Recipe Text Classification using Graph Neural Networks

Stanford CS224N Custom Project

Goutham Krishna Teja Muppala
Department of Computer Science
Stanford University
krish24@stanford.edu

Abstract

Recipe text classification is used to categorize new recipe's based on the knowledge acquired from training dataset. Here we utilized Graph Convolution Neural Networks to perform the text classification. Graph Convolution Networks are very effective in non euclidean domains where complex relationships and interdependency between the objects are represented as graph. Graph Convolution neural networks are used to classify Food recipe text into various categories (like vegetarian, meat etc..,) and compare it with baseline of Long Short Term Memory (LSTM) neural network. we observed better validation performance with GCN compared to LSTM on same train and test dataset. Also GCN quickly achieves higher train and test accuracy at very early epoch compared to LSTM.

1 Key Information to include

- Mentor: Michihiro Yasunaga
- External Collaborators (if you have any): No
- Sharing project: No

2 Introduction

Text Classification is classical problem in natural language processing. Text classification has various wide applications like organizing documents, spam detection, web filtering, opinion mining and various other area. we will be using recipe text data from food.com to evaluate the performance of the classifiers. Food recipe have common features based on the category they belong to, as part of this task we will be classifying food recipes into categories like vegetarian and meat based recipes.

There are various text classification techniques [6] from shallow neural networks to deep learning. we can utilize support vector machines (SVM), bag of words, n-gram techniques which utilized localized features. Also, there are deep learning techniques like convolution neural networks(CNN's) and Recurrent Neural networks (RNN's) like LSTM can very well capture semantic and syntactic information in local consecutive words sequences. Where as graph neural network's are very effective at establishing global structure information using graph with graph embeddings.

3 Related Work

Shallow neural networks like SVM for text classification tend to perform better for less number of classes with low computational complexity. n-grams [7] probabilistic language model for predicting the next item in such a sequence in the form of a (n 1)–order which provides better feature engineering of words.

several others have utilized deep learning techniques employing CNN and RNN which are better at feature extraction. Tai et.al, used LSTM tree structures to improve the schematics representation. our work mainly utilizes [1] where text GCN is fed with document graph prepared before start of training.

4 Approach

To understand better on Graph Convolution Networks lets understand what is a graph. A graph represents the relations (edges) between a collection of entities (Vertex).

- **V** Vertex (or node) attributes e.g., node identity, number of neighbors
- **E** Edge (or link) attributes and directions e.g., edge identity, edge weight
- **U** Global (or master node) attributes e.g., number of nodes, longest path

GCN is semi-supervised learning on graph-structured data. It is based on an efficient variant of convolutional neural networks which operate directly on graphs[4]. The choice of convolutional architecture is motivated via a localized first-order approximation of spectral graph convolutions. The model scales linearly in the number of graph edges and learns hidden layer representations that encode both local graph structure and features of nodes. As for traditional CNN's, a GCN consists of several convolutional and pooling layers for feature extraction, followed by the final fully-connected layers.

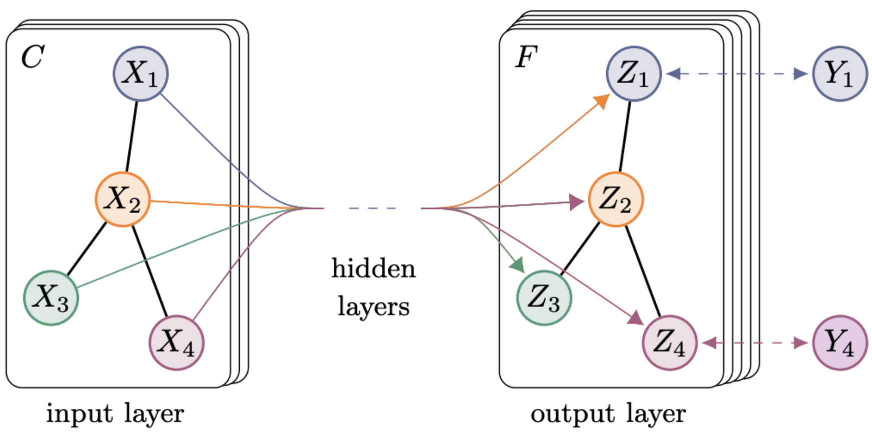

Figure 1: Graph Convolution Network[4]

A GCN [4] is a multilayer neural network that operates directly on a graph and induces embedding vectors of nodes based on properties of their neighborhoods. Formally, consider a graph G = (V, E), where V (|V | = n) and E are sets of nodes and edges, respectively. Every node is assumed to be connected to itself, i.e., (v, v) E for any v. Let X R n×m be a matrix containing all n nodes with their features, where m is the dimension of the feature vectors, each row xv R m is the feature vector for v. We introduce an adjacency matrix A of G and its degree matrix D, where Dii = P j Aij . The diagonal elements of A are set to 1 because of self-loops. GCN can capture information only about immediate neighbors with one layer of convolution[1].

5 Experiments

In this section we will evaluate Graph Convolution Network on Recipe Data set to find answers for below questions:

- can graph convolution network preform better compared to LSTM?
- which of the neural network model can achieve better accuracy earlier during training and validation?

5.1 Data

	category	recipe	encoded
0	0	make a choice and proceed with recipe dependin...	[[2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, ...
1	0	mix all ingredients& boil for 2 1 / 2 hours , ...	[[35, 82, 83, 84, 85, 22, 86, 87, 14, 86, 88, ...
2	0	toast the fennel seeds and lightly crush them ...	[[121, 58, 122, 21, 5, 123, 124, 125, 126, 82,...
3	0	drain green chiles sprinkle cornstarch on shee...	[[136, 137, 138, 139, 140, 10, 141, 142, 143, ...
4	0	heat the oil or margarine in a soup pot and ad...	[[186, 58, 26, 18, 187, 105, 3, 188, 189, 5, 1...

Figure 2: Recipe Text Dataset

1. **Data set** used is subset of food recipe data set[3] for training the neural network. we will be using only the category and recipe description for our classification task.As this dataset is huge with 250k entries, will be using equal distribution 5k recipe for our experiment.
2. **Pre-processing** Original dataset is prepossessed to fetch only the category and recipe text is con-catted to form a single continues string and special characters are removed before tokenized the text.

5.2 Evaluation method

we will evaluate the performance of the GCN with baseline of LSTM neural network with similar dataset and same ratio of train and test datasets. main comparison criteria will be on F-1 score/accuracy of the test dataset on model. Also we will see how fast the validation accuracy reaches higher over the lesser epoch's. Time taken for training will also be a key for performance.

5.3 Experimental details

Two layer graph convolution neural network is used to train our dataset with model parameters as hidden layer one size is 330, hidden layer two size is 130,number of classes as 2, number of epochs is 1000 and learning rate of 0.011. Also we have split dataset in 80 vs 20 for training and test use. GCN is implemented based on [1] and below is the graph preparation and GCN approach.

1. **Nodes** in the text graph |V | is the number of documents (corpus size) plus the number of unique words (vocabulary size) in a corpus. We simply set feature matrix X = I as an identity matrix which means every word or document is represented as a one-hot vector as the input to Text GCN
2. **Edges** among nodes based on word occurrence in documents (document-word edges) and word co-occurrence in the whole corpus (word-word edges).The weight of the edge between a document node and a word node is the term frequency-inverse document frequency (TF-IDF) of the word in the document.
3. **co-occurrence** point-wise mutual information (PMI), a popular measure for word associations, is used to calculate weights between two word nodes. A positive PMI value implies a high semantic correlation of words in a corpus, while a negative PMI value indicates little or no semantic correlation in the corpus.
4. **GCN** After building the text graph, we feed the graph into a simple two layer GCN as in the second layer node (word/document) embeddings have the same size as the labels set and are fed into a softmax classifier

5.4 Results

Comparison between test accuracy vs train accuracy over the epochs for graph convolution network is provided in Figure 3. For the recipe classification using GCN, we observed higher validation accuracy for GCN compared to LSTM network. Below table show the F1-score comparison between two. Also GCN can achieve better prediction accuracy even with smaller training dataset.

Neural Network	weighted F1-score	Validation Accuracy
GCN	0.74	0.74
LSTM	0.66	0.66

Table 1: A table caption.

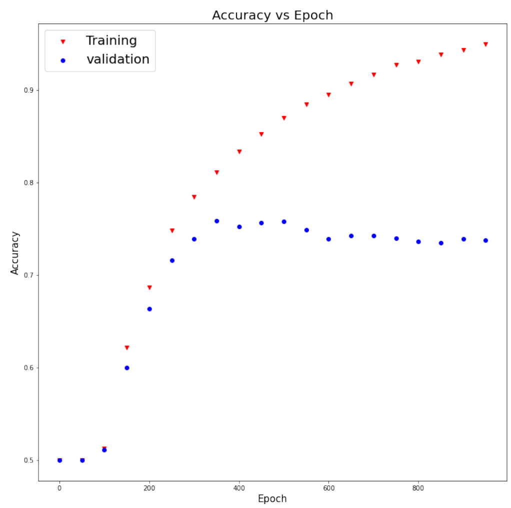

Figure 3: GCN train vs test accuracy

6 Analysis

Graph convolution networks perform better but they do have overhead of graph generation before the actual training begins which is time consuming as the number of document and dataset becomes huge. This comes with advantage of faster training compared to LSTM.

7 Conclusion

For the recipe classification using GCN, we observed higher validation accuracy for GCN compared to LSTM network. Also GCN can achieve better prediction accuracy even with smaller training dataset. As future work we can see how attention can help GCN to perform better. As part of this project i learnt how graph neural networks works and their benefits.

References

[1] Yao, Liang, Chengsheng Mao, and Yuan Luo. "Graph convolutional networks for text classification." Proceedings of the AAAI conference on artificial intelligence. Vol. 33. No. 01. 2019.
[2] Huang, L., Ma, D., Li, S., Zhang, X., Wang, H. (2019). Text level graph neural network for text classification.
[3] Food recipe Dataset `https://www.kaggle.com/shuyangli94/food-com-recipes-and-user-interactions`
[4] Kipf, Thomas N., and Max Welling. "Semi-supervised classification with graph convolutional networks." arXiv preprint arXiv:1609.02907 (2016).

[5] Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." Neural computation 9.8 (1997): 1735-1780.

[6] Li, Qian, et al. "A survey on text classification: From shallow to deep learning." arXiv preprint arXiv:2008.00364 (2020).

[7] Wang, Sida I., and Christopher D. Manning. "Baselines and bigrams: Simple, good sentiment and topic classification." Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2012.

[8] Tai, Kai Sheng, Richard Socher, and Christopher D. Manning. "Improved semantic representations from tree-structured long short-term memory networks." arXiv preprint arXiv:1503.00075 (2015).

A Appendix (optional)

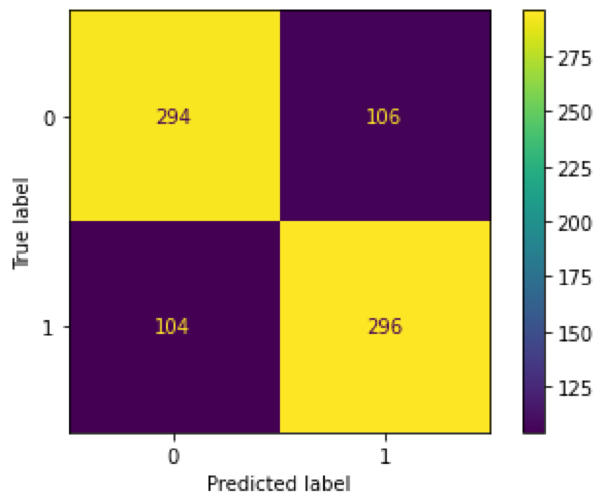

Figure 4: Confusion Matrix for GCN model validation