Project Proposal for Semi-Supervised Classification with Graph Convolutional Networks

Zhisheng Xu and Goutham Pakalapati
{zx41, gpakal2}@illinois.edu

Group ID: 28
Paper ID: 11

1 Problem

The Semi-Supervised Classification with Graph Convolutional Networks (Kipf and Welling, 2017) aims to solve the problem of semi-supervised classification on graph-structured data. The objective is to:

- develop a scalable and efficient method that can accurately classify nodes or predict labels for unseen data points in the graph which has limited amount of labeled data;
- apply concepts of convolutional neural networks, such as when filtering with images, towards graphs.

2 Approach

The paper presents several key contributions.

First, the authors provide a method of propagating over layers of the neural network model on graph structures. They also show that the Graph Convolutional Network (GCN) models can be used to accurately and efficiently label graph nodes in a classification task, such as finding potential dating candidates.

Second, the authors propose a new layer that performs graph convolutions by leveraging a localized first-order approximation of spectral graph convolutions as the basis of the model architecture. This simplification reduces the computational complexity and allows for direct convolution operations on graph data, making the model more scalable and efficient.

Third, graph edges should not be considered solely for node similarity. They may also contain additional valuable information. To avoid explicit graph-based regularization in the loss function, the authors train the model directly on all nodes with labels.

3 Hypotheses

Our aim is to replicate the process detailed in the paper and reproduce the results presented by the authors using the same data. Furthermore, we want to demonstrate that the new GCN method is more accurate and efficient compared to the other methods covered in the paper. Finally, we seek to verify that the complexity of GCN is indeed linear.

4 Ablations

We want to explore a few potential ideas to experiment with to see which are the most feasible and effective. First, we would like to try to modify the propagation model. As discussed in the original paper, the authors utilized ReLU as the activation function. We would like to experiment with alternative methods, such as Sigmoid or Tanh, to observe any variations in the results. Additionally, we would like to understand the memory and time complexity of training the model by messing with the number of edges of the input graph.

5 Access to the Data

The GitHub repository [1] corresponding to the paper has a data folder that should contain all the original data that we will be using to produce results in the reproduction of this paper.

6 Computational Feasibility

Compared to the 16-core Xeon CPU E5-2640 v3 @ 2.60GHz and GeForce GTX Titan X used by the authors of original paper, we will be running on an AMD Ryzen 3700 processor with 32GB of RAM and a GeForce RTX 3080 Ti graphics card. We think we will have similar performance in terms of GPU and CPU, but the original paper does say that the process may be RAM-intensive without listing

[1] https://github.com/tkipf/gcn/tree/master/gcn/data

their RAM specifications. As a result, we will need to determine if the RAM we have is sufficient or if more is required. We believe that our current RAM is likely adequate for the task.

7 Re-use Existing Code

We will attempt to reproduce the methodology ourselves, using the existing code [2] as a reference as needed. The overall implementation will likely be close to the original paper.

References

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. *Proceedings of the International Conference on Learning Representations (ICLR)*.

[2] https://github.com/tkipf/gcn