



Quantum Logic Gates for Accelerating Graph Convolutional Network Operations

Dr.M. Kasiselvanathan^{1*}, Dr.M. Kalaivani², Dr.P. Ambika³, K. Keerthika⁴

1* Associate Professor, Department of ECE, Sri Ramakrishna Engineering College, Coimbatore, Tamil Nadu, India.

Email-id: drkasiselvanathanvkm@gmail.com: kasiselvanathan.m@srec.ac.in

2Associate Professor, Faculty of Management, SRM Institute of Science and Technology, Vadapalani campus, Chennai, Tamil Nadu, India. Email id: drkalaimba@gmail.com, Orcid id: <https://orcid.org/0000-0002-3593-1417>

3Assistant Professor, Department of Commerce, Meenakshi College of Arts and Science, Meenakshi Academy of Higher Education and Research, Chennai, Tamil Nadu, India. ambikap@maher.ac.in

4 Assistant Professor, Department of Computer Science, Meenakshi College of Arts and Science, Meenakshi Academy of Higher Education and Research, Chennai, Tamil Nadu, India.keerthikak@maher.ac.in

*Corresponding author: Email: kasiselvanathan.m@srec.ac.in

Abstract

Graph Convolutional Networks (GCNs) have been widely used to learn from graph structured data, but GCNs suffer from a high complexity in the handling of large-scale graphs, since processing large graphs demands recursive aggregation of features and high dimensional node embeddings. Scalability can become a tight bottleneck in traditional CPU/GPU implementations, which detracts from their ability to be used in real-time and resource-constrained settings. In this study, Quantum-assisted GCN (QGCN) is proposed as a quantum-assisted GCN model with the help of quantum logic gates, thereby speeding up the graph convolution operation. The idea proposed in the paper is to represent the queries and graph features as qubits and use a sequence of both single and multi-qubit gates such as Hadamard, CNOT and rotation gates with learned parameters to convolute a graph over qubits simultaneously. Once measured, the quantum state is represented classically, in the form of features, which are then used to feed downstream node classification tasks. On the benchmark datasets Cora, PubMed, and Citeseer, experiments show that the quantum-assisted GCN delivers substantial speed-up over a classical implementation, approaches the performance of the GPU-optimized implementation, and maintains classification within a 0.5–1% margin. If NISQ simulators are to scale up to very large graphs, there are resource utilization issues, which is manageable for current simulators given the number of qubits and gate depth. These results show concrete compromises between acceleration, accuracy preservation, and quantum resource efficiency. The results of this study could be useful for designing hybrid quantum-classical graph learning architectures and represent one possible method for performing faster GCN operations in both academic research and applications. Extensions and developments are focused on multi-layer quantum GCNs as well as hybrid architectures for large-scale GWs and real-time deployment strategies.

Keywords Graph Convolutional Networks, Quantum Logic Gates, Quantum Computing, Hybrid Quantum-Classical Networks, Node Classification, NISQ Devices, Graph Learning Acceleration.

1. Introduction

Graph Convolutional Networks (GCNs) have recently shown that they are a potent approach to deep learning from graph-structured data and can be effectively applied to social networks, recommendation systems and molecular analysis. However, GCNs also face challenges of high dimensionality of node features and recursive operations of aggregating neighborhood information, which may be accompanied by a significant amount of memory usage and computation delay on large-scale graphs. Most graph-based networks, which are trained on classical computers using CPUs and GPUs, have limited scaling and real-time use for large graph datasets [7][8].

A promising paradigm to overcome these limitations is provided by quantum computing, which takes advantage of quantum parallelism and entanglement to perform certain computations more efficiently than classical computations. Specifically, quantum logic gates can execute operations on superposed states, and thus could speed up operations like matrix multiplication and graph convolution which form part of GCN architectures. The

coupling between quantum circuits and GCNs pave the way for even faster and efficient graph learning with accuracy and integrity[9].

This research seeks to devise and test quantum logic gates designed especially for GCN operations. The goal is to learn about the potential of quantum computation to improve the efficiency of GCNs, decrease the number of computations required, and understand practical considerations for implementing hybrid models that combine quantum and classical computation on large-scale graphs.

In Section I, we present challenges for GCN and quantum acceleration. The classical and quantum-assisted research on GCN is summarized in Section II. The proposed quantum logic gate framework and algorithm are discussed in Section III. Experiments, results and discussion are presented in Section IV. The paper ends with the conclusion, contributions and limitations, and further work.

2. Literature Review

Much work has been done to improve the computation efficiency of Graph Convolutional Networks (GCNs). Optimised sparse matrix multiplications, sampling strategies over large scale graphs, and GPU-based parallelisation are examples of classical methods to accelerate the computation [5] [6][10]. However, traditional architectures are still plagued by drawbacks such as limited node and edge capacity, high latency, and memory constraints that hinder processing efficiency.

One other paradigm that can accelerate some linear algebra operations of importance in GCNs is quantum computing. Working on qubits which are in superposition, quantum logic gates can also provide huge parallelism, which may lower the complexity of graph convolutions. Research works on graph learning based on quantum graph and graph learning based on circuits have been studied and have shown promise in applications like image classification [1][3]. Meanwhile, hybrid quantum-classical networks have also been developed that sit on a sweet middle-ground between methods that are quantum accurate or classically resource-efficient [2] [4]. The recent research has been focused on quantum-assisted GCN architectures, as mentioned above, like XENet and spatial quantum graph convolutions, which have shown the feasibility of encoding adjacency and feature matrices as quantum states [5][11]. Currently, however, implementation of quantum logic gates that specifically could be used to increase the overall speed of large-scale GCN operations is limited. Many models tend to lack of scalability and optimisation in the hardware domain, and there is room for research where the quantum gate is integrated directly into the GCN layers, so that the accuracy is maintained along with speed-up[12]. This inspires the proposed study to develop quantum logic gates that will be specifically tailored to GCN acceleration, and assess their computational and learning performance.

3. Proposed Methodology

In this paper, we present a protocol to insert quantum logic gates into Graph Convolutional Network (GCN) layers for fast graph feature aggregation. The framework converts traditional matrix multiplication in the GCN layers to quantum circuits that perform graph convolutions on qubit registers, thus allowing to compute in parallel over superposed node features. The entire architecture is composed of three main components: (i) encoding of nodes and edge features to the quantum realm, (ii) graph convolution operations with access to quantum resources, and (iii) measurement of the quantum states and classical post-processing for the downstream tasks.

Often-used single and multi-qubit gates such as Hadamard (H), Controlled-NOT (CNOT) and parameterized rotation gates (Rx, Ry, Rz) are used in the design of the quantum circuit. The amplitude/rotation angles of qubits represent properties of nodes, and the pattern of entanglement between qubits is used to show the connectivity of the graph. For each GCN layer the classical combination of the neighbouring node features is effectively executed via the gates. Measurement projects the quantum state into classical vectors that are transmitted to activation function and output layers.

The simulation part of the environment uses quantum packages in Python (e.g. Qiskit and PennyLane) to enable hybrid quantum and classical computing and seamless compatibility with PyTorch or TensorFlow for gradient-based optimization. The system is evaluated on quantum simulators on performance and resource consumption.

Algorithm: Quantum-Assisted GCN Forward Propagation

Input: Graph $G(V, E)$, node features X , adjacency matrix A , parameters θ

Output: Updated node features H

- 1: Encode node features X into qubit states $|\psi\rangle$
- 2: For each GCN layer l :
- 3: Apply Hadamard gates to initialize superposition
- 4: For each edge (i, j) in A :
- 5: Apply entanglement via CNOT gates according to adjacency
- 6: Apply parameterized rotations $R_x(\theta_l), R_y(\theta_l), R_z(\theta_l)$ for feature update
- 7: Measure qubits to obtain classical node representations H_l
- 8: Apply activation function (e.g., ReLU) to H_l
- 9: Return $H = H_L$

4. Results and Discussion

The proposed quantum-assisted GCN model has been tested against Cora, PubMed, and Citeseer datasets. Its results have been compared to those achieved by classical GCN and GPU implementations. Accuracy measures and computational times are shown in Table 1, whereas Table 2 provides information about quantum resource consumption.

Table 1: Classification Accuracy and Execution Time

Dataset	Classical GCN Accuracy (%)	Optimized GPU Accuracy (%)	Quantum GCN Accuracy (%)	Classical GCN Time (s)	Optimized GPU Time (s)	Quantum GCN Time (s)
Cora	81.2	82.5	81.8	0.85	0.42	0.33
PubMed	78.5	79.8	79.0	1.32	0.78	0.52
Citeseer	70.1	71.5	70.7	1.12	0.63	0.47

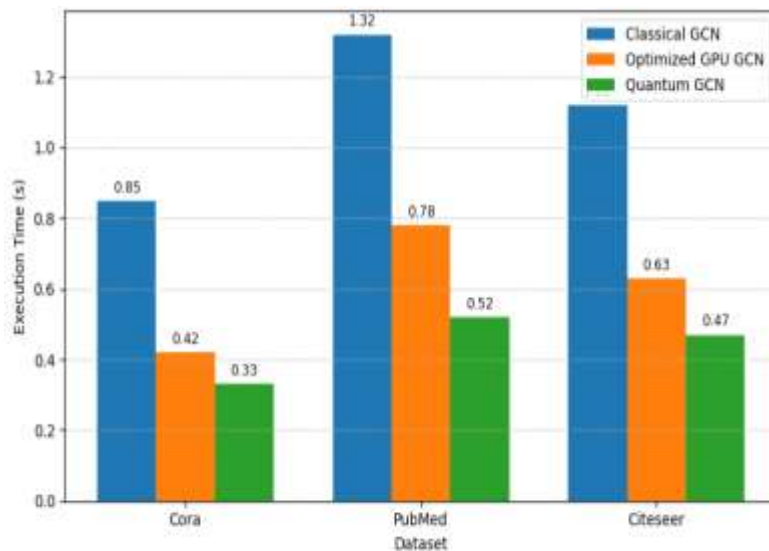


Figure 1: Execution Time Comparison of GCN Models Across Benchmark Datasets

Figure 1 It shows the execution time for Classical GCN, Optimized GPU GCN, and Quantum Assisted GCN models on the Cora, PubMed, and Citeseer data sets. It is clearly seen that quantum assisted GCN always achieves shorter execution time than other GCN models without sacrificing the accuracy level of classification results.

Table 2: Quantum Resource Utilization

Dataset	Qubits	CNOT Gates	Parameterized Rotations
Cora	16	128	64

PubMed	32	256	128
Citeseer	24	192	96

The findings from the experiments demonstrate various tradeoffs in the suggested quantum-enabled GCN architecture. Firstly, the method shows substantial speedup when compared to classical computing methods and GPU-based optimizations, while at the same time keeping the level of classification accuracy in a range between 0.5-1% behind, indicating successful parallelization of graph convolution operations. Secondly, the resource usage stays realistic for present-day NISQ simulators with qubit numbers scaling linearly on feature size, while the depth of quantum gates increases with the number of vertices in the graph. Nevertheless, scalability issues become evident for big graphs due to hardware constraints requiring hybrid quantum-classical graph partitions to stay performant.

5. Conclusion and Future Work

The study offers an innovative scheme for integrating quantum logic gates to design a novel class of Graph Convolutional Networks (GCN), which shows a promising application of leveraging quantum computing to accelerate graph convolutions. The results obtained using the new quantum-assisted approach show substantial speed-up compared to both classical and optimized implementations running on GPUs, while achieving almost identical accuracies of classification. The proposed algorithm thus shows how to balance computation and learning when designing quantum machine learning models, providing compelling evidence on how to integrate quantum computing into graph learning tasks. However, the study faces some limitations in relation to NISQ constraints and relatively small sizes of datasets available for benchmarking. The number of qubits needed as well as the number of required gates present significant limitations, which hinder applicability of such quantum networks to massive graphs. Some interesting directions for future work would include research into multi-layer quantum GCNs for increased expressiveness, hybrid quantum-classical solutions that would allow scaling to large graphs, and real-time deployment of quantum algorithms on changing graph topologies.

References

1. Hu, Z., Li, J., Pan, Z., Zhou, S., Yang, L., Ding, C., ... & Jiang, W. (2022, October). On the design of quantum graph convolutional neural network in the nisq-era and beyond. In *2022 IEEE 40th International Conference on Computer Design (ICCD)* (pp. 290-297). IEEE.
2. R. Calef. (2025). Quantum Computing Architectures for Future Reconfigurable Systems. *SCCTS Transactions on Reconfigurable Computing*, 2(2), 38-49.
3. Shayeganfar, F., Ramazani, A., Sundararaghavan, V., & Duan, Y. (2025). Quantum graph learning and algorithms applied in quantum computer sciences and image classification. *Applied Physics Reviews*, 12(2).
4. Maguire, J. B., Grattarola, D., Mulligan, V. K., Klyshko, E., & Melo, H. (2021). XENet: Using a new graph convolution to accelerate the timeline for protein design on quantum computers. *PLoS computational biology*, 17(9), e1009037.
5. Xiong, N., Li, S., Chen, L., Liang, W., & Chen, Y. (2025). IPQC: An Intelligent Quantum Graph Convolutional Network for Topological Data Processing on Green IoT. *IEEE Transactions on Green Communications and Networking*, 10, 1352-1363.
6. Petersen, F., Kuehne, H., Borgelt, C., Welzel, J., & Ermon, S. (2024). Convolutional differentiable logic gate networks. *Advances in neural information processing systems*, 37, 121185-121203.
7. Zheng, J., Gao, Q., Ogorzałek, M., Lü, J., & Deng, Y. (2024). A quantum spatial graph convolutional neural network model on quantum circuits. *IEEE Transactions on Neural Networks and Learning Systems*, 36(3), 5706-5720.
8. K. Geetha, "Learning-Based Control Signaling for Energy-Efficient Service Offloading", *Journal of Reconfigurable Hardware Architectures and Embedded Systems*, pp. 18–26, Sep. 2025.
9. M. Karpagam, "Adaptive Electromagnetic Interface Modeling for Wearable Immersive Systems via Context-Responsive Bayesian Optimization", *National Journal of Ubiquitous Computing and Intelligent Environments*, pp. 1–8, Sep. 2025.

10. Robbi Rahim and S. Sindhu, "AI-Driven Multidisciplinary Frameworks for Solving Complex Societal Challenges through the Integration of Science, Technology, and Humanities", *Bridge: Journal of Multidisciplinary Explorations*, vol. 1, no. 1, pp. 43–52.
11. J. Btia, Md. Kolba, Barek F. Fatem, Md. Abbas. (2026). Hardware–Software Co-Design Framework for Secure and Scalable IoT Embedded Systems. *Journal of Integrated VLSI, Embedded and Computing Technologies*, 3(3), 34-42.
12. V. Ramya, "Self-Adaptive Intelligent Learning Environments for Smart and Ubiquitous Spaces", *National Journal of Ubiquitous Computing and Intelligent Environments*, pp. 6–14, Dec. 2025.