Graph2Seq Implementation Report

Anwar Said

January 2023

1 Introduction

This report presents the detail of Graph2Seq [5] implementation in PyTorch. Graph2Seq is an end-to-end deep learning framework that uses both Graph Neural Networks (GNNs) and Recurrent Neural Networks (RNNs) to perform Sequence to Sequence (Seq2Seq) learning. The key advantage of this framework is considering the input as a graph which is a natural and powerful representation for many real-world systems. Taking the sequences as input in the form of a graph, Graph2Seq first uses GNNs to learn node and graph representations, and then uses RNNs with attention mechanism to decode the representations obtained through GNNs. The model has been evaluated on three tasks: natural language generation, finding shortest paths and BABAI TASK 19 and showed encouraging results. The specifics of the implementation are provided in the next sections.

2 Problem Definition

This assignment focuses on the natural language generation task. Specifically, given a graph-representation of a set of SQL queries, the objective is to generate a natural language description of their meaning. The WikiSQL dataset was intended for this purpose. However, due to the absence of graphical representations of WikiSQL, I opted to use the Shortest Path synthetic dataset created by the authors in the same paper.

3 Experimental Results

In this section, I present some preliminary results obtained using an SP-L dataset generated synthetically. This dataset was constructed by the authors (see section 4.2 of the main paper [5]), which contains 3000 graphs with 100 nodes each. The labels consist of a START token, node labels that are part of the shortest path, and an END token. The objective is to identify the shortest directed path between two nodes by predicting the sequence.

Model	Accuracy
Baseline	99.3
GCN	100
SAGE	99.97
ResGatedConv	99.97
GraphConv	99.97

Table 1: Comparison of different GNNs and baseline on SP-L datasets

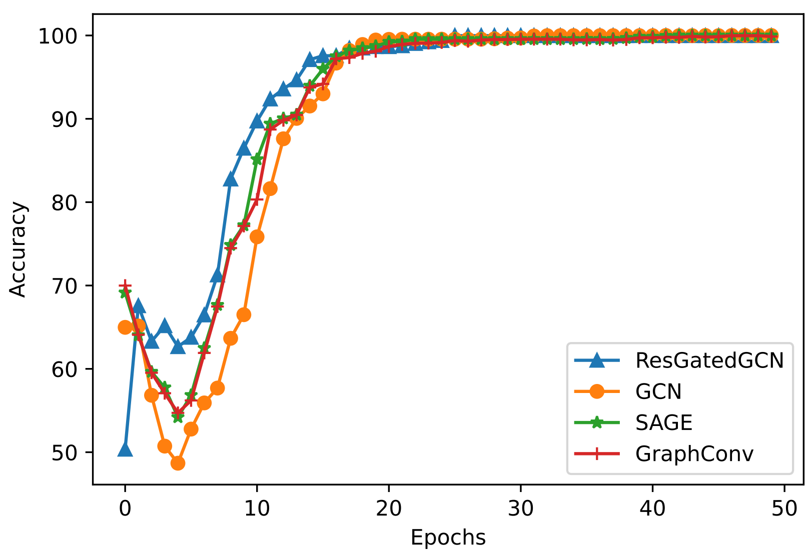

Figure 1: Testing accuracy

Baseline: Here I consider Graph2Seq results presented in Table 1 (SP-L) of the original paper as a baseline for evaluation. In comparison, I consider four well-known GNNs models: Graph Convolutional Network (GCN) [3], GraphSAGE [2], Residual Gated Graph Convnet [1] and Graph Convolution [4]. The experimental setup with hyper-parameters is the same as mentioned in the paper. The results comparison in terms of accuracy is presented in Table 3. We also provide the test accuracy and loss curves of these models in Figures 1 and 2.

References

[1] Bresson, X., and Laurent, T. Residual gated graph convnets. *arXiv preprint arXiv:1711.07553* (2017).

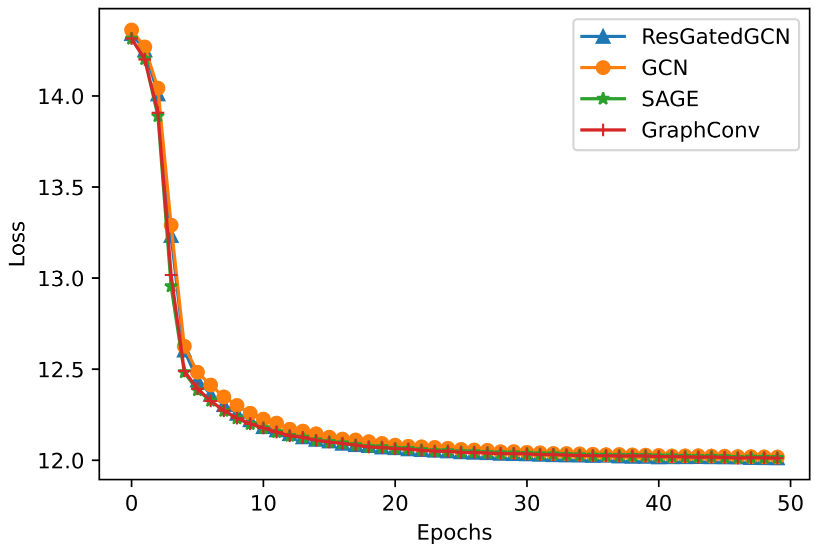

Figure 2: train Loss

[2] Hamilton, W., Ying, Z., and Leskovec, J. Inductive representation learning on large graphs. *Advances in neural information processing systems 30* (2017).

[3] Kipf, T. N., and Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[4] Morris, C., Ritzert, M., Fey, M., Hamilton, W. L., Lenssen, J. E., Rattan, G., and Grohe, M. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence* (2019), vol. 33, pp. 4602–4609.

[5] Xu, K., Wu, L., Wang, Z., Feng, Y., Witbrock, M., and Sheinin, V. Graph2seq: Graph to sequence learning with attention-based neural networks. *arXiv preprint arXiv:1804.00823* (2018).