Summary of Graph Convolutional Networks for Text Classification Summary
Liang Yao, Chengsheng Mao, Yuan Luo
AAAI Conference on Artificial Intelligence (2018)

Clément Chauvet * Elise Chin * Mathilde Da Cruz *

Abstract

In this document we will try to provide a brief explanation of the main points of the paper: Graph Convolutional Networks for Text Classification. This paper builds a classification model for text documents that work in semi-supervised cases where we have only a small percentage of our data labelled. The text corpus is transformed into a graph based on word co-occurrences and document word relations, then a two-layer graph convolutional network, called TextGCN, is trained on the graph to classify unlabelled documents.

1. Introduction

This paper aims to solve a very classic task in the NLP field: document classification. But before this, an essential intermediate step for text classification is text representation. Where classic methods try to represent a corpus of documents with a table using statistical information, a corpus is here represented by a graph whose vertices are words or documents and edges are particular links between them. Traditional methods would then use Bayes formula to predict the probability of belonging to one class, or other models such as random forests. Other methods use deep learning, but their quality is often based on the quality of the word embeddings used. Some researches have also shown that Graph neural networks can be good at this task, but the difference with the method explained here is in the graph's construction. After the construction, which is the main contribution of the paper, a very simple Graph Convolutional Network is used to classify the vertices (and thus the texts). The originality of the method (transforming a text classification problem into a node classification problem) is also a contribution. This method can be called semi-supervised, as we are training on the entire set of data, but only a little part is labelled. Another contribution is that the model learns jointly words and document embeddings. The model described in this paper achieves state-of-the-art results for some classical datasets without any use of pre-trained model or external knowledge.

2. Method

2.1. Graph Construction

From the text corpus, the authors build a heterogeneous graph where nodes can represent either a word or a document. Then they add an edge between a word and a document valued with the term frequency inverse document frequency (TF-IDF), and an edge between two words is valued with the Pointwise Mutual Information (PMI). This score is based on the co-occurence of words inside a fixed-size sliding window. The PMI value of a word pair i, j is computed as

$$\text{PMI}(i, j) = \log \frac{p(i, j)}{p(i)p(j)} \tag{1}$$

$$p(i, j) = \frac{\#W(i, j)}{\#W} \tag{2}$$

$$p(i) = \frac{\#W(i)}{\#W} \tag{3}$$

where $\#W(i)$ is the number of sliding windows in a corpus that contain word i, $\#W(i, j)$ is the number of sliding windows that contain both word i and j, and $\#W$ is the total number of sliding windows in the corpus. The higher the PMI, the higher the semantic correlation of the pair.

2.2. Graph Convolutional Network

A Graph Convolutional Network (GCN) is a network made for semi-supervised learning on graph-structured data. It is built as an adaptation of convolutional neural network for graphs, and is able to induce embeddings for each node based on the feature of the node itself and those of its neighboring nodes. For the task of task classification, it tries to learn the label of the document node from the edges and closely connected nodes. The i-th layer take as input the neighbours of degree i of the target node. Here the GCN compute predicted labels of the whole graph but the loss is computed only on labelled nodes. The idea is to learn the structure of the documents from the unlabelled data and to propagate the labels through the graph based on a few labelled examples.

3. Experiment

The authors evaluate their TextGCN, composed of two layers, on two experimental tasks: the first one is text classification and the second one embeddings learning.

For both tasks, they compare TextGCN with various state-of-the-art models and ran their experiments on five widely used benchmarks datasets (20-Newsgroups, Ohsumed, R52, R8, Movie Review (MR)).

3.1. Text classification results

Without using any pre-trained models or embeddings, TextGCN significantly outperforms all baseline models on the first four datasets, based on test accuracy. They are two main reasons for that:

1. The text graph can capture relations between words and between words and documents
2. With the use of a two-layer GCN, label information of documents are passed to their neighboring word nodes, which are connected to other word nodes and documents, and thus enable to classify unlabelled documents by label propagation.

Parameter Sensitivity. The authors also evaluate the test accuracy with different sliding window sizes (hyperparameter for the computation of the PMI) on R8 and MR. They found out that varying window sizes does not result in test accuracy significant differences.

Similar results were observed by varying the first-layer of TextGCN embeddings' dimension.

Effetcs of the size of labeled data. This experiment consists in evaluating the test accuracy of the best performing models and TextGCN with different proportion of labeled training data (1%, 5%, 10%, 20%) of 20NG and R8. They showed that TextGCN achieves high accuracy with limited labelled documents, compared to the other models. For instance, with 20% of labelled training documents of 20NG, TextGCN already achieves a test accuracy of 0.8063. On R8, only 5% of labelling is necessary for TextGCN to outperform other models with a test accuracy of 0.8830. This results shows that GCN can perform well on dataset with a low label rate, and thus suggest that its ability to propagate document label information well.

3.2. Embeddings results

A comparison visualization of the document embeddings learned by TextGCN and two other models shows that:

- TextGCN's embeddings are more discriminative than the two other models
- The embeddings of the second layer of TextGCN are more discriminative than the embeddings of the first layer.

4. Discussion

TextGCN achieves strong text classification results and is able to jointly learn word and documents embeddings. However, one major drawback of a GCN lies in its transductive nature: a GCN is training simultaneously on training and unlabelled testing documents nodes, thus for unseen test documents, the GCN needs to be retrain in order to incorporate new documents.

References

Yao, L., Mao, C., and Luo, Y. Graph convolutional networks for text classification. *CoRR*, abs/1809.05679, 2018. URL http://arxiv.org/abs/1809.05679.