Unified Relational GNN Architecture for AC Optimal Power Flow Calculations

Ali Trigui
Electric Power Grid Solutions
Qubit Engineering Inc.
Knoxville, TN
trigui@qubitengineering.com

Mohammed Olama
Computational Sciences and Engineering Division
Oak Ridge National Laboratory
Oak Ridge, TN
olamahussemm@ornl.gov

George Siopsis
Quantum Optimization Solutions
Qubit Engineering Inc.
Knoxville, TN
george@qubitengineering.com

Hatem Eldakhakhni
Electric Power Grid Solutions
Qubit Engineering Inc.
Knoxville, TN
hatem@qubitengineering.com

Marouane Salhi
Quantum Optimization Solutions
Qubit Engineering Inc.
Knoxville, TN
salhi@qubitengineering.com

***Abstract*—Solving the AC optimal power flow (OPF) problem remains a computationally intensive task, particularly for large-scale power networks where real-time decision-making is critical. Traditional numerical solvers, while precise, are often too slow for deployment in dynamic environments. Recent data-driven approaches, including fully connected neural networks and standard graph neural networks (GNNs), have been introduced to accelerate AC OPF predictions. However, these models typically treat the power grid as a homogeneous graph, overlooking the rich diversity in node and edge types, such as generators, loads, and different transmission elements, which carry distinct semantic roles. This limits their ability to generalize across varying grid topologies and reduces robustness under structural perturbations. To bridge this gap, we propose a unified framework that extends conventional GNN architectures into their relational counterparts. By allocating separate parameters for each edge type, the relational GNN explicitly models the heterogeneous structure of power networks. This relational modeling enables more accurate representation of physical interactions within the network, leading to improved prediction performance and stronger resilience to topology variations. Numerical results on the IEEE 118-bus test power system demonstrate that the proposed framework not only enhances accuracy in AC OPF predictions but also sets a scalable path forward for future grid-aware deep learning solutions.**

***Index Terms*—Optimal power flow, graph neural network, graph convolution network, graph attention network, graph isomorphism network, relational graph neural network.**

I. Introduction

Solving AC optimal power flow (OPF) problems is a critical task in power system operations and planning, addressing both technical constraints and economic efficiency for electrical utilities. Over the years, a variety of approaches have been developed to tackle this challenge, ranging from linear and nonlinear optimization techniques to heuristic and metaheuristic algorithms [1]–[5]. Although these classical methods have proven effective in many contexts, they often suffer from high computational complexity, making them impractical for real-time use and instantaneous decision-making, especially for large-scale power grids.

In an effort to accelerate AC OPF computations, deep neural networks (DNNs) have been explored as data-driven surrogates for traditional solvers [6], [7]. Although DNNs can offer rapid inference once trained, they typically struggle with scalability across different grid topologies and sizes because of their limited ability to encode structural and relational information inherent in power networks. This shortcoming has motivated a shift toward graph-based learning approaches.

The intuition of modeling power grids as graphs capable of naturally modeling the topological structure of electrical networks and building self-supervised graph networks to imitate complex algorithms [8] emerged early as a solution to solve AC OPF problems. Graph neural networks (GNNs) [9]–[12] followed as scalable and easy to train supervised models that are well suited for AC OPF predictions. However, early applications of GNNs in this domain often relied on homogeneous graph representations, which treat all nodes and edges uniformly. This simplification limits the ability of the model to capture the true complexity of power systems comprising multiple types of entities and various interactions.

To bridge this gap, recent research [13] has begun exploring heterogeneous or relational graph representations, in which node and edge types are explicitly modeled to capture the semantic structure of power grids. Building on this foundation, our work presents a generalized framework that extends several well-known GNN architectures into their relational counterparts. We benchmark these relational GNNs on a suite of AC OPF prediction tasks under realistic power grid sce-

This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (https://www.energy.gov/doe-public-access-plan).

979-8-3315-8138-1

narios. The results demonstrate that relational learning more effectively captures complex interactions within heterogeneous power networks, leading to enhanced predictive performance and scalability compared to traditional homogeneous GNNs.

The remainder of this paper is organized as follows. Section II details the relational variants of GNNs, motivated by the heterogeneous structure of power grids, and presents the proposed generalized relational framework. Section III details the empirical study, including model training, evaluation setup, and experimental results benchmarking the proposed models in multiple AC OPF scenarios, followed by a discussion of performance trends. Finally, Section IV concludes the paper and outlines directions for future research.

II. Relational GNNs for Heterogeneous Power Grid Modeling

A. Heterogeneous Topologies in Power Grids

Smart power grids are inherently heterogeneous, composed of various types of interconnected components such as buses, generators, loads, and transmission lines. Modeling these systems with homogeneous graphs oversimplifies their structure and neglects the rich diversity and semantics of the entities and their interactions.

A more representative approach involves explicitly distinguishing between the different types of nodes and edges that are constructing a heterogeneous graph that captures the structural and functional asymmetry of the grid. This enriched graph formulation enables the learning model to extract more meaningful patterns by focusing on entity and relation types during message passing. However, the main challenge lies in adapting traditional GNNs, originally developed for homogeneous graphs, to effectively handle this semantic richness and complex relational structure.

Here, we present different GNN architectures that will build the proposed variants tailored specifically for AC OPF predictive tasks. Various architectures were evaluated, and the most prominent—those demonstrating the highest performance on benchmark tests—were selected. Each model offers a unique inductive bias and aggregation mechanism.

B. Relational Graph Convolutional Networks

Relational graph convolutional networks (R-GCNs) [14] address the heterogeneity challenge by introducing relation-specific transformations during message aggregation. For each relation type r, a separate weight matrix is used to project neighbor features, thus encoding the distinct semantics of each edge type. The node update rule in R-GCNs is defined as [14]:

$$\mathbf{h}_i^{(l+1)} = \sigma\left(\sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{N}_i^r}\frac{1}{c_{i,r}}\mathbf{W}_r^{(l)}\mathbf{h}_j^{(l)} + \mathbf{W}_0^{(l)}\mathbf{h}_i^{(l)}\right) \tag{1}$$

where $\mathcal{N}_i^r$ denotes the set of neighboring nodes of i under relation r, $\mathbf{W}_r^{(l)}$ is the relation-specific weight matrix, and $c_{i,r}$ is a normalization constant, typically set to $|\mathcal{N}_i^r|$. The self-loop term $\mathbf{W}_0^{(l)}\mathbf{h}_i^{(l)}$ ensures that the node's own features are retained during the update.

C. Relational Graph Attention Networks

Relational graph attention networks (R-GATs) [15] enhance the R-GCN structure by incorporating the adaptive learning of GAT using the attention mechanism that learns the importance of each neighbor per relation type instead of treating all equally. This results in a finer-grained and relation-aware neighbor aggregation.

The attention score between node i and a neighbor j under relation r is computed as [15]:

$$e_{ij}^r = \text{LeakyReLU}\left(\mathbf{a}_r^\top\left[\mathbf{W}_r\mathbf{h}_i \,\|\, \mathbf{W}_r\mathbf{h}_j\right]\right) \tag{2}$$

where $\mathbf{a}_r$ is a learnable attention vector for the relation r, $\|$ denotes concatenation, and $\mathbf{W}_r$ is the shared linear transformation for that relation. These scores are normalized using a softmax function as [15]:

$$\alpha_{ij}^r = \frac{\exp(e_{ij}^r)}{\sum_{k\in\mathcal{N}_i^r}\exp(e_{ik}^r)} \tag{3}$$

The final node representation is obtained by aggregating the weighted neighbor features across all relation types as [15]:

$$h_i^{(l+1)} = \sigma\left(\sum_{r\in\mathcal{R}}\sum_{j\in\mathcal{N}_i^r}\alpha_{ij}^r W_r^{(l)} h_j^{(l)}\right) \tag{4}$$

where $h_i^{(l)}$ and $h_j^{(l)}$ represent the feature vectors of node i and its neighbor j at layer l, respectively. The updated node representation $h_i^{(l+1)}$ is obtained by a weighted sum over its neighbors' transformed features. Here, $W_r^{(l)}$ is the relation-specific weight matrix, and α_{ij}^r is the attention coefficient for neighbor j under relation r.

D. Unified Relational GNNs

Inspired by the effectiveness of R-GCNs and R-GATs in capturing semantic diversity, we extend this relational message passing paradigm to other powerful GNN architectures. Fig. 1 provides a visual overview of the generalized relational GNN architecture that makes it easier to integrate many other relational variants. Designing a new relational GNN essentially involves choosing a specific homogeneous GNN variant as the backbone of the framework and adapting its learned weights to handle each relation type independently. This allows the model to capture distinct relational semantics and later aggregate them into a unified, relation-aware latent representation for each node that reflects structural diversity. The modular nature of this approach facilitates flexible component selection, interpretability, and seamless adaptation of different GNN architectures to heterogeneous graph settings.

In this study, we introduce relational variants of GraphSAGE [16] and graph isomorphism network (GIN) [17], referred to as R-GraphSAGE and R-GIN, respectively. These models retain the structural foundations of their base counterparts, but they are adapted to process and learn from heterogeneous relations.

979-8-3315-8138-1

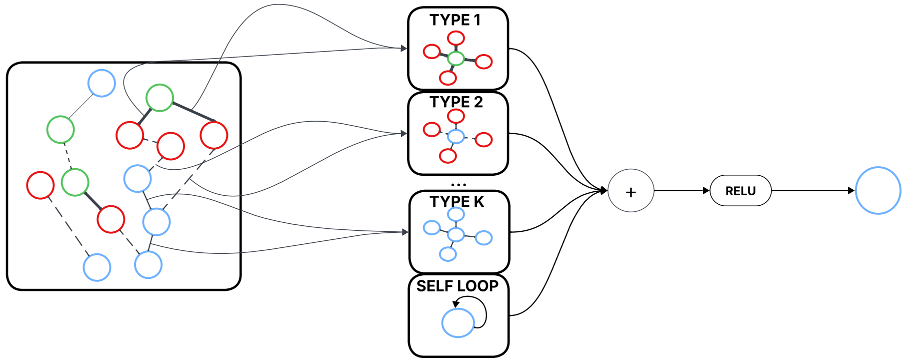


Fig. 1: General relational GNN structure. For each relation type, the message passing is computed separately, and then all relation types are aggregated altogether to compute the final node representation.

- In R-GraphSAGE, neighbor sampling and aggregation are performed per relation, with each type having its own transformation.
- In R-GIN, separate multilayer perceptrons (MLPs) and parameters are used to reflect the semantics of the relation types in injective aggregation.

Table I details the different aspects of the selected methods. The proposed framework allows us to systematically benchmark different relational GNNs in the context of smart power grids. We believe that power grids provide the proper ground to test the capability of these models considering the semanticity of relations between its entities and how different relations can express extremely different inherent meanings, making a relational framework the most suitable.

III. Empirical Study

To evaluate and compare the performance of various relational GNN architectures, we conducted a series of controlled experiments on multiple scenario types derived from public AC OPF datasets. Each type of scenario simulates different operating conditions in power networks, including variations in load, generation, and topology.

A. Graph Representation and Feature Encoding

Each power grid is modeled as a heterogeneous graph $G = (V, E)$, where nodes V represent electrical entities, such as generators, loads, and slack buses, and edges E correspond to physical components, such as transmission lines or transformers. The node types are encoded explicitly, enabling the models to differentiate among functional roles in the network.

The input features for each node include power injection values (active P and reactive Q, when applicable), and voltage magnitudes and phase angles from a flat start or previous scenario state. The features of the generator nodes also include the generator capacity and setpoint information, while the load nodes carry demand forecasts. These features are normalized and used as input to relational GNN models, which employ type-aware message-passing mechanisms to propagate information across the grid structure.

B. Loss Function and Prediction Targets

All models are trained to jointly predict voltage angle (VA) and voltage magnitude (VM) in each bus. The loss is computed by comparing the predicted values with the ground truth values from the power flow simulations. To ensure that the models learn concise and meaningful representations, the loss function integrates both bus-level and generator-level predictions:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{bus}} + \mathcal{L}_{\text{gen}} \tag{5}$$

where $\mathcal{L}_{\text{bus}}$ is the mean squared error between the predicted and actual voltage magnitudes and angles on each bus, and $\mathcal{L}_{\text{gen}}$ captures the error in the active and reactive power predicted at the generator nodes. This dual-target setup ensures that the model not only fits the global voltage profile but also pays special attention to generator output, which plays a critical role in system balance and control.

C. Experimental Protocol

We train and test the proposed variants of GNN models using 75 thousand scenarios of the standard IEEE 118-bus test power grid system in [18]. We train the various models using consistent hyperparameter, on 90% of the scenarios grouped into batches. The training is conducted over 50 epochs for all the models to ensure a fair comparison across architectures.

We evaluate the relational variant of each of the four most common GNNs as well as the normal homogeneous GCN on the remaining 10% of the case scenarios, both in standard configuration and topologically perturbed networks that reflect real-world changes such as line failures or network reconfigurations. The evaluation includes errors on the predicted VA and VM for each scenario type. Table II summarizes the test performance, reported as Mean Squared Error (MSE), for each model in all scenario types. The best-performing results are highlighted in bold.

D. Evaluation Results

The quantitative results presented in Table II demonstrate that all relational models consistently outperform the homogeneous baseline in all evaluation metrics. This performance

979-8-3315-8138-1

TABLE I: Particular Differences between Selected Methods

Model	Aggregator Type	Relation Parameters	Permutation-Invariant	Expressiveness/Cost Trade-Off
R-GCN	Normalized linear sum	One weight matrix W_r per relation and a self loop weight matrix W_0	Yes	Simple and fast fewer parameters but less expressive than GAT and GIN.
R-GAT	Attention-weighted sum	One weight matrix W_r and attention vector a_r per relation (often multi-head)	Yes	More expressive (learns relative importance per neighbor) although slower (attention overhead).
R-GraphSAGE	Pooling (mean/max/LSTM)	One weight matrix W_r and potential pooling parameters per relation	Yes (if using mean/max; LSTM variant breaks strict invariance but is made invariant by shuffling inputs)	Flexible: can pick a cheap mean, mid cost max-pool, or heavier LSTM. Mid range expressiveness. Addictional trade-off depending on the number of neighbors to aggregate from.
R-GIN	Sum + MLP	One MLP (deep, multi-layer) per relation	Yes	Highest expressive power (injective sum + MLP), but MLPs per relation increase parameters.

TABLE II: Comparison of Various GNN Models on AC OPF Prediction: Voltage Magnitude (VM) and Voltage Phase Angle (VA) Mean Squared Errors (MSE) under Normal and Perturbed Topologies

Model	VA Error (Normal)	VM Error (Normal)	VA Error (Perturbed)	VM Error (Perturbed)	Training Time (s/epoch)	Parameters (×10³)
Homogeneous GCN	$1.2e^{-3}$	$9.6e^{-5}$	$3.5e^{-3}$	$1.7e^{-4}$	117.60	1.5
Relational (Heterogeneous) GNNs						
R-GCN	$6.4e^{-4}$	$3.7e^{-5}$	$2.7e^{-3}$	$\mathbf{7.4e^{-5}}$	**285.86**	**13**
R-GAT	$6e^{-4}$	$\mathbf{3.5e^{-5}}$	$2.4e^{-3}$	$8.3e^{-5}$	557.02	14
R-GraphSAGE	$7.9e^{-4}$	$7.3e^{-5}$	$2.73e^{-3}$	$1.24e^{-4}$	332.66	13.5
R-GIN	$\mathbf{5.7e^{-4}}$	$5.6e^{-5}$	$\mathbf{2.2e^{-3}}$	$9.3e^{-5}$	368.04	20

gain can be attributed to their abilities to distinguish between different types of relationship within the power grid, unlike homogeneous models that operate under a single-type edge representation and thus fail to capture relational diversity.

Among the relational models, R-GIN achieves the strongest performance on VA prediction, consistent with the fact that VA depends on global power balance and long-range phase coupling; R-GIN's injective aggregation is well suited to capturing these broad structural dependencies. In contrast, R-GAT performs best on VM, which is driven mainly by local power injections and immediate electrical connections—interactions that attention mechanisms model effectively by prioritizing the most influential neighbors. R-GCN shows the highest robustness under perturbed topologies, as its relation-specific transformations preserve the physical meaning of different edge types, allowing the model to generalize reliably when the network structure changes. Meanwhile, R-GraphSAGE underperforms across all metrics: its mean-based aggregation tends to oversmooth features, and its neighborhood sampling strategy is inefficient in power networks where nodes typically have only one neighbor of each relation type, leaving little benefit to GraphSAGE's sampling-based design and limiting its expressiveness in this domain.

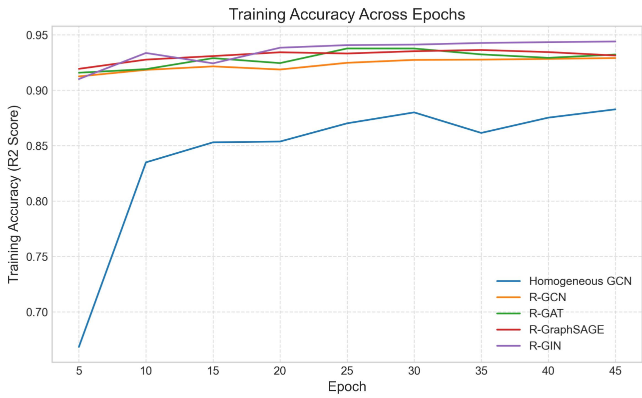
Fig. 2 shows a comparison of training dynamics of the various GNN models. As illustrated in Fig. 2a, relational models exhibit faster convergence during training as compared to the homogeneous baseline model. Despite the additional parameters introduced by relation-specific processing, they achieve optimal performance in fewer epochs. This suggests that relational GNNs are more effective in capturing the semantic structure of the grid early in the training process, leading to more efficient learning and ultimately improved predictive performance. Fig. 2b shows the training loss curves of the various relational GNN models. It is observed that all relational models show similar behavior with a slightly more remarkable convergence for R-GIN.

As reported in Table II, while relational models achieve higher predictive performance, they do so at the expense of computational efficiency. They exhibit substantially larger parameter counts and require roughly three to five times longer training times as compared to the homogeneous model. This trade-off is expected, as relational architectures introduce additional learnable parameters through their relation-specific weight parametrization.

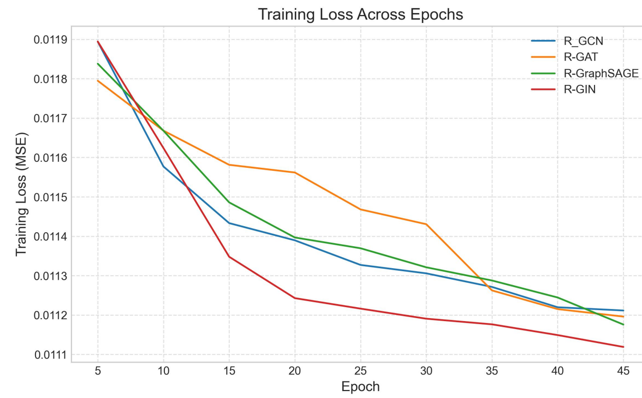
IV. Conclusion and Future Work

This work introduced a unified framework for extending various GNN architectures into their relational counterparts, making them more suitable for modeling the heterogeneous structure of power grids. A comparative evaluation showed that the models instantiated within this framework consistently outperform baseline homogeneous GNNs in both standard and contingency grid configurations. Furthermore, relational

979-8-3315-8138-1



(a) Training accuracy of the various GNN models throughout training: Different relational models converge faster and show better performance than homogeneous GNN. Accuracy here refers to the R2 score calculated from the predicted voltage magnitude and angle.



(b) Training loss curves of the various relational GNN models: All relational models show similar behavior with a slightly more remarkable convergence for R-GIN. The calculated loss here refers to the total loss function in (5).

Fig. 2: Comparison of training dynamics of the various GNN models.

models exhibit faster and more stable convergence during training, highlighting their improved learning efficiency.

Although our experiments focus on the IEEE 118-bus case, we believe that the modular relational design can extend to larger systems; future work will test the scalability on grids with thousands of buses. The flexibility of the proposed framework facilitates the exploration of additional GNN backbones within the context of power systems, and it supports the integration of diverse prediction targets for practical, real-world scenarios where minimizing forecast errors is critical. We believe that this work lays the foundation for future research that can extend the relational paradigm to capture higher-order, global graph interactions to achieve more expressive and scalable representations for power systems' analytics.

Acknowledgment

This work is supported by a Cooperative Research and Development Agreement (CRADA) through the Oak Ridge National Laboratory (ORNL) Innovation Crossroads Program.

References

[1] D. I. Sun, B. Ashley, B. Brewer, A. Hughes, and W. F. Tinney, "Optimal power flow by Newton approach," *IEEE Transactions on Power Apparatus and Systems*, vol. 103, no. 10, pp. 2864–2880, 1984.

[2] K. Y. Lee, Y. M. Park, and J. L. Ortiz, "A united approach to optimal real and reactive power dispatch," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-104, no. 5, pp. 1147–1153, 1985.

[3] O. Alsac, J. Bright, M. Prais, and B. Stott, "Further developments in LP-based optimal power flow," *IEEE Transactions on Power Systems*, vol. 5, no. 3, pp. 697–711, 1990.

[4] F. Zhou, J. Anderson, and S. H. Low, "The optimal power flow operator: Theory and computation," *IEEE Transactions on Control of Network Systems*, vol. 8, no. 2, pp. 1010–1022, 2021.

[5] S. Gupta, M. Bilal, M. Zuhaib, L. Srivastava, H. Malik, F. P. García Márquez, and A. Afthanorhan, "Optimal power flow solution using novel optimization technique: A case study," *Expert Systems with Applications*, vol. 287, p. 128163, 2025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S095741742501783X

[6] T. Zhao, X. Pan, M. Chen, A. Venzke, and S. H. Low, "DeepOPF+: A deep neural network approach for DC optimal power flow for ensuring feasibility," in *2020 IEEE Int. Conf. on Communications, Control, and Computing Tech. for Smart Grids (SmartGridComm)*, 2020, pp. 1–6.

[7] R. Zafar, B. H. Vu, and I.-Y. Chung, "A deep neural network-based optimal power flow approach for identifying network congestion and renewable energy generation curtailment," *IEEE Access*, vol. 10, pp. 95 647–95 657, 2022.

[8] B. Donon, R. Clément, B. Donnot, A. Marot, I. Guyon, and M. Schoenauer, "Neural networks for power flow: Graph neural solver," *Electric Power Systems Research*, vol. 189, p. 106547, 2020.

[9] D. Owerko, F. Gama, and A. Ribeiro, "Unsupervised optimal power flow using graph neural networks," in *IEEE Int. Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 6885–6889.

[10] S. Liu, C. Wu, and H. Zhu, "Topology-aware graph neural networks for learning feasible and adaptive AC-OPF solutions," *IEEE Transactions on Power Systems*, vol. 38, no. 6, pp. 5660–5670, 2023.

[11] T. Pham and X. Li, "Reduced optimal power flow using graph neural network," in *North American Power Symposium (NAPS)*, Oct. 2022.

[12] A. Deihim, D. Apostolopoulou, and E. Alonso, "Initial estimate of AC optimal power flow with graph neural networks," *Electric Power Systems Research*, vol. 234, p. 110782, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378779624006680

[13] S. Ghamizi, A. Ma, J. Cao, and P. Rodriguez Cortes, "OPF-HGNN: Generalizable heterogeneous graph neural networks for AC optimal power flow," in *2024 IEEE PES General Meeting*, 2024, pp. 1–5.

[14] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. van den Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *The Semantic Web*, A. Gangemi, R. Navigli, M.-E. Vidal, P. Hitzler, R. Troncy, L. Hollink, A. Tordai, and M. Alam, Eds. Cham: Springer International Publishing, 2018, pp. 593–607.

[15] K. Wang, W. Shen, Y. Yang, X. Quan, and R. Wang, "Relational graph attention network for aspect-based sentiment analysis," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds. Online: Association for Computational Linguistics, Jul. 2020, pp. 3229–3238. [Online]. Available: https://aclanthology.org/2020.acl-main.295/

[16] W. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," *arXiv.1706.0221*, 06 2017.

[17] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=ryGs6iA5Km

[18] S. Lovett, M. Zgubic, S. Liguori, S. Madjiheurem, H. Tomlinson, S. Elster, C. Apps, S. Witherspoon, and L. Piloto, "OPFData: Large-scale datasets for AC optimal power flow with topological perturbations," *arXiv.2406.07234*, 06 2024.