CMPE493: Information Retrieval Assignment 2

Elvan Karasu

1 Introduction

In this assignment, multinomial Naive Bayes (NB) and multivariate Bernoulli Naive Bayes algorithms are implemented using the Python Standard Library. Reuters-21578 data set is used, which contains 21578 documents. [1] In order to classify the topics of these documents, various text classification models are developed and compared.

2 Preprocessing

The documents with top-10 topics are extracted. The distribution of topics among training and test sets are shown in Table 1. For normalization, case-folding used, stopwords, punctuation, and numbers are removed. After these steps, the resulting vocabulary contains 29971 unique words.

Table 1: Number of documents for training and test sets

Topic	Training set	Test set
earn	2868	1083
acq	1636	717
money-fx	503	164
grain	387	137
trade	350	112
crude	344	167
interest	225	101
ship	147	60
wheat	20	3
corn	10	1
Total	6490	2545

3 Method

There are 6490 and 2545 documents in the training and test sets, respectively. For the development set, 1000 documents from the training set are chosen randomly. The resulting train/dev/test set ratio is 61/11/28. The models are trained with the training set and using different values of alpha, and the performance of these models are evaluated on the development set. Micro and Macro F scores are shown in Table 2 and 3 for multinomial and multivariate Bernoulli NB, respectively.

4 Results

Table 2: Performance of multinomial NB on the development set

alpha	Micro F Score	Macro F Score
alpha = 0.5	0.932	0.690
alpha = 1	0.929	0.680
alpha = 2	0.927	0.647

Table 3: Performance of multivariate Bernoulli NB on the development set

alpha	Micro F Score	Macro F Score
alpha = 0.5	0.824	0.502
alpha = 1	0.814	0.490
alpha = 2	0.805	0.479

Multinomial and multivariate models are trained using the training set and varying values of alpha. The commands for training multinomial and multivariate NB models can be found in Figure 1 and Figure 2. After evaluating the models on the development set, best values of alpha are found to be 0.5 for both models. The results are given in Table 2 and Table 3. After tuning the parameter alpha, the models are trained again, this time using the training and development tests together. The performances are evaluated on the test set and the results are given in Table 4. In order to compare the two models, approximate randomization test is applied and p-value is found to be 0.001. It is smaller than 0.05, hence the null hypothesis is rejected. The multinomial NB model with alpha = 0.5 is significantly different than multivariate NB model, and another advantage is that it also runs faster. This result is reasonable, since the performance of multinomial models are better than that of multivariate one for every experiment, as seen in Table 4. The command given in Figure 3 trains the best models for multinomial and multivariate NB, and applies approximate randomization test in order to compare them. The running times of the commands given in Figure 1, 2, and 3 are approximately 18, 876, and 599 seconds, respectively.

Table 4: Performance on the test set

Model	Micro F Score	Macro F Score
Multinomial NB	0.932	0.664
Multivariate Bernoulli NB	0.851	0.499

References

[1] "UCI Machine Learning Repository: Reuters-21578 Text Categorization Collection Data Set — archive.ics.uci.edu," https://archive.ics.uci.edu/ml/datasets/reuters-21578+text+categorization+collection, [Accessed 04-May-2023].

```
elvankarasu@Elvans-Laptop text-classification-with-naive-bayes % python main.py ./reuters21578/ multinomial-nb

Multinomial Naive Bayes with alpha = 0.5. Performance on the dev set:
Macro F-score: 0.690
Micro F-score: 0.932

Multinomial Naive Bayes with alpha = 1. Performance on the dev set:
Macro F-score: 0.680
Micro F-score: 0.929

Multinomial Naive Bayes with alpha = 2. Performance on the dev set:
Macro F-score: 0.647
Micro F-score: 0.927

Multinomial Naive Bayes with alpha = 0.5. Performance on the test set:
Macro F-score: 0.664
Micro F-score: 0.932
```

Figure 1: Command for running the multinomial NB model

```
elvankarasu@Elvans-Laptop text-classification-with-naive-bayes % python main.py ./reuters21578/ bernoulli-nb

Multivariate Bernoulli Naive Bayes with alpha = 0.5. Performance on the dev set:
Macro F-score: 0.502
Micro F-score: 0.824

Multivariate Bernoulli Naive Bayes with alpha = 1. Performance on the dev set:
Macro F-score: 0.490
Micro F-score: 0.814

Multivariate Bernoulli Naive Bayes with alpha = 2. Performance on the dev set:
Macro F-score: 0.479
Micro F-score: 0.805

Multivariate Bernoulli Naive Bayes with alpha = 0.5. Performance on the test set:
Macro F-score: 0.499
Micro F-score: 0.851
```

Figure 2: Command for running the multivariate Bernoulli NB model

```
elvankarasu@Elvans-Laptop text-classification-with-naive-bayes % python main.py ./reuters21578/ all

Multinomial Naive Bayes with alpha = 0.5. Performance on the test set:
Macro F-score: 0.664
Micro F-score: 0.932

Multivariate Bernoulli Naive Bayes with alpha = 0.5. Performance on the test set:
Macro F-score: 0.499
Micro F-score: 0.851

Randomization test:
p: 0.0010 <= 0.05, hence we reject the null hyptohesis.
```

Figure 3: Command for running multinomial and multivariate NB models