. The approach ensures that the influence of the outdated information due to the dynamic nature of the environment is addressed. The graph clustering ensures that the model can identify the colluding agents to penalise them with lower trust scores.

Life-long learning is a machine learning paradigm where a model can continuously learn and adapt to new data and tasks over time. This is particularly important in dynamic environments where the data distribution may change.

3 PROPOSED FRAMEWORK

Our proposed framework for trust and reputation assessment in dynamic networks integrates Distributed Online Life-Long Learning (DOL3) with graph clustering. The framework consists of the following components:

3.1 Agent Representation

Each agent is represented by a feature vector that captures its characteristics and attributes. The feature vector $\vec{x}_i$ of the i^{th} agent can include information such as the agent's role, past behavior, reputation, and other relevant features. The feature vector can be used to carry the context of the interaction as well as the belief of trust on other agents. Some of the characteristics considered in this framework include:

- Behavioral data: Past actions, interactions, and decisions.
- Reputation scores: Ratings or assessments from other agents.
- Social connections: Relationships with other agents in the network.
- Domain-specific features: Features relevant to the particular application or domain.

By representing these characteristics in feature vectors, we can apply machine learning algorithms and data mining techniques to analyze agent behavior, identify patterns, and make predictions about their trustworthiness.

Let A be the set of all agents in an environment. a_i represents the individual agents. F is the set of features characterising an agent. f_{ij} be the specific feature of the i^{th} agent's j^{th} feature. Each agent can be represented by a feature vector $\vec{x}_i$:

$$\vec{x}_i = [f_1(a_i), f_2(a_i), .., f_n(a_i)] \tag{1}$$

where:

- n is the total number of features
- $f_j(a_i)$ is the value of feature f_j for agent a_i

3.2 Graph Construction

A graph $G = (V, E)$ is constructed to represent the relationships between agents. Vertex V in the graph corresponds to set of agents which are represented by nodes , and edge E represents connection or interaction between agents. Edge weight w can be assigned based on factors such as similarity, trust, or frequency of interactions.

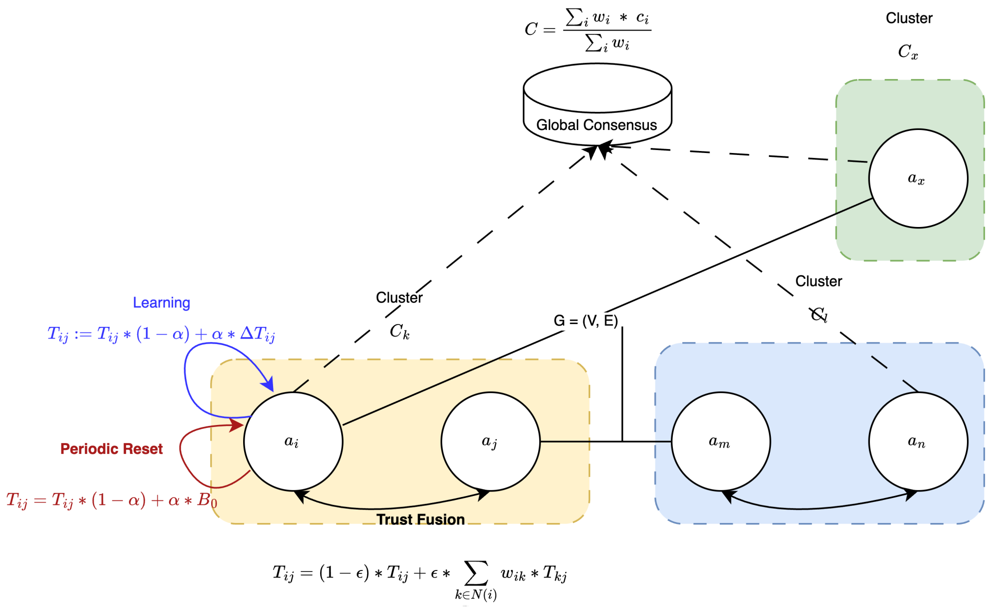

Figure 1: Graph clustering integrated with DOL3 framework.

We used a directional graph to represent the agents' interactions among each other. This helps in identifying the influence and community formation among the agents. To represent the dynamic nature of the environment, the overall graph G was simulated to be random representing certain number of nodes and edges. The edge weights also offer flexibility in defining the algorithm by offering various characteristics like:

- Frequency of Interactions: The number of times agents have interacted.
- Reciprocity: Whether the relationship is mutual or one-sided.
- Trust Levels: The level of trust between agents.
- Similarity: The degree of similarity between agents' features.

Edge weight function $w(a_1, a_2)$ between two agents / nodes a_1 and a_2 can be defined using different functions like Similarity-based for n features:

$$w(a_1, a_2) = sim(a_1, a_2) \tag{2}$$

$$d(a_1, a_2) = \sqrt{\sum_{i=1}^{n} (x_1[i] - x_2[i])^2} \tag{3}$$

$$sim(a_1, a_2) = \frac{1}{(1 + d(a_1, a_2))} \tag{4}$$

where $sim(u, v)$ represents the similarity between the two agents u and v. x_1 and x_2 represent the features of a_1 and a_2 respectively. In this paper, all features are represented numerically and hence the similarity function with respect to euclidean function is considered. There are other similarity functions as defined in (Ontañón, 2020), but the Euclidean distance based function is more appropriate for this specific use case.

Cluster modeling In the initialization phase, the agents are assigned to their own clusters $c_i = \{a_i\}$ for $i = 1, 2, 3..n$. where n is the number of agents. During the merging phase, the distance is calculated using (3) between each pair of clusters. In Hierarchical Clustering, even if a single point / feature is close to each other between two clusters, its called **Single-linkage** and **Complete-linkage** is when all the points are close to each other. The single-linkage, complete-linkage and average-linkage formulations are represented below:

$$d(C_i, C_j) = \min d(a_i, a_j) | a_i \in C_i, a_j \in C_j$$

$$d(C_i, C_j) = \max d(a_i, a_j) | a_i \in C_i, a_j \in C_j$$

$$d(C_i, C_j) = \frac{1}{(|C_i| * |C_j|)} * \sum d(a_i, a_j) | a_i \in C_i, a_j \in C_j \tag{5}$$

Based on the type of linkage, the clusters C_i and C_j are merged into a new cluster C_k.

3.3 Distributed Online Life-Long Learning (DOL3)

DOL3 enables agents to continuously learn and update their models based on new information. Agents maintain local models that are periodically updated through a distributed consensus mechanism. The consensus mechanism ensures that agents converge on a shared understanding of the network and its dynamics. As discussed in (Ramamoorthy et al., 2024), DOL3 consists of four phases: Periodic reset (to handle dynamic nature), Communication (Agents share information), Trust Fusion (Weighted fusion of trust rates with the neighbours) and Learning (update the trust weights with decay). The periodic reset phase is introduced in this framework to ensure non-stationary nature is handled.

Data Collection and Update with Decay: includes agents continuously collecting information about their interactions and relationships with other agents as mentioned in the Section 3.2. This data can include new edges, changes in edge weights, or node attributes. As new data becomes available, agents update their local clustering models. Assign weights to new data points based on the chosen decay mechanism.

In a dynamic social network, the evolution of agents' interaction is crucial in determining the collusion behavior. We considered the decay mechanism as stated by (Reitter and Lebiere, 2012) for updating the information about the social network at every iteration. Decay mechanism is introduced into the DOL3-based graph clustering framework to address the potential influence of outdated information and ensure that the system remains responsive to recent changes. DOL3 iterations consist of the following steps:

- **Periodic Reset.** Regularly reset the trust values to a predefined baseline. This phase is implicitly incorporated into the trust update mechanism. The α parameter in the trust update equation can be adjusted to control the rate at which past information decays. A smaller α will cause older information to have less influence on current trust values, effectively resetting trust over time.

$$T_{ij} = T_{ij} * (1-\alpha) + \alpha * B_0 \tag{6}$$

  In this formulation, the first term $T_{ij} * (1-\alpha)$ decays the previous trust value over time, while the second term $\alpha * B_0$ adds a portion of the baseline trust value.

  By adjusting the value of α and the baseline B_0, you can control the rate of decay and the level to which trust values are reset. For example, a higher α will result in a faster decay, while a higher B_0 will set the trust to a higher baseline value.

  This formulation ensures that trust values are periodically reset to a predefined level, helping to maintain the system's responsiveness to changing conditions and preventing the accumulation of outdated information.

- **Communication.** Agents exchange information about their local models and observations. The communication phase is reflected in the calculation of the weighted average in the global consensus step. Agents exchange information about their local models and trust values, which are then combined to form the global consensus.

- **Trust Fusion.** Agents update their trust in other agents based on the shared information and previous trust values. The trust fusion step is explicitly represented in the equation for updating trust values:

$$T_{ij} = (1-\varepsilon) * T_{ij} + \varepsilon * \sum_{k \in N(i)} w_{ik} * T_{kj} \tag{7}$$

  This equation calculates the updated trust value T_{ij} based on the weighted average of the trust values from neighboring agents a_i and a_j. The first term $(1-\varepsilon) * T_{ij}$ represents the agent's own assessment of trust. The second term $\varepsilon * \sum_{k \in N(i)} w_{ik} * T_{kj}$ represents the influence of neighboring agents on the trust value.

- **Learning.** Agents update their local models using the new data and the updated trust values. The learning phase is incorporated into the model update step: $M_i.fit(D_i)$. This step updates the local model M_i based on the new data D_i. The updated model can then be used to make predictions and influence the agent's behavior. The learning phase incorporates the decay functionality as mentioned above.

$$T_{ij} := T_{ij} * (1-\alpha) + \alpha * \Delta T_{ij} \tag{8}$$

  where ΔT_{ij} is the change in trust between agents a_i and a_j based on recent interactions or observations. The equation essentially updates the trust value T_{ij} by combining the previous trust value with a weighted average of the recent change in trust. The α parameter determines the balance between maintaining existing trust relationships and incorporating new information.

3.4 Global Consensus

Agents periodically exchange their updated clustering models with neighboring agents during the Trust fusion stage of DOL3 as mentioned in Section 3.3. A consensus mechanism is used to combine these models into a global clustering solution. We propose a weighted average approach based on the number of shared edges:

$$C = \frac{\sum_i w_i \, * \, c_i}{\sum_i w_i} \tag{9}$$

where C is the global clustering solution, c_i is the local clustering model of agent a_i, and w_i is the weight of agent a_i in the consensus. Introducing the decay mechanism during the Global consensus would improve the framework's responsiveness to the recent changes in the network.

$$w_i = \alpha^{(t \, - \, t_i)} \tag{10}$$

The Global consensus consists of three components:

- Calculation of time difference: $t - t_i$ calculates the time elapsed since data point i was observed.
- Exponential decay: The term $\alpha^{(t \, - \, t_i)}$ applies exponential decay to the weight. As time passes, the exponent $t - t_i$ increases, causing the weight to decrease exponentially.
- Weight assignment: The calculated value is assigned to w_i , effectively reducing the influence of older data points on the clustering model.

The global consensus model represents a collective view of the system, incorporating the knowledge and insights from all agents. The weights assigned to individual agents reflect their relative importance in the consensus process.

The DOL3 framework allows for continuous adaptation to changing environments and the incorporation of new information.

4 EXPERIMENTAL SETUP

To evaluate the performance of our proposed framework, we conducted a series of experiments using the Mesa agent-based modeling framework. We simulated a dynamic network environment where agents represented nodes in a graph, and their interactions were modeled as edges. We ran the experiment for 500 Monte Carlo simulations to provide more robust result and quantify uncertainty. We used NVIDIA RTX 4000 GPU with 3840 CUDA cores and 8GB GDDR6 memory.

4.1 Hyperparameters and Initial Setup

The number of agents N in the simulation directly affects the scale and complexity of the network. A larger number of agents can lead to more diverse interactions and a richer dataset. We conducted the study by varying the number of agents from 100 to 500. Network Topology plays a vital role on the performance of the algorithm. The structure of the network can influence the dynamics of agent interactions and the formation of clusters (Kołaczek, 2010). Different topologies, such as random graphs, small-world networks, and scale-free networks, was explored to assess their impact on the results.

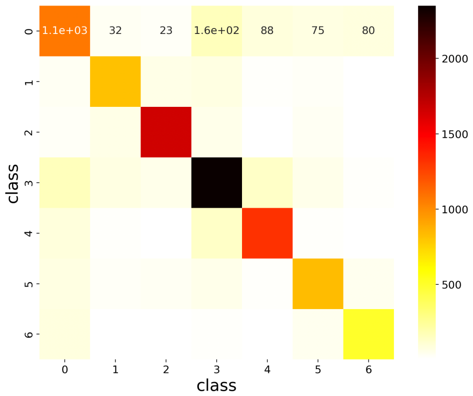

Figure 2: Class connectivity in Cora dataset.

The initial trust values assigned to agents can affect the starting point of the simulation. Random initialization or assigning initial values based on domain knowledge is considered. For example, all the agents' trust scores are initialised as 1, meaning fully trusted in Step 1. The learning rate controls the speed at which agents update their trust values. A higher learning rate can lead to faster adaptation but may also introduce instability. Based on the studies conducted, the learning rate (η) of 15 is considered for the experiment. The decay factor (α) determines how quickly past information is discounted. A higher decay factor can make the system more responsive to recent changes, while a lower decay factor can preserve historical information. We considered the decay factor of $\alpha = 0.5$. The choice of clustering algorithm can impact the quality of the clustering results. Different algorithms, such as Louvain Modularity Optimization, spectral clustering, or hierarchical clustering, is evaluated. We defaulted Louvain Modularity Optimization for local clustering considerations.

4.2 Dataset

We used the Cora dataset (McCallum, 2024), a popular benchmark dataset for graph-based machine learning tasks. The Cora dataset consists of 2708 scientific papers categorized into seven classes: Case-based reasoning, Genetic Algorithms, Neural Net-

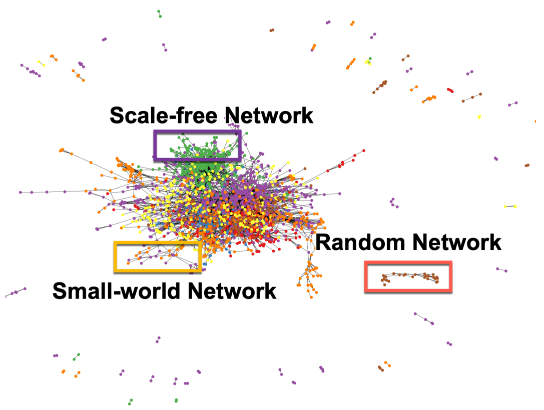

Figure 3: Types of networks in Cora dataset.

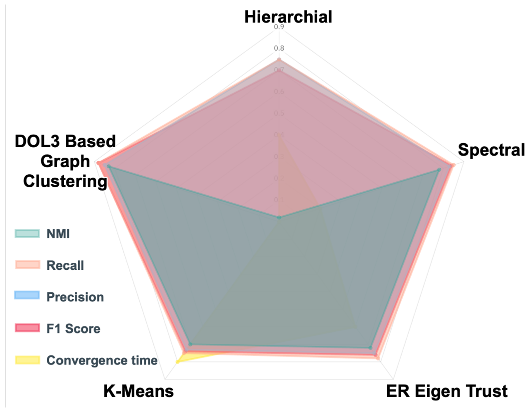

Figure 4: Metric Result Analysis.

works, Probabilistic Methods, Rule-Based Methods, Theory of Computation, and Uncertainty. Each paper is represented by a 1433-dimensional feature vector, capturing the bag-of-words representation of the paper's content. Figure 3 illustrates the different types of social networks in Cora dataset as identified during the simulation. The dataset provides a well-defined graph structure representing citation relationships between papers, which is similar to the dynamic networks in which our framework is intended to operate.

Figure 2 explains how the network types are arrived at. The general patterns that are identified include:

- Intra-class Connectivity: Papers within the same class are likely to have stronger connections than papers from different classes. This is because papers on similar topics often cite each other more frequently
- Inter-class Connectivity: There may be some connections between papers from different classes, especially if the topics are related or complementary.
- Hierarchical Structure: The class connectivity might exhibit a hierarchical structure, with some classes being more central or influential than others.

4.3 Baseline Evaluation

The state-of-the-art models like hierarchical clustering, K-means clustering, spectral clustering and ER EigenTrust based clustering are chosen based on the literature surveys from (Nezamoddini and Gholami, 2022) and (Marciano, 2024) as mentioned in Section 2. These models are run against the simulated dynamic environment with Cora dataset.

4.4 Evaluation Method

Larsen and Aone's F Score was the evaluation method used for comparing the clustering as refered by (Johnson et al., 2013). Since the Cora dataset already provides the labels which can be looked at as clustering, the objective of the evaluation would be to check the identical clustering output. An F score closer to 1 would mean that the clustering is identical. Higher the F-Score, better the algorithm works in identifying the colluding agents thereby penalising the malicious agents.

Apart from the F score, we used the typical metrics like Normalized Mutual Information (NMI) to measure the accuracy of the clustering results (Newman et al., 2020). Convergence time was also used to evaluate the performance of the algorithm along with the usual metrics like precision and recall to evaluate the performance of agent trust assessment and malicious agent detection.

The F Score $F(C,L)$ for the Class C and Label L is calculated using the below equation:

$$F(C,L) = 2 * \frac{P(C,L).R(C,L)}{P(C,L)+R(C,L)} \quad (11)$$

where $P(C,L)$ and $R(C,L)$ are the precision and recall. This is the F Score for a specific label. For calculating the $FScore$ for the entire cluster, we ideally take the weighted average over the labeling of every class in the cluster.

4.5 Result

In this section, we present the result of running the 500 Monte Carlo simulations on Cora dataset with manipulation of edges and vertices to simulate the dynamic environment. Table 1 gives the summary of the metrics across each of the approaches. From Figure

Table 1: Comparison of metrics on all approaches.

Approach	NMI	Recall	Precision	F1- Score	Convergence Time (s)
Hierarchical Clustering	0.6	0.75	0.75	0.7	15
Spectral Clustering	0.76	0.85	0.83	0.84	20
ER EigenTrust	0.72	0.78	0.75	0.65	10
K-Means Clustering	0.70	0.75	0.73	0.74	5
DOL3 Enhanced Graph Clustering	0.85	0.88	0.87	0.88	25

4, it is clear that DOL3 based Graph Clustering approach outperforms the baselines in terms of NMI, indicating better clustering accuracy. Spectral Clustering is likely to perform well due to its ability to capture complex structures in the graph. ER EigenTrust has lower performance due to its reliance on trust relationships and the potential for manipulation. The area where DOL3 based approach under-performs is the time it takes to converge on a consensus. Convergence Time is expected to be higher for DOL3-Enhanced Graph Clustering. This is due to the additional steps involved in DOL3, such as trust fusion, learning, and global consensus. These steps require iterative calculations and communication between agents, which can increase the computational overhead. From the table 1, we could see that DOL3 integration improves the accuracy by 11.8%. The formula that was used to calculate was : Percentage Improvement = ((Accuracy of our Approach - Accuracy of Best Baseline (Spectral Clustering)) / Accuracy of Best Baseline) * 100

However, the trade-off is that DOL3 offers improved adaptability and robustness, making it suitable for dynamic environments where relationships and interactions may change over time.

We extended this experiment to further test on DOL3 based Graph clustering approach's ability to respond when the number of agents and the social graph is changed over time. In addition to the evaluation metrics mentioned in the Table 1, we introduced further metrics to measure the performance of the algorithms in dynamic environment. As seen in the result 5, the DOL3 based Graph Clustering approach performs consistently over the time with number of cluster changing along with the social graph.

The analysis of the Figure 5 and Figure 6 clearly shows that the DOL3 algorithm demonstrated exceptional resilience in navigating dynamic network environments. Its four-phase structure proved instrumental in swiftly adapting to system changes, outperforming other algorithms that struggled to cope with the uncertain number of malicious agents.

- Periodic Reset: This phase ensured that outdated information was regularly discarded, allowing the algorithm to remain responsive to evolving conditions.

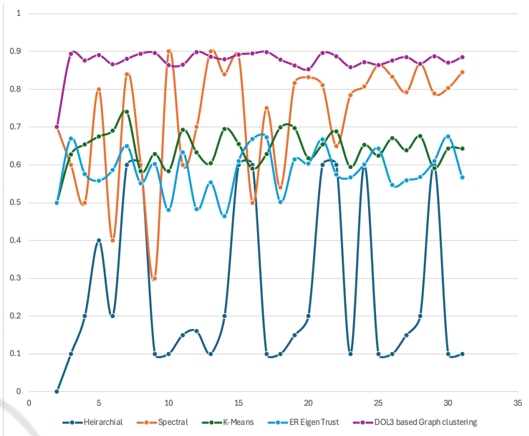

Figure 5: Dynamic Network F1 Score.

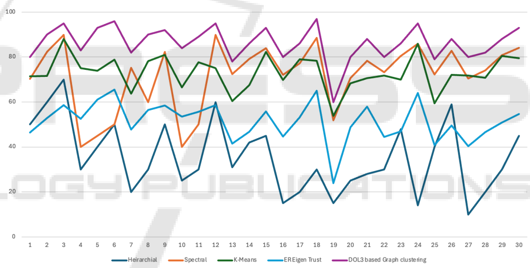

Figure 6: Dynamic Network Algorithm Resilience.

- Communication: Efficient communication between agents facilitated the rapid dissemination of critical information, enabling timely updates and adjustments.
- Trust Fusion: By carefully combining trust values from multiple sources, DOL3 mitigated the impact of potential manipulation by malicious agents.
- Learning: The continuous learning mechanism allowed the algorithm to adapt its models to changing circumstances, ensuring its effectiveness in dynamic settings.

It could be seen from the sharp down slides in Figure 6 that DOL3 based Graph Clustering takes very minimal time to respond to dynamic evolution of the network. In contrast, other algorithms often faced challenges in handling the uncertainty associated with fluctuating numbers of malicious agents. Their inabil-

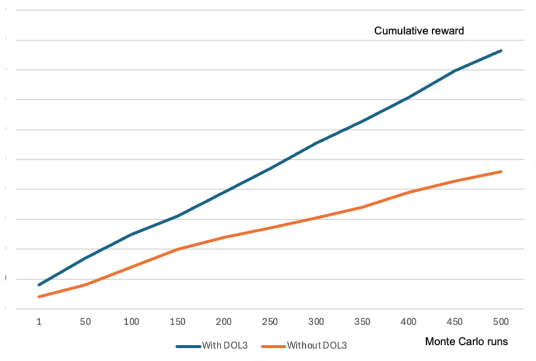

Figure 7: Impact of DOL3 on performance.

ity to adapt quickly to these changes hindered their performance and compromised the system's security.

4.6 Ablation Studies

We performed ablation studies to understand the importance of some of the parameters in the DOL3 framework. The study included measuring cumulative rewards (identifying non-malicious agents) across 50 agents over several Monte Carlo simulation runs. In each of the runs, the network social structure was changed at random to replicate a real-life scenario.

4.6.1 Impact of DOL3

We compared the performance of our approach with a variant that excluded the DOL3 component. Results showed that DOL3 significantly improved the accuracy and adaptability of the system, especially in dynamic environments. It is noticed that DOL3 can help to reduce the impact of noise and uncertainty in the data. By incorporating a learning rate and decay factor, the system can gradually update its beliefs based on new information, avoiding sudden changes that might be caused by outliers or temporary fluctuations. DOL3 can make the system more robust to malicious agents and adversarial attacks. By continuously updating trust values and detecting anomalies, DOL3 can help identify and mitigate the influence of malicious actors. This is evident from the rate of climb of the cumulative reward as shown in the Figure 7. Overall, the DOL3 component is essential for the success of our proposed framework, providing the necessary adaptability and robustness to address the challenges of dynamic networks. By integrating DOL3 with graph clustering and trust fusion, we are able to create a more effective and reliable system for trust and reputation assessment.

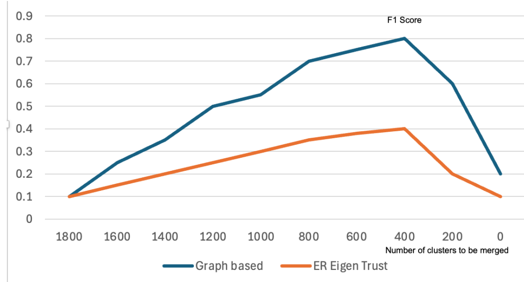

Figure 8: Impact of Graph Clustering on F1 score.

4.6.2 Impact of Graph Clustering

We compared the performance with a version that did not use graph clustering. The results demonstrated that graph clustering was crucial for identifying meaningful communities of agents and improving the accuracy of trust and reputation assessment. We compared the number of clusters remaining to be merged over the period of time between a non graph clustering technique like ER EigenTrust with the Graph clustering approach. While Graph Clustering can reveal hidden patterns through the network structure, ER EigenTrust method depends on the trust recommendation from other agents. ER EigenTrust seemed to be extremely vulnerable to noise and manipulations from malicious agents. We simulated more clusters and compared the number of clusters to be merged with the F1 score at that point in time. It is seen that the trend on the F1 score remains exactly the same. However, the Graph based clustering has better F1 score through out compared to ER EigenTrust as seen in the Figure 8. Graph clustering is generally a more effective approach for identifying communities and assessing trust in complex networks, as it directly analyzes the structural properties of the network rather than relying solely on recommendations. However, the choice between graph clustering and ER EigenTrust may depend on the specific characteristics of the network and the desired outcomes.

4.6.3 Impact of Trust Fusion

Trust Fusion is an important phase in DOL3 framework as mentioned in the Section 3.3. The agents consider the neighbouring agents' trust values over weights to finalise the own assessment. We performed ablation studies to check on the impact of not having the trust fusion in the DOL3 framework. As seen in the Figure 9, the key benefits of Trust Fusion includes reduced uncertainty, improved accuracy and faster adaptability. During this ablation study, we recognized that trust may be context-dependent, and different factors may influence trust in different situations. Considering the context while sharing the fea-

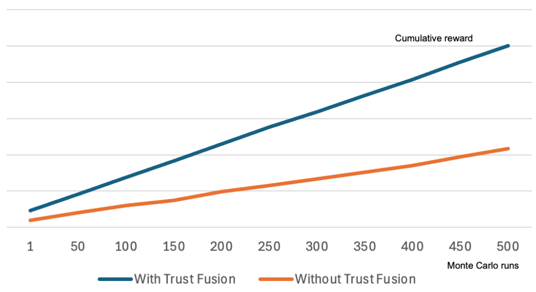

Figure 9: Impact of Trust Fusion on cumulative reward.

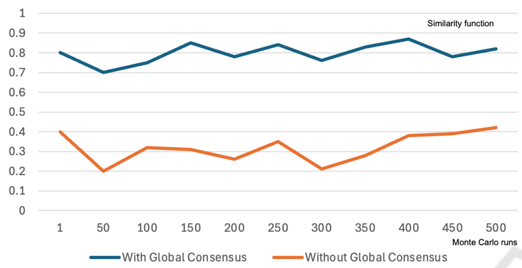

Figure 10: Impact of Global Consensus on Similarity function.

ture information would have more closer trust value than the one without context.

4.6.4 Impact of Global Consensus

Global consensus, the process of aggregating the local models of individual agents into a shared understanding, is another critical component of our proposed framework. It plays a significant role in ensuring consistency, coherence, and accuracy in trust and reputation assessment. Some of the key benefits of Global Consensus includes:

- Consistency: Global consensus helps to ensure that agents have a consistent understanding of the network and its dynamics. This can prevent inconsistencies and conflicts in decision-making.
- Coherence: By combining the insights from multiple agents, global consensus can provide a more coherent and comprehensive view of the network.
- Accuracy: Global consensus can improve the accuracy of trust and reputation assessments by leveraging the collective knowledge of the system.
- Resilience: Global consensus can make the system more resilient to the influence of individual outliers or malicious agents.
- Scalability: Global consensus can be applied to large-scale networks, as it allows for distributed processing and information sharing.

To understand the consistency, the similarity function mentioned in the equation (4) is considered. This exhibits the similarity among the agent's features thereby measuring the clustering among the agents group.

From the Figure 10, it can be seen that the system exhibits consistency when Global Consensus is considered in the framework.

5 CONCLUSION

We have demonstrated on how to solve the collusion problem in Multi-Agent system in a dynamic environment. While the DOL3 integrated Graph based clustering approach takes more computation cycles, the trade off with regards to stability, performance and resilience is more. Our experimental results demonstrate the superior performance of our approach compared to traditional methods, particularly in terms of accuracy, adaptability, and robustness. The DOL3 framework effectively addresses the challenges of dynamic environments by enabling agents to continuously learn and update their models.

Graph clustering provides a valuable tool for identifying communities of agents and assessing their trustworthiness. The combination of DOL3 and graph clustering offers a robust and scalable solution for trust and reputation assessment. Our approach outperforms traditional methods in terms of accuracy, particularly when dealing with dynamic networks and changing agent behaviors.

One of the topics that is not considered in this paper is that of sharing the context while identifying the malicious agent. The future direction would be to explore techniques for incorporating contextual information into the consensus mechanism to improve the accuracy and relevance of the global model. One of the possible approaches could be to extend the feature vector to include the context without losing the privacy on sharing the information with other agents. Another future direction would be to investigate methods to optimize the computational complexity of the algorithm, particularly for large-scale networks. We could also possibly extend the study to improve the computational efficiency through parallel and distributed implementations. We could also explore contextual trust fusion process to make more informed decisions.

REFERENCES

Barbosa, R., Santos, R., and Novais, P. (2023). Trust-based negotiation in multiagent systems: a systematic review. In *International Conference on Practical Ap-*

plications of Agents and Multi-Agent Systems, pages 133–144. Springer.

Bonald, T., C. B. G. A. . H. A. (2018). Hierarchical graph clustering using node pair sampling. *Phys. Rev. E*.

Despalatović, L., Vojković, T., and Vukicević, D. (2014). Community structure in networks: Girvan-newman algorithm improvement. In *2014 37th international convention on information and communication technology, electronics and microelectronics (MIPRO)*, pages 997–1002. IEEE.

Drawel, N., Bentahar, J., Laarej, A., and Rjoub, G. (2022). Formal verification of group and propagated trust in multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 36(1):19.

Galluccio, L., Michel, O., Comon, P., and Hero, A. O. (2012). Graph based k-means clustering. *Signal Processing*, 92(9):1970–1984.

Ghosh, S., Halappanavar, M., Tumeo, A., and Kalyanarainan, A. (2019). Scaling and quality of modularity optimization methods for graph clustering. In *2019 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–6. IEEE.

Govindaraj, R., Govindaraj, P., Chowdhury, S., Kim, D., Tran, D.-T., and Le, A. N. (2021). A review on various applications of reputation based trust management. *International Journal of Interactive Mobile Technologies*, 15(10).

Gupta, S. and Sundaram, S. (2023). Moving-landmark assisted distributed learning based decentralized cooperative localization (dl-dcl) with fault tolerance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 6175–6182.

Johnson, D. M., Xiong, C., Gao, J., and Corso, J. J. (2013). Comprehensive cross-hierarchy cluster agreement evaluation. In *Workshops at the Twenty-Seventh AAAI Conference on Artificial Intelligence*.

Johnson, J. and Sokol, D. D. (2020). Understanding ai collusion and compliance. *Cambridge Handbook of Compliance,(D. Daniel Sokol & Benjamin van Rooij, editors),(Forthcoming)*.

Khalid, R., Samuel, O., Javaid, N., Aldegheishem, A., Shafiq, M., and Alrajeh, N. (2021). A secure trust method for multi-agent system in smart grids using blockchain. *IEEE Access*, 9:59848–59859.

Kołaczek, G. (2010). Social network analysis based approach to trust modeling for autonomous multi-agent systems. In *Agent and Multi-agent Technology for Internet and Enterprise Systems*, pages 137–156. Springer.

Liu, J. and Han, J. (2018). Spectral clustering. In *Data clustering*, pages 177–200. Chapman and Hall/CRC.

Marciano, A. (2024). Trust and reputation systems: detection of malicious agents and a novel equilibrium problem.

McCallum, A. (2024). Cora.

Mohanty, P. (2020). *A computational approach to identify covertness and collusion in social networks*. PhD thesis, University of Minnesota.

Newman, M. E. J., Cantwell, G. T., and Young, J.-G. (2020). Improved mutual information measure for clustering, classification, and community detection. *Phys. Rev. E*, 101:042304.

Nezamoddini, N. and Gholami, A. (2022). A survey of adaptive multi-agent networks and their applications in smart cities. *Smart Cities*, 5(1):318–347.

Ontañón, S. (2020). An overview of distance and similarity functions for structured data. *Artificial Intelligence Review*, 53(7):5309–5351.

Ramamoorthy, H., Gupta, S., and Sundaram, S. (2024). Distributed Online Life-Long Learning (DOL3) for Multi-agent Trust and Reputation Assessment in E-commerce.

Reitter, D. and Lebiere, C. (2012). Social cognition: Memory decay and adaptive information filtering for robust information maintenance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, pages 242–248.

Rodríguez, M. J. G., Rodríguez-Montequín, V., Ballesteros-Pérez, P., Love, P. E., and Signor, R. (2022). Collusion detection in public procurement auctions with machine learning algorithms. *Automation in Construction*, 133:104047.

Rossetti, G. and Cazabet, R. (2018). Community discovery in dynamic networks: a survey. *ACM computing surveys (CSUR)*, 51(2):1–37.

Seifikar, M., Farzi, S., and Barati, M. (2020). C-blondel: an efficient louvain-based dynamic community detection algorithm. *IEEE Transactions on Computational Social Systems*, 7(2):308–318.

Sievers, M. (2020). *Modeling Trust and Reputation in Multiagent Systems*. Springer International Publishing, Cham, 1st edition.

Wang, W., Tang, T., Xia, F., Gong, Z., Chen, Z., and Liu, H. (2022). Collaborative filtering with network representation learning for citation recommendation. *IEEE Transactions on Big Data*, 8(5):1233–1246.

Wu, L., Zhang, Q., Chen, C.-H., Guo, K., and Wang, D. (2020). Deep learning techniques for community detection in social networks. *IEEE Access*, 8:96016–96026.