Jing Li (S5374553)

A Report about train and testing SVM models

This report is about training and testing SVM models on a dataset of 5860 messages with 5 unique labels for fine-grained classification and 3 unique labels for coarse-grained classification. The distribution of the messages across the fine-grained labels is as follows: class 3 has the highest count with 1607 messages, followed by class 1 with 1515 messages, class 2 with 1117 messages, class 4 with 860 messages, and class 0 with 761 messages. The distribution of the messages across the coarse-grained labels is as follows: class 1 has the highest count with 2467 messages, followed by class -1 with 2276 messages, and class 0 with 1117 messages. The average number of tokens per message in the training data is 19.17.

Using the dataset, we developed 3 different linear SVM models using a combination of word and character n-grams and Tfidf or Count vectorization methods. The models were trained on the train data and tested on both the dev and test data.

Model 1 used CountVectorizer for both word and character n-grams, Model 2 used TfidfVectorizer for both word and character n-grams, and Model 3 used CountVectorizer for word n-grams and TfidfVectorizer for character n-grams. For each model, a feature union was defined using the FeatureUnion() method to combine the features extracted from different vectorizers. Then, a pipeline was created using the Pipeline() method, which sequentially applied the feature union and the SVM classifier with a Linear Kernel. Finally, the pipeline was applied to the test set to predict the sentiment labels, and the accuracy, macro F1, and weighted F1 scores were calculated using the accuracy_score() and f1_score() functions.

	Model 1	**Model 2**	**Model 3**
Accuracy	0.5660633484162896	0.6357466063348416	0.5846153846153846
Macro F1	0.49254851081937323	0.518419290691118	0.5073185463780766
Weighted F1	0.5638487037126336	0.6059621637227968	0.5796806609875855

Table 1 Model Score

The results (Table 1) showed that model 2, which used TfidfVectorizer for both word and character n-grams, achieved the highest accuracy, macro F1, and weighted F1 scores on both the development and test datasets.

However, when the best model was used to predict the sentiment of the sentences in the file “DH_CollectingData2022_review”, it showed poor performance. The accuracy is 0.41, which means that only 41% of the predictions were correct. The macro average F1-score is 0.37, which indicates that the model is not performing well in predicting any of the classes. The weighted average F1-score is 0.39, which is slightly better but still indicates poor performance. The precision and recall values for each class were also quite low, indicating that the model was not able to correctly classify the comments into the three sentiment categories (-1, 0, 1).