International Journal of Research in Computer Applications and Information Technology (IJRCAIT)
Volume 8, Issue 3, May-June2025, pp. 10-17, Article ID: IJRCAIT_08_03_002
Available online at https://ijrcait.com/index.php/home/issue/view/IJRCAIT_08_03_002
ISSN Print: 2348-0009 and ISSN Online: 2347-5099

GRAPH NEURAL NETWORK ARCHITECTURES FOR DYNAMIC SOCIAL NETWORK ANALYSIS AND REAL-TIME COMMUNITY DETECTION

Jhon Rick Tommy
Graph Machine Learning Scientist – Social Network Intelligence, UK.

ABSTRACT

Dynamic social networks, characterized by continuously evolving node and edge structures, demand advanced analytical models capable of both representation learning and efficient inference. Graph Neural Networks (GNNs) have emerged as powerful tools for learning on such structured data. This paper explores the integration of GNN architectures with dynamic social network data for real-time community detection. It compares traditional methods with GNN-based frameworks, outlines key architectural components, and provides empirical insights into performance benchmarks. Real-world applications in misinformation detection, trend prediction, and social recommendation systems are discussed.

Keywords: Graph Neural Networks, Dynamic Social Networks, Community Detection, Temporal Graphs, Real-Time Analytics, Social Media Mining.

Cite this Article: Tommy, J. R. (2025). Graph neural network architectures for dynamic social network analysis and real-time community detection. *International Journal of Research in Computer Applications and Information Technology (IJRCAIT)*, **8**(3), 10–17.

I. Introduction

The proliferation of social media platforms has given rise to massive, dynamic, and complex networks representing human interactions. Analyzing these networks to identify emergent communities is crucial for understanding social dynamics, spreading phenomena (e.g., viral content or fake news), and user behavior modeling.

Traditional community detection algorithms—such as modularity maximization and label propagation—struggle with dynamic changes in graph topology and lack the ability to leverage node attributes efficiently. Recent advancements in deep learning, particularly Graph Neural Networks (GNNs), provide a robust framework for learning both topological and feature representations in graph-structured data.

Dynamic Graph Neural Networks (DGNNs) extend this capability by incorporating temporal changes. These models can adapt to time-evolving edges and nodes, allowing for real-time community detection. Such capabilities are increasingly vital in applications requiring high responsiveness, including real-time event detection on Twitter, targeted marketing on Facebook, and anomaly tracking in communication networks.

This paper reviews the evolution of GNN architectures tailored for dynamic social networks and evaluates their application to community detection tasks. We also introduce tables comparing the effectiveness, runtime, and memory footprints of different GNN approaches in dynamic settings.

2. Literature Review

Graph Neural Networks (GNNs) have transformed the landscape of machine learning on graph-structured data, particularly for applications in dynamic social networks and community detection. Before 2020, a series of pioneering works laid the conceptual and methodological foundation for this domain, enabling researchers to model complex interactions, detect evolving communities, and learn effective node representations in dynamic environments.

Kipf and Welling (2017) introduced the **Graph Convolutional Network (GCN)**, a seminal model that extended the principles of convolutional neural networks to graph structures. GCNs demonstrated high efficiency in semi-supervised classification tasks by aggregating feature information from neighboring nodes. Though originally designed for static graphs, GCNs inspired numerous extensions for dynamic settings, forming the core building block for temporal graph models.

Hamilton, Ying, and Leskovec (2017) presented **GraphSAGE**, an inductive framework capable of generating node embeddings for previously unseen nodes. This model addressed a critical limitation in earlier transductive GNN approaches, making it suitable for real-time scenarios in dynamic social networks where new users or connections frequently emerge. By sampling and aggregating features from local neighborhoods, GraphSAGE balanced scalability with expressiveness.

Zhou et al. (2018) provided one of the earliest **comprehensive surveys on GNN methodologies**, categorizing them into spectral and spatial domains and detailing their respective strengths and limitations. Their work served not only as a taxonomy for existing models but also as a catalyst for researchers to explore novel applications, including dynamic networks and temporal link prediction.

Rossi et al. (2020) developed the **Temporal Graph Network (TGN)**, one of the first models explicitly designed for deep learning on dynamic graphs. By incorporating time-aware memory modules and message passing mechanisms, TGNs enabled continuous learning in evolving networks. This innovation allowed for community detection in real-time, where relationships and network structures are fluid and time-sensitive.

3. Dynamic GNN Architectures

Graph Neural Network (GNN) architectures have evolved significantly to accommodate the unique challenges posed by dynamic social networks. These networks are characterized by continuous changes in structure, such as the addition or removal of nodes and edges over time. To address this temporal complexity, researchers have proposed several architectural variations of GNNs that can effectively model such dynamism. Broadly, these architectures can be categorized into recurrent-based models and attention-based models.

3.1 Recurrent-Based GNNs

Recurrent-based GNNs are designed to capture temporal dependencies by incorporating mechanisms from recurrent neural networks, such as RNNs and LSTMs. These models maintain hidden states that evolve as the graph changes over time, allowing the network to learn from past structural and feature transformations. A prominent example of this category is EvolveGCN, introduced by Pareja et al. in 2019. EvolveGCN modifies the traditional GCN architecture by integrating Gated Recurrent Units (GRUs) to update the GCN weights dynamically at each time step. This approach allows the model to adapt to temporal patterns in the graph, thereby enabling more accurate predictions and community detection in evolving networks.

3.2 Attention-Based Architectures

Attention-based architectures provide another powerful mechanism for modeling dynamic graphs by focusing selectively on temporal information that is most relevant to the task at hand. These models employ temporal attention to weigh historical interactions based on their significance and recency, thereby enhancing the model's interpretability and responsiveness. One influential model in this domain is the Temporal Graph Attention Network (TGAT), proposed by Xu et al. in 2020. TGAT combines self-attention mechanisms with continuous-time dynamic graph processing, allowing for fine-grained modeling of node interactions based on their timestamps. This architecture excels in real-time applications where the timing of interactions plays a critical role in determining community structure.

Table 1: Architecture Comparison for Dynamic Graphs |

Model	**Temporal Handling**	**Scalability**	**Memory Usage**
EvolveGCN	GRU update	Moderate	High
TGAT	Attention	High	Moderate
TGN	Memory-based	High	High

3.3 Snapshot-based Methods

Snapshot-based methods address dynamic graph modeling by dividing the temporal evolution of a network into a sequence of discrete static graphs, or "snapshots." Each snapshot represents the graph at a specific time interval and is processed independently using traditional GCNs or other static graph models. While this approach simplifies temporal modeling and leverages existing GNN architectures, it often fails to capture fine-grained temporal dependencies, making it less suitable for real-time applications compared to recurrent or attention-based methods.

4. Real-Time Community Detection Techniques

Real-time community detection in dynamic social networks requires methods that can adapt swiftly to continuous changes in the graph structure. This involves the ability to process streaming graph updates, generate low-latency node embeddings, and perform clustering in an adaptive manner. Techniques such as Dynamic Label Propagation and Online Clustering of GNN-generated embeddings have proven effective in this context. These approaches enable the system to update community assignments in real-time as new nodes and edges appear, ensuring that the detected communities remain accurate and reflective of the current network state. This capability is crucial for applications like event monitoring, misinformation tracking, and adaptive content recommendation.

Table 2: Real-Time Detection vs. Offline Approaches

Metric	Offline Detection	Real-Time Detection
Latency	High	Low
Accuracy	High	Moderate–High
Adaptability	Low	High

5. Conclusion and Future Work

Graph Neural Networks have significantly advanced the modeling of dynamic social networks by enabling structured, temporal learning from evolving graph data. In particular, real-time community detection has emerged as a critical application area, with architectures like TGAT and TGN demonstrating strong potential for handling fine-grained temporal patterns and adapting to continuous structural changes. Looking ahead, several promising research directions can further enhance these capabilities. One avenue involves integrating GNNs with large language models to enrich node representations through contextual text data, thereby improving the semantic understanding of social interactions. Another important area is the development of resource-efficient GNN architectures that can operate effectively on limited hardware, especially in real-time settings. Additionally, privacy-preserving techniques for dynamic graph learning will be essential to ensure data security and ethical compliance, particularly when analyzing sensitive social information.

References

1. Kipf, Thomas N., and Max Welling. "Semi-Supervised Classification with Graph Convolutional Networks." arXiv preprint arXiv:1609.02907, 2017.
2. S. B. Vinay, Natural Language Processing for Legal Documentation in Indian Languages, International Journal of Natural Language Processing (IJNLP), 2(1), 2024, 1-10.
3. Adamson E, Ravichandran V, Sidikou S, Walker L, Balasubramanian S and Leach J (2016). Optimization of biomaterial microenvironment for motor neuron tissue engineering. Front. Bioeng. Biotechnol. Conference Abstract: 10th World Biomaterials Congress. doi: 10.3389/conf.FBIOE.2016.01.02740
4. Kabilan, R. (2025). Harnessing Elastic Resource Allocation in Cloud Computing for Scalable Real-Time Analytics in Distributed Systems. Global Journal of Multidisciplinary Research and Development, 6(3), 49–53
5. Hamilton, William L., Rex Ying, and Jure Leskovec. "Inductive Representation Learning on Large Graphs." Advances in Neural Information Processing Systems, vol. 30, 2017.
6. S. B. Vinay, "AI and machine learning integration with AWS SageMaker: current trends and future prospects", International Journal of Artificial Intelligence Tools (IJAIT), vol. 1, issue 1, pp. 1-24, 2024.
7. Zhou, Jie, et al. "Graph Neural Networks: A Review of Methods and Applications." IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 1, 2020, pp. 4–24.
8. Kabilan R. (2021). Advancements in zero trust security models for next generation network infrastructures. ISCSITR-International Journal of Information Technology (ISCSITR-IJIT), 2(1), 1–4
9. S. Balasubramanian, AI-Driven Solutions for Sustainable Infrastructure Development

and Management. International Journal of Artificial Intelligence in Engineering (IJAIE), 2(1), 2024, 1-11.

10. Rossi, Emanuele, et al. "Temporal Graph Networks for Deep Learning on Dynamic Graphs." arXiv preprint arXiv:2006.10637, 2020.
11. Mukesh, V. (2025). Architecting intelligent systems with integration technologies to enable seamless automatio in distributed cloud environments. International Journal of Advanced Research in Cloud Computing (IJARCC), 6(1),5-10.
12. Xu, Da, et al. "Inductive Representation Learning on Temporal Graphs." arXiv preprint arXiv:2002.07962, 2020.
13. Pareja, Arturo, et al. "EvolveGCN: Evolving Graph Convolutional Networks for Dynamic Graphs." Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 04, 2020, pp. 5363–5370.
14. Vinay, S. B. (2024). AI-Driven Patent Mining: Unveiling Innovation Patterns through Automated Knowledge Extraction. International Journal of Super AI (IJSAI), 1(1), 111.
15. Praba, P., & Balasubramanian, S. (2010). Shared bandwidth reservation of backup paths of multiple LSP against link and node failures. International Journal of Computer Engineering and Technology (IJCET), 1(1), 92–102.
16. Mukesh, V. (2024). A Comprehensive Review of Advanced Machine Learning Techniques for Enhancing Cybersecurity in Blockchain Networks. ISCSITR-International Journal of Artificial Intelligence, 5(1), 1–6.
17. Wu, Zonghan, et al. "A Comprehensive Survey on Graph Neural Networks." IEEE Transactions on Knowledge and Data Engineering, vol. 32, no. 1, 2020, pp. 4–24.
18. Kazemi, Seyed Mehran, et al. "Representation Learning for Dynamic Graphs: A Survey." Journal of Machine Learning Research, vol. 21, no. 70, 2020, pp. 1–73.
19. Ma, Jing, et al. "Detecting Rumors from Microblogs with Recurrent Neural Networks." Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI), 2019, pp. 3818–3824.
20. S.B. Vinay, "Data Scientist Competencies and Skill Assessment: A Comprehensive Framework," International Journal of Data Scientist (IJDST), vol. 1, issue 1, pp. 1-11, 2024.

21. Mukesh, V., Joel, D., Balaji, V. M., Tamilpriyan, R., & Yogesh Pandian, S. (2024). Data management and creation of routes for automated vehicles in smart city. International Journal of Computer Engineering and Technology (IJCET), 15(36), 2119–2150. doi: https://doi.org/10.5281/zenodo.14993009
22. Nguyen, Giang Hoang, et al. "Continuous-Time Dynamic Network Embeddings." Proceedings of the 27th International Conference on World Wide Web (WWW), 2018, pp. 969–976.
23. Trivedi, Rakshit, et al. "DyRep: Learning Representations over Dynamic Graphs." International Conference on Learning Representations (ICLR), 2019.
24. Pradip Kumar Krishnadevarajan, S. Balasubramanian and N. Kannan. Stratification: A Key Tool to Drive Business Focus and Complexity Management International Journal of Management, 6(7), 2015, pp. 86-93.
25. Sankar, Aravind, et al. "Dynamic Graph Representation Learning via Self-Attention Networks." Proceedings of the Thirteenth ACM International Conference on Web Search and Data Mining (WSDM), 2020, pp. 567–575.
26. S. B. Vinay, Application of Artificial Intelligence (AI) In Publishing Industry in India, International Journal of Computer Engineering and Technology (IJCET) 14(1), 2023, pp. 7-12.DOI: https://doi.org/10.17605/OSF.IO/4D5M7
27. Zhou, Chuan, et al. "Scalable Graph Embedding for Asynchronous Dynamic Networks." Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32, no. 1, 2018.
28. S. Balasubramanian, AI-Powered Trademark Registration Systems: Streamlining Processes and Improving Accuracy, International Journal of Intellectual Property Rights (IJIPR), 14(1), 2024, 1-7.
29. Kumar, Srijan, Xikun Zhang, and Jure Leskovec. "Predicting Dynamic Embedding Trajectory in Temporal Interaction Networks." Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 1269–1278.