News Category Prediction

Zoe Zheng, Ting-Chih Chen, Huayu Liang, Yilin Liu
Virginia Polytechnic Institute and State University
Blacksburg, Virginia, USA
{zoez,tingchih,liupp,huayu98}@vt.edu

Abstract

We propose 4 methods (Naive Bayes, Logistic Regression, SVM and Neural Network) to solve this multi-class classification problem. The dataset is extraced from Microsoft News Dataset. Our main approach is to use bag of word and tf-idf to process text. In order to compare performance, we not only adopt F1-score and accuracy but also use confusion matrix to understand the models' advantage and disadvantage.

ACM Reference Format:
Zoe Zheng, Ting-Chih Chen, Huayu Liang, Yilin Liu . 2022. News Category Prediction. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 Introduction

In this project, we analyze the open source data set from Microsoft which as known as MIND (Microsoft News Data set). Our goal is to train four models to classify the news category from their titles and abstract. After training, our different models can predict the category of the test news. Using the predict results, we compare to the correct category within the data set and examine the accuracy of each model.

2 Dataset

Our project's dataset is from Microsoft News Dataset(MIND). We extract the news title, abstract and category from MIND. We have 81,222 news in the training set and 20,305 news in the testing set. There are 16 categories in the dataset. Categories include autos, entertainment, finance, food and drink, health, kids, lifestyle, movies, music, news, north america, sports, travel, tv, video and weather.

Figure 1 is training set and Figure 2 is testing set.

Conference'17, July 2017, Washington, DC, USA
2022. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

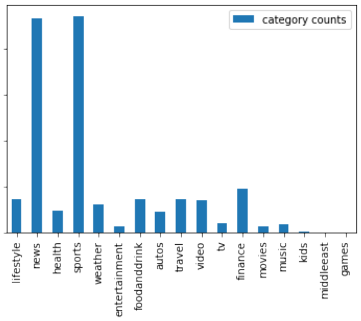


Figure 1. Training set

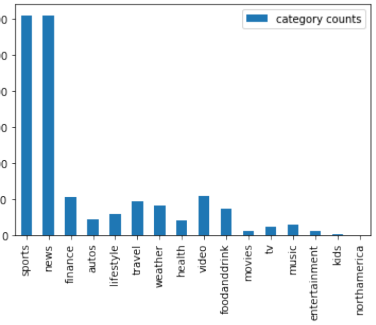
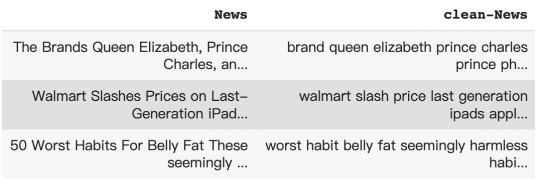


Figure 2. Testing set

2.1 Text prepossessing

For each news corps, we have feature **Title** and feature **Abstract**. Experiments have shown that using the combine text data of **Title** and **Abstract** will result a better category classification accuracy. To build a better BoW matrix and remove noise from the text data, we normalize the news to all lower letter words; remove punctuation and stopwords; remove single letter words and numbers. The text preprocessing example is in Figure 3.



News	clean-News
The Brands Queen Elizabeth, Prince Charles, an...	brand queen elizabeth prince charles prince ph...
Walmart Slashes Prices on Last-Generation iPad...	walmart slash price last generation ipads appl...
50 Worst Habits For Belly Fat These seemingly ...	worst habit belly fat seemingly harmless habi...

Figure 3. Text preprocessing

3 Methods

3.1 TF-IDF

Term Frequency indicates the frequency of the term appearing in the text, we fit it into the BoW matrix as the first experiment input for training our classification models. However, some common words do not have much effect on the sentences, but some words that appear less frequently can express the sentences of the article more, so simply using TF is not suitable. Inverse Document Frequency indicates brings in more information of the term. The fewer documents containing the term, the larger the IDF, which means the term has a good ability to distinguish between categories. We fit it into the BoW matrix as the second experiment input for training our classification models.

3.2 Naive Bayes

We make two different training sets for Naive Bayes experience. One is that the training set only includes the news title; the other one is that the training set includes the news title and abstract. We make 2 different Naive Bayes models to compare their performance and accuracy. In the Naive Bayes program, we use sklearn.CountVectorizer() to convert the training set of text documents to a matrix of token counts. In addition, we put this matrix of token counts to transform to a normalized tf-idf representation using sklearn.TFidfTransformer(). At last, we implement Naive Bayes models and test in our testing set.

3.3 Logistic Regression

For Logistic Regression, we use the same two training sets. For the implementation, we use two distinct Logistic Regression models to train and test our datasets and compare their performance and accuracy. Firstly we use the sklearn.CountVectorizer() to convert text documents to a matrix of token counts. Then we use the LogisticRegression() from sklearn.linearmodel to train our training set and using the trained model to test our testing set. Finally, we use the confusion matrix and classification report from sklearn.matrics to show the performance and F1 score of each category.

3.4 SVM

For the method of SVM, we used same data setting as we did in Naive Bayes, one include news and title and one contains the news title and also the abstract. For the program, we use sklean.svm for classification. In sklearn.svm, svm.SVC() svm.linearSVC() can be used for Support Vector Classification. Here we are dealing with larger number of data samples, so we chose svm.linearSVC() in our code. We make two models from each data sets.

3.5 Neural Network

We use the Keras Python library to perform multi-class classification task on news category prediction using Neural Network. The input sequence is a BoW matrix with each row represent each news and each column represent each words in that news. We built a fully connected Neural Network with one dense layer and a Softmax layer at the end to generated the category probabilities. The Figure 4 is our Neural Network model.

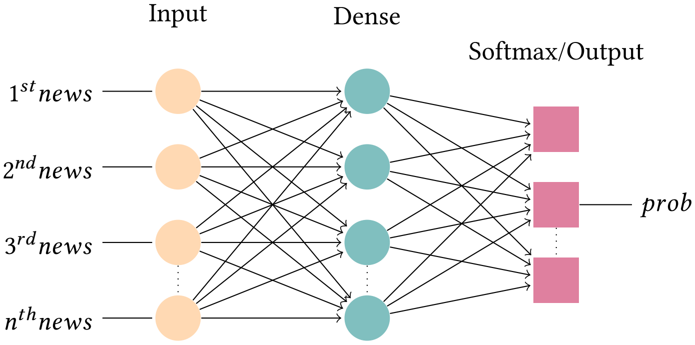


Figure 4. The architecture of Neural Network Model

4 Results

4.1 Naive Bayes

In the Naive Bayes model, we have 63% accuracy in training sets that only have news title and news title and abstract. We have a nice prediction in the news and sport categories. In the dataset, news that in the news and sport categories have lots of news counts. This affects Naive Bayes model cannot learn the words in others' categories.

Figure 5 is Naive Bayes confusion matrix.

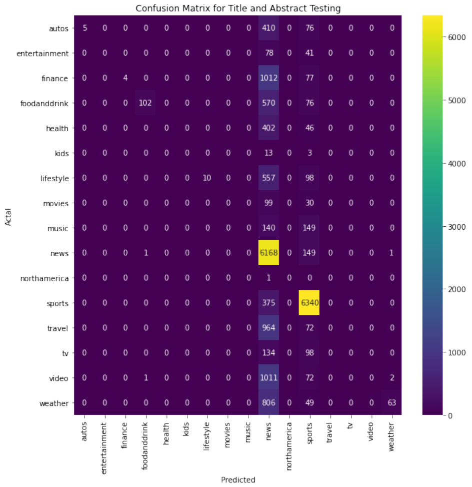


Figure 5. Naive Bayes confusion matrix

4.2 Logistic Regression

In the Logistic Regression training model, we have 70% in training set with only the title feature and 73% in training set with combination of title and abstract. The two models both make a pretty prediction on categories News and Sports which consist much part of the news counts. That's why the other categories' accuracy are lower than News and Sports. Figure 6 is Logistic Regression confusion matrix.

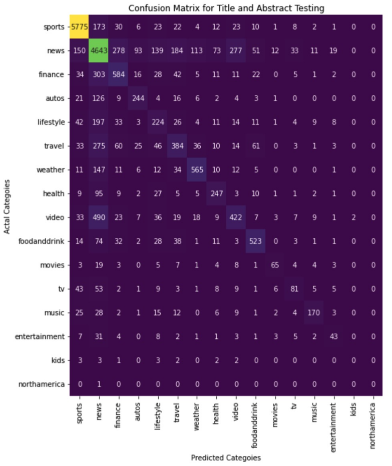


Figure 6. Logistic Regression confusion matrix

4.3 SVM

For SVM modelling, the accuracy for only title is 68.49%, the accuracy for abstract and title is 70.57%. Figure 7 is SVM confusion matrix.

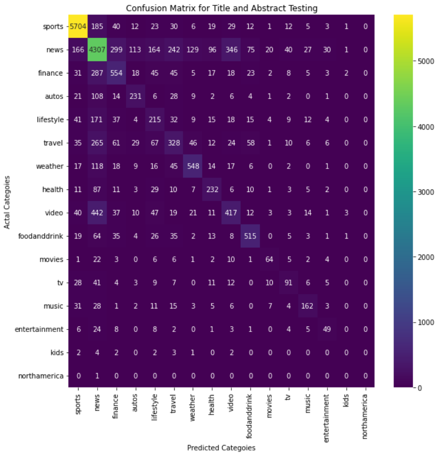


Figure 7. SVM confusion matrix

4.4 Neural Network

We experimented models with one hidden dense layers with{50, 128, 256} nodes and two layers with each 64 nodes, all for 10 epochs.

The best performing model is one layer with 126 nodes which has 73.38% accuracy. The prediction result is shown below in the confusion matrix Figure 8.

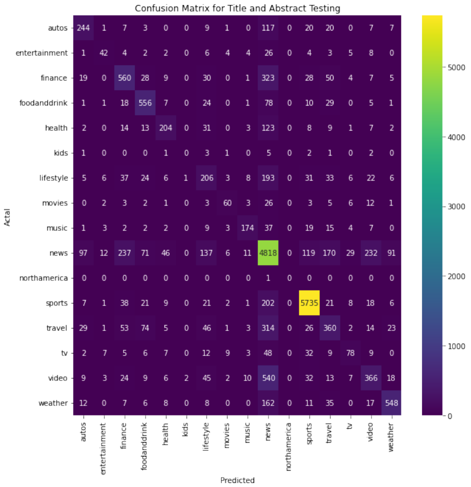


Figure 8. Neural Network confusion matrix

5 Conclusion

Overall, the neural network model has the highest accuracy and followed by logistic regression. Naive Bayes has the worst performance. Performing text cleaning results in general 1% more accuracy. Performing TF-IDF results in general 2% more accuracy than TF.

6 Contributions

Experiment	Assignee	Details
Text clean	Zoe	Performed text prepossessing.
Neural Network	Zoe	Built Neural Network models.
Neural Network	Zoe	NN hyperparameter tuning.
Neural Network	Zoe	NN model evaluation.
Data pre-processing	Ting-Chih	Extracted dataset from MIND.
Implement model	Ting-Chih	Performed Naive Bayes model.
Evaluation	Ting-Chih	F1 score, Confusion matrix, and Accuracy.
Implement model	Huayu	Logistic Regression model
Evaluation	Huayu	F1 score, Confusion matrix, and Accuracy
Implement model	Yilin	Performed SVM model.

Table 1. The Contribution of Each Team Member