Review of *Modeling Relational Data with Graph Convolutional Networks*

Samar Abdelghani (2415295)

Summary

This paper introduces the **Relational Graph Convolutional Network (R-GCN)**, an extension of standard Graph Convolutional Networks (GCNs) designed to handle multi-relational graphs. The paper tackles real-world datasets where entities are connected by multiple types of edges (relations), and **traditional GCNs cannot distinguish between these relation types**. Therefore, the authors propose a message-passing mechanism in which each relation type has its own transformation matrix. The model addresses parameter explosion through two regularization strategies: basis decomposition (sharing weights across relations) and block-diagonal decomposition (restricting transformations to smaller subspaces).

R-GCNs are applied to two tasks:

- **Entity classification:** predicting labels of nodes (entities) in relational graphs.
- **Link prediction:** predicting missing relationships in a knowledge graph using a decoder such as DistMult which is considered as a scoring function.

Experiments conducted on benchmark datasets(AIFB, MUTAG, BGS, and AM for the entity classification task, and FB15k, FB15k-237, and WN18 for the link prediction task) demonstrate that R-GCN achieves competitive performance compared to baseline models including RDF2vec, WL, and Feat in entity classification and DistMult, CP, HolE, ComplEx, TransE, and LinkFeat in link prediction. On the FB15k-237 dataset (where inverse relations have been removed), R-GCN attains a 29.8% improvement over previous methods, highlighting its capability in modeling multi-relational structures.

Reasons to accept the paper

1. **Novelty**: The paper presents **R-GCN** an innovative adaptation of Graph Convolutional Networks to relational and heterogeneous graphs, addressing the limitation of traditional GCNs that operate only on homogeneous structures. R-GCN is well easily represented in equation(2).
2. **Comprehensive Evaluation**: The experiments in tables 2, 4, 5, and 7 demonstrate consistent improvements across multiple knowledge graphs of different sizes and tasks, confirming the robustness of the proposed approach.
3. **Clarity and Presentation**: The paper includes clear visualizations(pdf figures with high resolutions) and is well-organized, easy to read, and understandable, making the methodology and results accessible to a wide range of readers.
4. **Availability of Code**: The implementation is publicly available, and its results are fully reproducible as I have successfully reproduced the experiments myself.

Reasons to reject the paper

- The datasets used in the experiments are relatively small in scale, containing up to approximately six million edges. In contrast, real-world knowledge graphs contain billions of edges, such as the **Microsoft Academic Graph (MAG)** with about 13 billion edges and **Wikidata** with around 3 billion edges.

- The proposed method does not employ any sampling or sparsification techniques, making it **non-scalable** to large-scale knowledge graphs. Training on such massive graphs would require an excessive amount of computational resources.

- The paper does not address or discuss the **over-smoothing phenomenon** in Graph Neural Networks (GNNs), where node embeddings become increasingly similar across layers, resulting in less accurate predictions. This issue typically arises when more than two aggregation layers are used. The paper uses only two layers though all experiments.

- The authors do not provide any **theoretical proof of convergence** during training, leaving the model's stability and convergence behavior unverified.

- The paper lacks an **ablation study** analyzing the effects of aggregation depth, graph density, and graph structure on model convergence and performance.

Recommendation

My recommendation is that the paper should be accepted(Strong accept), this is because the approach is novel and bridges the gap between GCN frameworks and knowledge graph embeddings, offering a foundation that many other works build upon.

Major Comments

DO excessive experiments which I mentioned in a previous section(reasons to reject):

1. Conduct experiments on real-world knowledge graphs contain billions of edges, such as MAG and Wikidata.

2. Study the effect of sampling techniques on large-scale knowledge graphs.

3. Study the over-smoothing phenomenon of GNNs by testing a number of layers greater than two.

4. Provide theoretical proof of convergence during training.

5. Provide an ablation study for analyzing the effects of aggregation depth, graph density, and graph structure on model convergence and performance.

6. Provide computational resource details (e.g., hardware specifications, runtime, memory usage) to assess scalability more concretely.

Minor comments

1. In the abstract: The phrase "a large improvement of 29.8% on FB15k-237" is slightly ambiguous. It would be clearer to specify the metric, e.g., "a large improvement of 29.8% in MRR on FB15k-237.

2. Missing comma after "i.e." in all pages.

3. The "adaption" should be replaced by "adaptation" in Page 7.

4. In section 5.2, add more background about the used metrics MRR and the Hits at n(H@n).