Multi-View Graph Neural Network: an Attentional Model to Incorporate with Views

Irene Li, Alex Fabbri, Swapnil Hingmire and Dragomir Radev

Yale

LILY Lab

Introduction

Recent years have witnessed the breakthroughs of deep learning in many fields, such as computer vision (CV) and natural language processing (NLP). Successful models, including CNNs, LSTMs and BERT have a strong ability to process images and texts. Apart from those models, another group of deep models is proposed to cope with graphs directly, such as the graph neural network (GNN) and graph convolutional network (GCN).

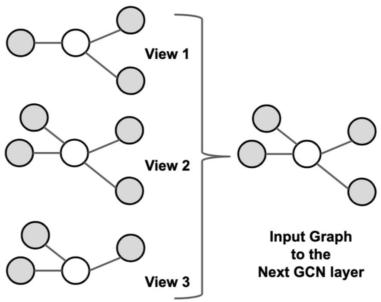
Our contributions can be summarized as the following: 1) we propose the MVA-GCN model (Multi-view Attentional Graph Convolutional Networks) to improve performance on node classification and link prediction by utilizing multiple views of a graph; 2) we test on two public datasets for node classification and achieve good primary results;

Related Work

GCNs have been widely applied for node classification, matrix completion, link prediction for various types of graphs. In the NLP domain, researchers investigated the methods for predicting document labels via inter-relations of words and documents within a graph; other research including learning the prerequisite chains in a concept-graph.

The existing GNN research falls into two main categories, either learning node representation or link prediction. However, limited research has focused on multi-view networks or graphs. In this project, we propose a variation of the GCN model, which takes into account multiple views and computes the node representation with the attention mechanism.

Figure 1. Model Illustration



MV-GNN Equation

$$e_{vij} = score_l(h_i, h_j)$$

$$\alpha_{vij} = \frac{exp(e_{vij})}{\Sigma_{v\in V}\Sigma_{k\in N_i} exp(e_{vik})}$$

$$h_i' = \sigma(\Sigma_{v\in V}\Sigma_{k\in N_i}\alpha_{vik}W_v h_k))$$

$$score_l = LeakyReLU(a_l[W_v h_i; W_v h_j])$$

Table 1. Accuracy of the models.

Method	Cora	Citeseer
Graph Neural Network (Kipf andWelling, 2017)	81.5	70.3
Multi-GCN (Kan et al., 2019)	82.5	71.3
Graph Attention Networks (Veličković et al.,2017)	83.0 ± 0.7	72.5 ± 0.7
Our methods		
MV+GCN	82.24 ± 1.0	~69.9
MV+GAT	**0.844**	~72.0

Dataset	Type	Node	V1 Edge	V2 Edge	V3 Edge	Class	Features	Density
Cora	Citation Network	2,708	5,429	2,846	-	7	1,433	0.052
Citeseer	Citation Network	3,327	4,732	3,492	-	6	3,703	0.036

Figure 2. A detailed statistics of the datasets.

Datasets

For node classification task, we start our method on citation networks such as Cora (2,708 nodes, 5,429 edges) and Citeseer (3,327 nodes, 4,732 edges); Other possible datasets are: actor co-occurrence network (640,134 nodes, 1,554,643 edges), Knowledge Bases such as WebKB (about 1051 classified pages) http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-11/www/wwkb/index.html; The latest Wikipedia network (11,631 nodes, 170,918 edges):https://github.com/benedekrozemberczki/datasets.

For link prediction task, we will try our model on our Lecturebank datasets, which is a topic-resource heterogeneous graph, so as to have a comprehensive analysis on our new graph neural network model for multi-view version.

Conclusion

From our preliminary results, we show the improvements on our proposed model on Cora dataset by introducing texts and deep node embeddings. However, with more views, we will have a larger number of parameters, and thus the overfitting becomes an issue. Besides, graph is a very special case in deep learning. It is very hard to train a large graph in parallel unless distributed systems are being applied.
Future work will be focused on to test our idea on more datasets and a various graph tasks like link prediction and relation inference.

Acknowledgement

Special thanks to my advisor Drago and our new postdoc visitor Swapnil for setting up weekly meetings with me. My sincere gratitude also goes to Alex, who is my best labmate and supports me all the time.