New classification techniques

Abstract

The study focused on comparing different techniques performance on text classification. We used 3 different methods to classify news topics, which were LDA, CNN, and RNN. The new data were THUCNews that originally classified into 14 topics. After the classification, the results were evaluated by calculating precision, recall and f1-score. Overall, the CNN and RNN were both good, their accuracies were over 90%. In opposite, the LDA performed poorly. The study may be served as a reference when choosing the method to perform text classification.

Keyword: text classification, LDA, CNN, RNN

1. *Introduction*

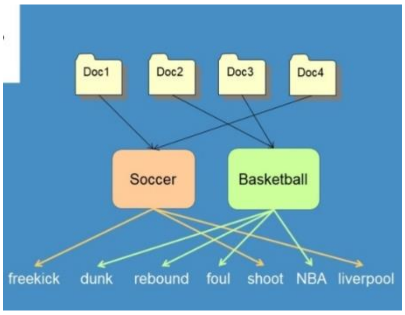
With the development of information technology, we are force to handle a huge volume of data every day. Thus, how to deal with data has become an important thing for us. As we know, there are many models for classifying data. In order to see which model's classification task performance is the best, we try to have a comparison of different models among LDA, CNN & RNN and measure them by calculating precision, recall, and f1-score etc.

2. *Literature review*

For our news classification project, we had some research on implementing text classification. LDA is a generative probabilistic model for collections of discrete data such as text corpora. Each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities. In the context of text modeling, the topic probabilities provide an explicit representation of a document. As for LDAvis, a web-based interactive visualization of topics estimated using Latent Dirichlet Allocation that is built using a combination of R and D3. Our visualization provides a global view of the topics (and how they differ from each other), while at the same time allowing for a deep inspection of the terms most highly associated with each individual topic.

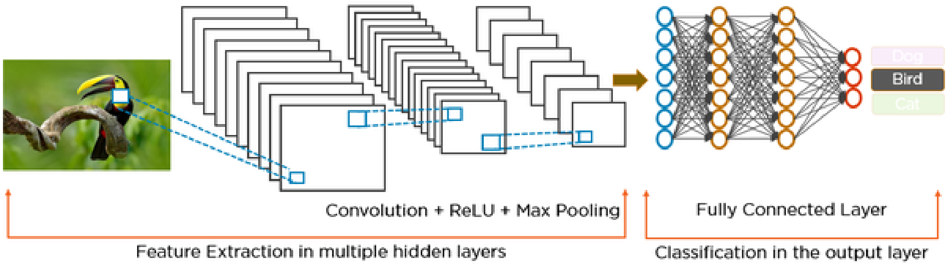
Since recurrent neural network is suit for sequence to sequence model and it considers not only the current input but also the previously received inputs, we picked RNN as the standard model to implement text classification tasks. Moreover, convolutional neural network is generally used for Image recognition and object classification tasks, so we put more effort on reading through papers discussing using CNN for text classification. By doing the embeddings to text, we were able to form the input as a 64 * 6000 matrix to represent our text data where 64 is the number of dimensions, and 6000 are the maximum word for our model. For more information, the input of the RNN model are processed embeddings of vector shape, which is also in 64 dimensions.

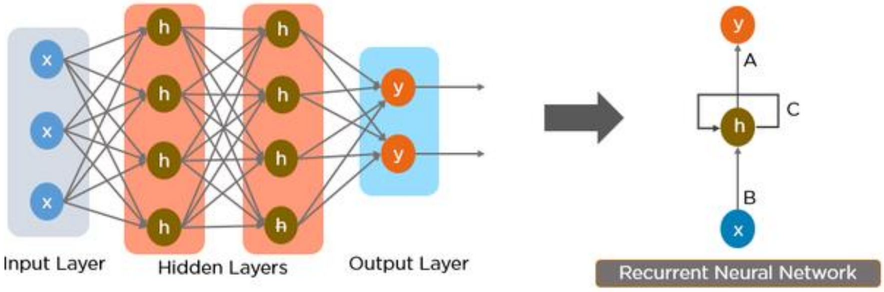
3. Architecture



LDA is described as a "generative probabilistic model of a corpus". It is a statistical technique that can extract underlying themes/topics from a corpus. In the LDA approach, instead of modeling the similarities between each text document and each word token directly, the "Latent variables" are introduced as "bridges." Each document within the corpus is characterized by a Dirichlet distribution over the topics and each topic is characterized by another Dirichlet Distribution over all the word tokens.

Convolutional Neural Networks (CNN) is one of the variants of neural networks used heavily in the field of Computer Vision. There are two main parts: multiple hidden layers for feature extraction and output layer for classification. In hidden layers, we will do the Convolution, ReLU and Mas Pooling.


Dog
Bird
Cat
Convolution + ReLU + Max Pooling
Fully Connected Layer
Feature Extraction in multiple hidden layers
Classification in the output layer

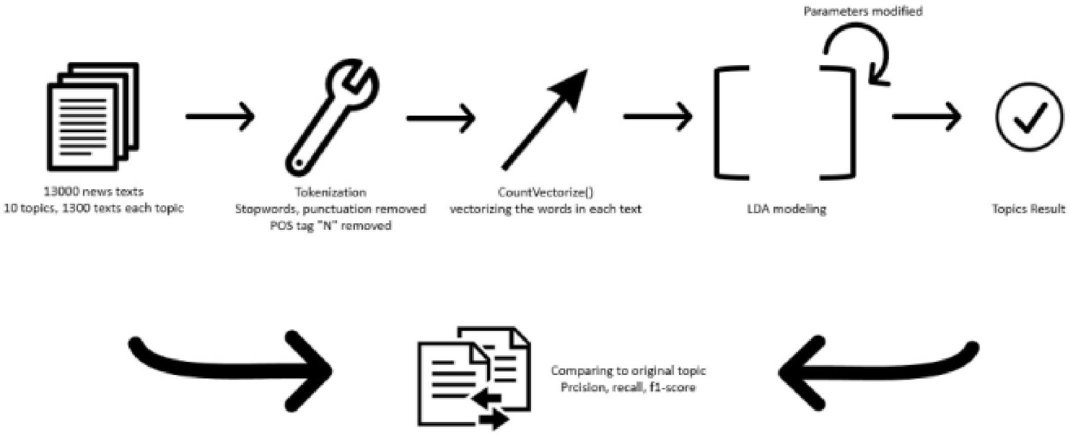


Recurrent Neural Networks (RNN) is a very important variant of neural networks heavily used in Natural Language

Processing. In a general neural network, an input is processed through a number of layers and an output is produced, with an assumption that two successive inputs are independent of each other.

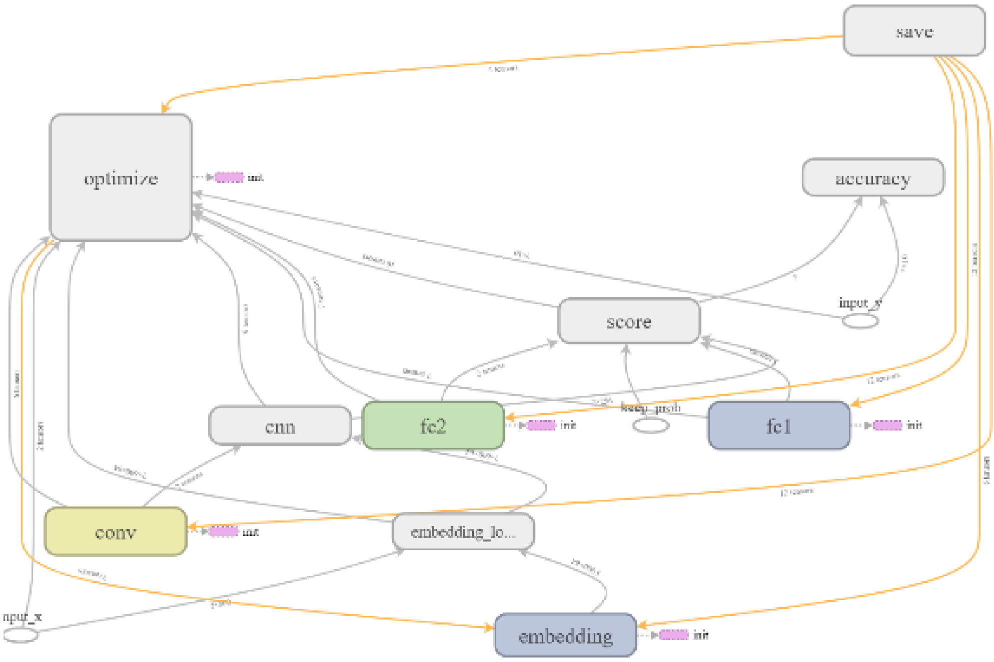
4. ***Implementation***

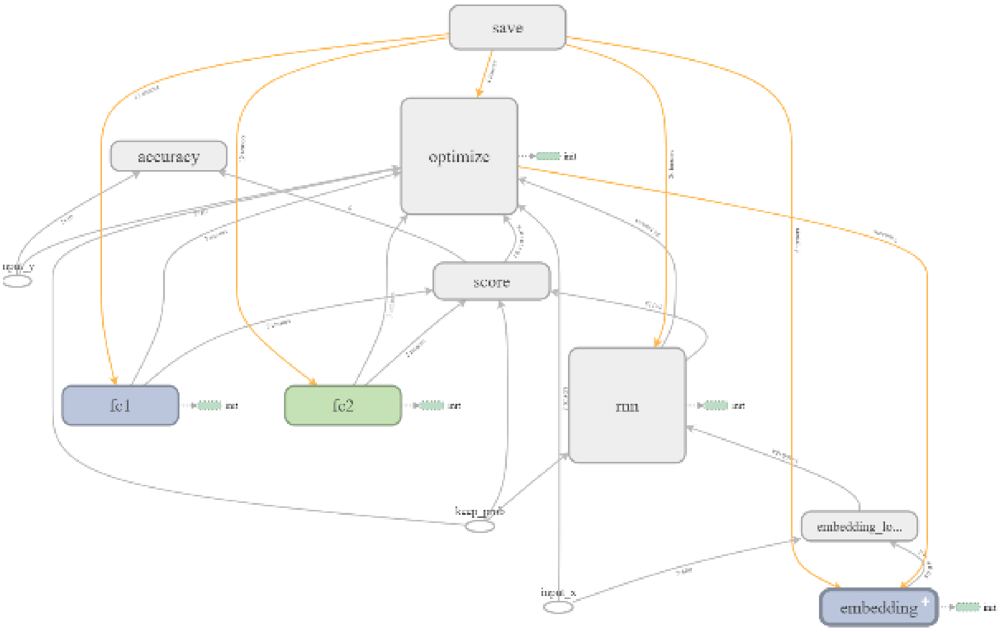
The LDA were used to do the unsupervised learning classification. All the data was preprocessed fisrt (tokenization, stopwords and punctuation removed, and POS tag ”N” remained). And then the data was vectorized and classified, which the phases repeated trials and errors to find the best parameters for performing in this study. The figure below shows the whole process.


Parameters modified
13000 news texts
10 topics, 1300 texts each topic
Tokenization
Stopwords, punctuation removed
POS tag "N" removed
CountVectorize()
vectorizing the words in each text
LDA modeling
Topics Result
Comparing to original topic
Prcision, recall, f1-score

a. CNN & RNN

   In the embeddings processing part, we had built the vocabulary list and named ids for all the words. Then we initialized the words to 64-dimension vector and will be trained later in our NN models. In RNN models, we built two RNN Layers followed with two Fully Connected Layers with the softmax activation function. As for CNN models, we built two CNN Layers followed with a Max Pooling Layers to reduce noises plus two Fully Connected Layers also with the softmax activation function. For more detailed part, the platform we are using is tensorflow, which provides user to record some detailed part in the training such as embeddings PCA illustration, accuracy and loss, model graph, etc. The graph presented below are CNN on the left and RNN on the right.

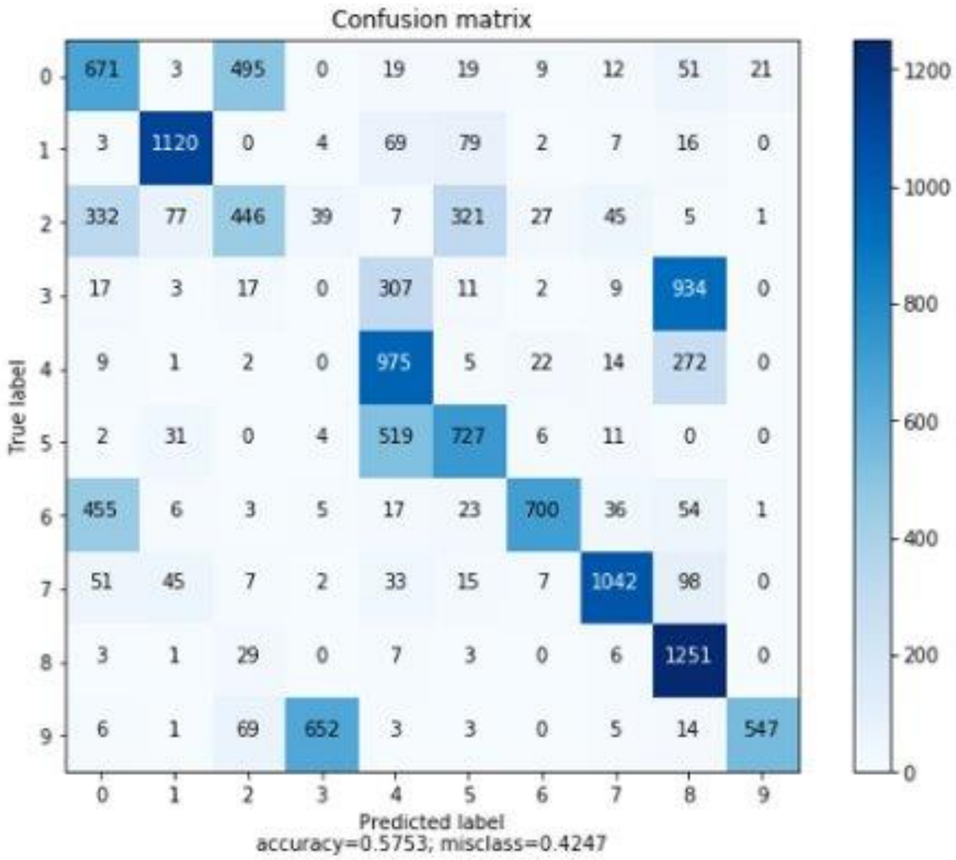




5. Evaluation (describe how you evaluate the system and the results)

a. LDA

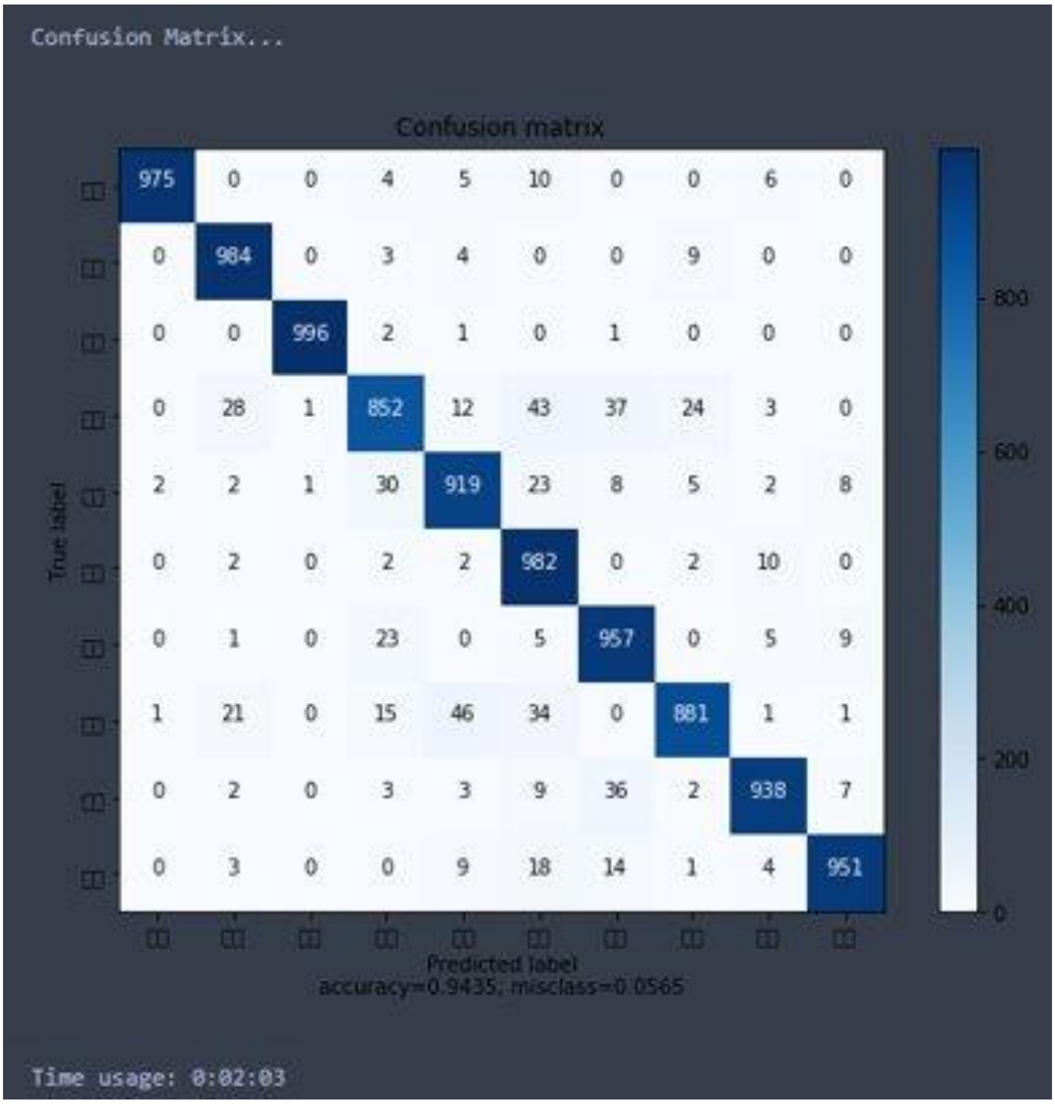
Since the method is unsupervised, we evaluated the model by manual comparing the result to the original topics from the resource. We calculated the score of precision, recall and f1-score and printed the confusion matrix for better presentation. And we used LDAvis to find out topics' keywords to further analyse the model (see the table in appendix). From the evaluation, we could see the performance of the LDA wasn't good.



True label \ Predicted label	0	1	2	3	4	5	6	7	8	9
0	671	3	495	0	19	19	9	12	51	21
1	3	1120	0	4	69	79	2	7	16	0
2	332	77	446	39	7	321	27	45	5	1
3	17	3	17	0	307	11	2	9	934	0
4	9	1	2	0	975	5	22	14	272	0
5	2	31	0	4	519	727	6	11	0	0
6	455	6	3	5	17	23	700	36	54	1
7	51	45	7	2	33	15	7	1042	98	0
8	3	1	29	0	7	3	0	6	1251	0
9	6	1	69	652	3	3	0	5	14	547

b. CNN & RNN

We evaluated our model by accuracy on the test set (we had validation set while training), and then we had the time of the system recorded to examine the execution time. Then we introduced the score of Precision, Recall, and F1-score for each category, the higher the f1-score, the better the model can predict the text from the category. Lastly, we print out the confusion matrix with heatmap in order to let human can easily observe which category has the better accuracy. The result of the CNN and RNN were very good.

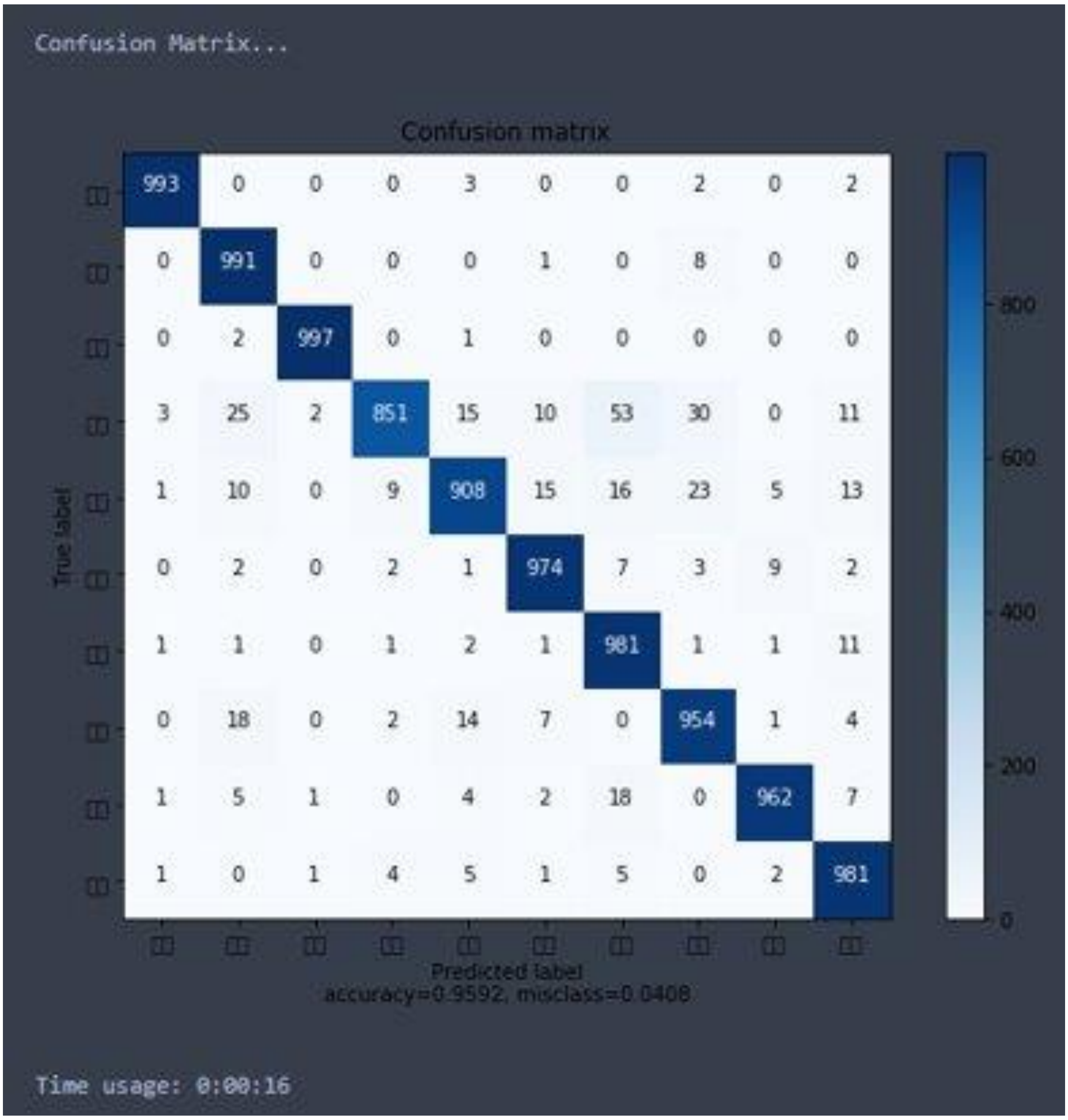


```
Confusion Matrix...
```

975	0	0	4	5	10	0	0	6	0
0	984	0	3	4	0	0	9	0	0
0	0	996	2	1	0	1	0	0	0
0	28	1	852	12	43	37	24	3	0
2	2	1	30	919	23	8	5	2	8
0	2	0	2	2	982	0	2	10	0
0	1	0	23	0	5	957	0	5	9
1	21	0	15	46	34	0	881	1	1
0	2	0	3	3	9	36	2	938	7
0	3	0	0	9	18	14	1	4	951

accuracy=0.9435; misclass=0.0565

```
Time usage: 0:02:03
```



```
Confusion Matrix...
```

993	0	0	0	3	0	0	2	0	2
0	991	0	0	0	1	0	8	0	0
0	2	997	0	1	0	0	0	0	0
3	25	2	851	15	10	53	30	0	11
1	10	0	9	908	15	16	23	5	13
0	2	0	2	1	974	7	3	9	2
1	1	0	1	2	1	981	1	1	11
0	18	0	2	14	7	0	954	1	4
1	5	1	0	4	2	18	0	962	7
1	0	1	4	5	1	5	0	2	981

```
Time usage: 0:00:16
```

6. ***Discussion LDA***

We used 10 different topics news dataset, hence, when modeling the LDA, we set the number of topics as 10. However, it seemed to cause problems. From the table below, we could notice the number of news in each topic is not equal to 1300. It seems the LDA model did not agree with the topic human-defined. For the “topic 3”, in the table below, we could find the number of texts were only 707, and there were words related to “sports”. Meanwhile, in the “topic 9” there were only 571 texts and the words were also related to “sports”. The LDA model classified the original sport topic into 2 different topics. For the “topic 8”, the number of news texts were 2695, the words were related to “fashion” and “entertainment”. The LDA model combined the “fashion” and “entertainment” to one topic. It is surmised that the result happened because some of topics might be highly related at first, the words used in the texts were common, and different types of noun, such as “name”, “jargon”,etc., were received differently in the model.

Since the LDA model generated without data labeling. The situation could not be avoided, but it could be improved. For instance, words selecting and topic selecting might be important in the beginning process, besides adjusts parameters in the LDA model. No labeling is the nature of the unsupervised model, It can’t be trained to be as precise as supervised model when involving human-defined subject. However, it is still useful and valuable when it comes to unlabeled data classification and underlying theme discovering.

a. CNN & RNN

   We did not have the clue that why the CNN model has the highest accuracy and the least training time. The training time interval from CNN model is about 250 shorter than RNN model, while CNN model even has a higher accuracy on predicting the categories each text belongs. Moreover, in the NN models, we did not use some techniques about tf-idf or other preprocessing methods while still acquire such great result.

7. Conclusion

From the result, LDA model is not performed as good as the CNN & RNN. From our perspective, the reason is that LDA use the One-Hot vector and CNN & RNN use 64dimension vector. The classification of CNN and RNN were not significantly different in our test. However, the time to finish classify process different.

Completion time: RNN > LDA > CNN (29 : 9 : 1).

To sum up, the CNN was quick and well performed model, it is recommended when classifying task.

8. Code (github website)

https://github.com/alousu0612/Text-Mining-news-classification

9. References

a. http://www.wildml.com/2015/12/implementing-a-cnn-for-text-classification-in-tensorflow/

b. https://arxiv.org/abs/1408.5882

c. https://arxiv.org/abs/1509.01626

d. http://thuctc.thunlp.org/

e. https://www.tensorflow.org/

f. https://towardsdatascience.com/understanding-neural-networks-from-neuron-to-rnn-cnn-and-deep-learning-cd88e90e0a90

g. BLEI, David M.; NG, Andrew Y.; JORDAN, Michael I. Latent dirichlet allocation. *Journal of machine Learning research*, 2003, 3.Jan: 993-1022.

h. SIEVERT, Carson; SHIRLEY, Kenneth. LDAvis: A method for visualizing and interpreting topics. In: *Proceedings of the workshop on interactive language learning, visualization, and interfaces*. 2014. p. 63-70.