Michael Potter
UID: 705644667
mlpotter124@g.uclda.edu

Project Proposal
CS260: Machine Learning Algorithms

2021-02-05

Goal of Project

The goal of my project is to replicate *Yao et al* [3], as it was a cornerstone paper for Graph Convolutional Networks (GCNs) in transductive text classification. Therefore, I will construct a Word-Document Graph which combines high level document nodes with low level word nodes for predicting document node level predictions (Figure. 1) This project will allow me to explore PyTorch Geometric, and research the intersection of Natural Language Processing (NLP) and Graph Theory.

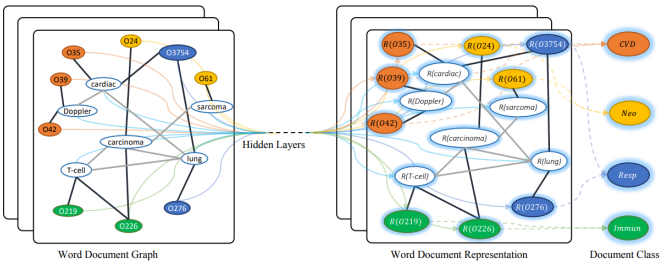

Figure 1: Schematic of Text GCN)

Why is it important?

Traditional deep learning methods like Convolutional Neural Networks (CNNs) and Long Short-Term Memory Recurrent Neural Networks (LSTMS), only focus on local consecutive word sequences, but fail to explicitly use global word co-occurrence information from the corpus. Additionally, document embeddings learned from the aforementioned methods do not infer from other documents in the corpus, which could provide strong supplementary information. State-of-the-art (SOTA) NLP models like Bidirectional Encoder Representations from Transformers (BERT) do not focus on local consecutive word sequences (because of the document-level attention mechanism), but also still fail to explicitly use global word co-occurence information from the corpus and supplementary document information. Therefore, a GCN has two main advantages: 1) the GCN may capture both document-word relations and global word co-occurrence information from the corpus; 2) the GCN model computers new features of a node as the weighted average of itself and second order neighbors comprising words and other documents. Document classification is an important task because it is used in applications such as email spam filtering, document security level classification, medical patient diagnosis, etc.

Potential Algorithms

The project will be using the algorithms proposed by *Yao et al* [3]. There are two critical aspects to the algorithm: 1) construction the input graph to GCN; 2) the GCN model. The document-word edges and the word-word edges are weighted edges calculated by Pointwise Mutual Information (PMI) and Term Frequency - Inverse Document Frequency (TF-IDF) scores (Eq. 1)

$$A = \begin{cases} PMI(i,j) & \text{i,j are words, } \; PMI(i,j) > 0 \\ TF - IDF_{ij} & i \text{ is document, } j \text{ is word} \\ 1 & i = j \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

The GCN loss function will be standard Cross-Entropy loss, and the GCN model will only be two layers (Eq. 2)

$$Z = softmax(\tilde{A}ReLU(\tilde{A}XW_0)W_1) \tag{2}$$

$$\tilde{A} = D^{-\frac{1}{2}}AD^{-\frac{1}{2}} \tag{3}$$

References

The project will be based on *Yao et al* [3], which heavily references *Kipf et al* [2] for comparison purposes. I will also implement BERT [1] to compare against GCN on text classification.

Bibliography

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[2] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks, 2017.

[3] L. Yao, C. Mao, and Y. Luo. Graph convolutional networks for text classification, 2018.