Text Mining Assignment1

Jiameng Ma(4255445) Huishi Wang(4256875)

October 2024

Question 1

We utilize the `sklearn.datasets` library to load the 20 Newsgroups dataset, ensuring that we classify all 20 categories. The dataset is split into training and testing sets using an 80-20 split.

Question 2

In this part, we choose TF-IDF feature, use Naive Bayes, Logistic Regression and LinearSVC on this task. We split the test set by 20%. We can see from

Metric	Precision	Recall	F1-Score
Accuracy		0.88	
Macro Avg	0.89	0.87	0.87
Weighted Avg	0.89	0.88	0.87

Table 1: Naive Bayes

Metric	Precision	Recall	F1-Score
Accuracy		0.90	
Macro Avg	0.90	0.90	0.90
Weighted Avg	0.90	0.90	0.90

Table 2: Logistic Regression

Metric	Precision	Recall	F1-Score
Accuracy		0.93	
Macro Avg	0.93	0.93	0.93
Weighted Avg	0.93	0.93	0.93

Table 3: Linear SVC

the tables above that LinearSVC performs the best among these three models

as indicated by its higher accuracy, macro average, and weighted average for precision, recall, and F1-score. .

Question 3

Three feature extraction methods are evaluated:

1. **Count Vectorizer**: Converts text documents to a matrix of token counts.
2. **TF (Term Frequency)**: Normalizes word counts.
3. **TF-IDF (Term Frequency-Inverse Document Frequency)**: Weighs words based on their frequency across documents, reducing the importance of common words.

The results of the evaluations yield the precision, recall, and F1-score for each classifier-feature combination. The following table summarizes the results:

Classifier	Feature Type	Precision	Recall	Recall
Naive Bayes	Count	0.89	0.87	0.87
Naive Bayes	TF	0.86	0.84	0.83
Naive Bayes	TF-IDF	0.83	0.87	0.87
Logistic Regression	Count	0.89	0.89	0.89
Logistic Regression	TF	0.87	0.87	0.87
Logistic Regression	TF-IDF	0.90	0.90	0.90
Linear SVC	Count	0.89	0.89	0.89
Linear SVC	TF	0.91	0.91	0.91
Linear SVC	TF-IDF	0.93	0.93	0.93

Table 4: The results of the evaluations

These results mean that the best performance comes from LinearSVC with TF-IDF features on the highest precision, recall, and F1-score among all tested combinations. This also proves that the LinearSVC works well in the case of multi-class text classification when the TF-IDF feature extraction method is applied, giving emphasis on more informative words by reducing the weight of common terms.

Question 4

In this part, we focused on several key parameters: lowercasing, stop words, analyzer configurations, and max features. The results from these experiments help us evaluate the impact of these parameters on the performance of the Linear SVC classifier.

Configuration	**Lowercasing**	**Stop Words**	**Analyzer**	**N-gram Range**	**Max Features**
Vectorizer 1	True	English	Word	(1, 1)	5000
Vectorizer 2	False	None	Word	(1, 1)	5000
Vectorizer 3	True	English	Word	(2, 2)	5000
Vectorizer 4	True	None	Char	(3, 3)	5000
Vectorizer 5	True	None	Word	(1, 1)	10000

Table 5: Configurations of CountVectorizer

Configuration	**Accuracy**
Vectorizer 1	0.88
Vectorizer 2	0.87
Vectorizer 3	0.74
Vectorizer 4	0.86
Vectorizer 5	0.90

Table 6: Accuracy Results for CountVectorizer Configurations

According to the accuracy of these five combinitions, vectorizer5 has the best performance.