Open Classification of Text Document Topics

Qian Yu
UC Berkeley MIDS
q̑ianyu@ischool.berkeley

Varadarajan Srinivasan
UC Berkeley MIDS
v̑aradarajan@ischool.berkeley

Leslie Teo
UC Berkeley MIDS
l̑teo01@ischool.berkeley

Abstract

Due to the dynamic nature of the online text, new online documents may not belong to any previously defined training classes. Deep Open classification (Shu, Xu, and Liu, 2017) is a new deep learning approach to solve this challenge. The architecture consist of a CNN architecture with a 1-vs-Rest output layer.

We propose to use a different approach. Specifically, we explore clustering algorithms as a separate step to determine open class documents. We compare our experiment with the results reported by the DOC reference paper (Shu, Xu, and Liu, 2017)

1 Credits

This project is based on a published paper on the subjects of open classification titled: DOC: Deep Open Classification of Text Documents (Shu, Xu, and Liu, 2017). We also refer to other papers on lifelong machine learning (Chen and Liu. 2016), convolutional neural network for sentence classification (Kim, 2014), paragraph vector (Le and Mikolov, 2014) and task clustering (Thrun and O'Sullivan, 1996)

We are also grateful for the mentoring and recommendations of Ian Tenney

2 Introduction

A supervised text classification model can be used to learn and classify documents based on topics or genres with good labeled training data. However, in the web 2.0 world, there are new contents constantly generated from social media, news articles, and blogs. Due to the dynamic nature of online text, a new document may not belong to any previously defined training classes. The key assumption of supervised learning is violated.

The problem is called open classification (Fei and Liu, 2016), in which the classifier can detect an unseen class. Open classification is also part of a new machine learning paradigm called Lifelong Machine Learning (LML) (Chen and Liu, 2014a). It is particularly valuable in learning the abundant and multifarious information from the web. In the natural language learning setting, open classification can be used to filter unwanted documents or discovery new categories. For example, we can use open classification method to detect subjectivity among news articles.

3 Background

Our project's main idea is based on (Shu, Xu, and Liu, 2017)'s approach of using a *Convolutional Neural Network* (CNN) architecture because CNN's excellent performance and efficiency on sentence classification (Kim, 2014). An 1-vs-Rest output layer with $m \times n$ sigmoid functions are used for open classification where m is size of classes and n is the batch size. The prediction of sigmoid function is reinterpreted at the testing time to determine the unseen open class. A document is belong the open class if its sigmoid probabilities are less than thresholds of all classes. In addition to using 0.5 as the threshold, an outlier detection method is used to obtain better thresholds for different classes.

We can apply the clustering method in replacement of 1-vs-Rest approach as the output layer after the CNN layer to determine an open class document. Using clustering method for online learning is a well-know practice. Particularly, task clustering (Thrun and O'Sullivan, 1996) is an old concept but is based on a similar idea to lifelong learning. This idea can also be applied to open classification problem. In task clustering, When a new task arrives, it first selects the most similar cluster then use the distance function of the cluster for classification in the new tasks (Trun, 1996b). Concretely, we can take the trained feature vectors of documents from CNN architecture and use them as input parameters of distance calculation of clustering analysis. Using Gaussian mixture algorithm as an example, we can calculate a new sample's distances to the centers of all cluster distributions. If the new sample is an outlier to all the cluster distributions, it is rejected as unseen class.

4 Methods

4.1 Data Set

We use **20 Newsgroups** (Rennie, 2008) data set for our experiment. The data set contains 20 non-overlapping classes in newsgroup topics. Each class has around 1000 samples.

4.2 Paragraph Vector model

We decided to use paragraph vector model(Le and Mikolov, 2014) as the baseline model for our experiment. The paragraph vector model produce a fixed-length feature representations of a paragraph from variable-length pieces of texts. Compared to Bag of Words (BOW) model, it provides a dense feature vector representation of documents, capturing ordering and semantics of words. This is very similar to a CNN model. Therefore, We decided to use paragraph vector model as the baseline model for our experiment

4.3 CNN Model Architecture

We use CNN as our main language model architecture. Based on the recent study of document and sentence classification using CNN (kim, 2014; Zhang and Wallace, 2016), CNN offer excellent performance in sentence classification compared to other state of the art languages such as LSTM-RNN, MV-RNN, and RAE. Using CNN architecture also allows us to have a fair comparison with DOC (Shu, Xu, and Liu, 2017)'s test settings and evaluation metrics.

We use pre-train word embedding model and training a word embedding model with CNN on the fly for the input layer to CNN. We will select the better word embedding model for our final input layer to CNN. Each document in the data is padded or cut into a fix length of S number of words, e.g. 1000 words. The document is then transformed into an $S \times d$ dense matrix using embedding look up table. The CNN internal dimension can be mirror to the DOC architecture (Shu, Xu, and Liu, 2017) where 3 regions of $[3, 4, 5]$ and 150 filters was used. Lastly, 2 fully connected layers with Relu activation is used before the output layer.

$$Output = W'(ReLU(Wh + b)) + b'$$

4.4 Open Classification Methods

The DOC (Shu, Xu, and Liu, 2017) used 1-vs-Rest layer to determine the open classification. Instead of using a softmax as the final output layer, sigmoid function is used for each document for each class. The objective function is the summation of all log loss of sigmoid function

$$loss = \sum_{i=1}^{M} \sum_{i=1}^{N} y_n log(p) + (1 - y_n) log(1 - p(y))$$

where m is the number of classes and n is the batch size. During the test, we determine if a new document is unseen or open class when its sigmoid probability of all classes is smaller than a threshold. Outlier detection method is used to determine the thresholds for different classes. We use 1-vs-Rest as the baseline for the open classification analysis. We use clustering method as an alternative method for open classification. The mechanics of the analysis is different from the 1-vs Rest approach. First, we train a CNN model using a softmax output layer instead of using 1-vs-Rest output layer as implemented in the DOC architecture. We then extract the trained feature parameters from the output of the CNN model use it as input parameters of the clustering analysis. We use clustering analysis on the test data to detect open class. Concretely, we compare the location of a sampe in comparison to the center of the cluster distributions. If a sample is an outlier in relations to the cluster distributions of all classes, it belongs to an unseen class. We use 2 clustering algorithm for open classification analysis:

- Gaussian Mixture Model
- Infinite Dirichlet process

5 Results and Discussion

5.1 Test Metrics

For a fair comparison to the DOC architecture, we use the same 60%, 10% and 30% data split for training, validation and testing. We hold out some classes (as unseen) in training and add them back during testing. We vary the number of training classes use 25%, 50%, 75% and 100% classes and randomly select classes for training and testing. We use macro F1-score over $m+1$ classes for evaluation

5.2 First Iteration: Concept Exploration

In the initial phase of the project, we reviewed a few simple text models such as TD-IDF vector representation, Bag of Words (BOW) model, and paragraph vector (Le and Mikolov, 2014) during the absences of CNN model. We decided to use paragraph vector as our baseline model since it provides a similar dense vector representation of document before we complete the implementation of CNN model.

We tested 1-vs-Rest open classification analysis steps using logistical regression and SVM with sigmoid kernel by hold out 1 class during the training and add it back during testing. We learned that it is important to use the right threshold for open classification. A statistical approach similar to the DOC's outlier detection method (Shu, Xu, and Liu, 2017) is required.

6 Next Steps

In the next iteration, We will implement a CNN architecture and build open classification block with both 1-vs-Rest method and clustering algorithms. Our experiment grid is define in table 1.

Language Model	**Open Classification**
paragraph vec	1-vs-rest
paragraph vec	GM
paragraph vec	IDPs
CNN	1-vs-rest
CNN	GM
CNN	IDP

Table 1: Experiment Table.

We will use the similar testing and evaluation metrics described in the DOC paper (Shu, Xu, and Liu, 2017) to report our final results and discovery.

6.1 Project Github (Working in Progress)

`https://github.com/qianyu88/W266_project_submission`

References

Lei Shu, Hu Xu, and Bing Liu. 2017. *DOC: Deep Open Classification of Text Documents*.

Zhiyuan Chen and Bing Liu. 2016. *Lifelong Machine Learning*.

Yoon Kim. 2014. *Convolutional neural networks for sentence classification* .

Ye Zhang and Byron C. Wallace. 2016. *A Sensitivity Analysis of (and Practitioners Guide to) Convolutional Neural Networks for Sentence Classification*

Zhiyuan Chen and Bing Liu. 2014. *Mining topics in documents: standing on the shoulders of big data.*

Quoc Le and Tomas Mikolov. 2014. *Distributed Representations of Sentences and Documents*

Sebastian Thrun and Joseph OSullivan. 2014. *Learning More From Less Data: Experiments With Lifelong Robot Learning*