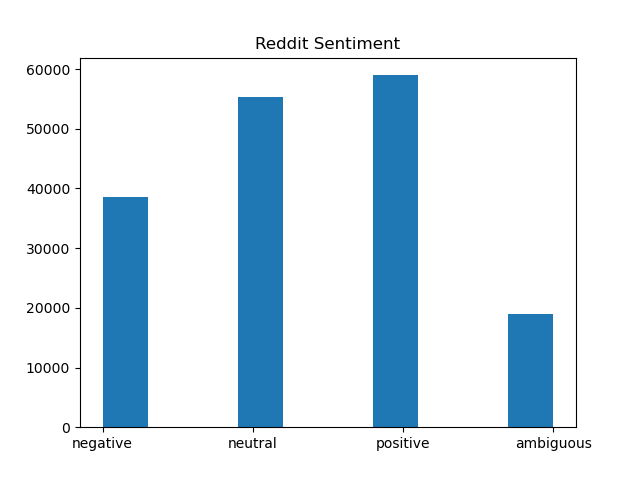
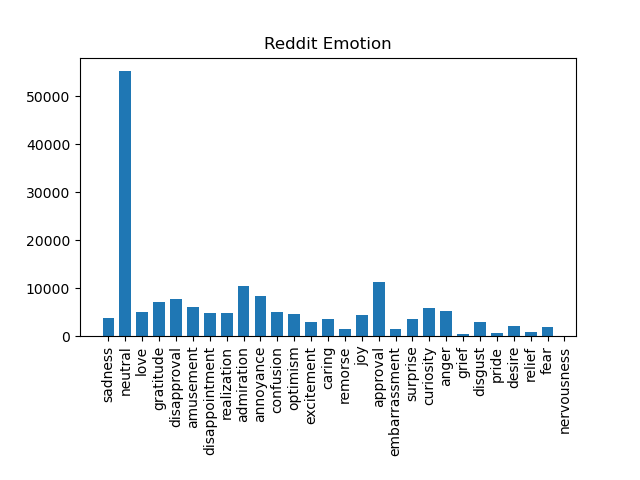
**4.1)**

The GoEmotions dataset given on Moodle contains around 58,000 Reddit comments labeled with one of the 28 emotions and one of the 4 sentiments. After plotting the distribution of the comments’ labels, we can see in the “Reddit Sentiment” plot below that there are fewer “ambiguous” comments than in the other sentiment categories. The rest of the comments are pretty well distributed in the three other sentiments.



This imbalance could impact the training process in a way that the classifier considers the “ambiguous” category to be less common. This bias could then impact the prediction process in a way that comments will have a smaller probability of being labeled as “ambiguous” even if they are categorized as so. In the “Reddit Emotion” plot below, we can see a huge imbalance in the emotions category.



There are excessively more comments labeled as “neutral” than as any other category. Since the classifier has been trained with way more neutral comments than with any other category, it could bias the prediction process to wrongfully label more comments as “neutral”. Due to this imbalance, not any metric can be used to measure the emotion classifier’s performance. Indeed, it will probably seem very well performing thanks to its high emotion accuracy as it predicts correctly for the majority class (“neutral”), but the F1-score, which balances prediction precision and recall, will be way lower. Hence, F1-score might be a better performance indicator for emotion prediction. In the case of sentiment prediction, F1-score will probably still be lower than the accuracy, but way closer. Hence, the choice of metric in that case might be less important than with emotion.

**4.2)**

An analysis of the results of all the models for both classification tasks. In particular, compare and contrast the performance of each model with one another, and with the datasets. Please note that your discussion must be analytical. This means that in addition to stating the facts (e.g. the macro-F1 has this value), you should also analyse them (i.e. explain why some metric seems more appropriate than another, or why your model did not do as well as expected.) Tables, graphs and contingency tables to back up your claims would be very welcome here.

Words as Features:

The first part of the mini project was to train various classifiers with words as features. Here’s a table of the various emotion classifiers’ performance:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Parameters | Accuracy | Macro-average F1 | Weighted-average F1 | Notes |
| Base-MNB | Default | 0.389 | 0.167 | 0.311 | Some classes don’t have predicted samples in the classification report. |
| Base-DT | Default | 0.360 | 0.278 | 0.361 |  |
| Base-MLP | Default | 0.373 | 0.288 | 0.373 | Hasn’t converged after 200 iterations. |
| Top-MNB | Alpha = 0.5 | 0.393 | 0.223 | 0.348 | Some classes don’t have predicted samples in the classification report. |
| Top-DT | Criterion = gini  Max depth = 4  Min samples split = 2 | 0.375 | 0.082 | 0.229 | Some classes don’t have predicted samples in the classification report. |
| Top-MLP | Activation = relu  Hidden layer sizes = [30, 50]  Max iter = 3  Solver = Adam | 0.435 | 0.275 | 0.388 | Hasn’t converged after 3 iterations. |

Table 1: Emotion classifier performance (words as features)

We can see that the Top-MLP classifier is the best one as it has the highest Accuracy and Weighted-average F1values. In the case of this highly imbalanced emotion dataset, Weighted-average F1-Score is the most important metric to consider. Indeed, the fact that it is weighted according to the individual sample sizes makes it more accurate to compare the actual performance with other models.

Here’s a table of the various sentiment classifiers’ performance:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Parameters | Accuracy | Macro-average F1 | Weighted-average F1 | Notes |
| Base-MNB | Default | 0.297 | 0.251 | 0.290 |  |
| Base-DT | Default | 0.274 | 0.250 | 0.278 |  |
| Base-MLP | Default | 0.268 | 0.252 | 0.286 | Hasn’t converged after 200 iterations. |
| Top-MNB | Alpha = 0.5 | 0.542 | 0.503 | 0.537 |  |
| Top-DT | Criterion = gini  Max depth = 4  Min samples split = 2 | 0.384 | 0.213 | 0.280 |  |
| Top-MLP | Activation = relu  Hidden layer sizes = [30, 50]  Max iter = 3  Solver = Adam | 0.572 | 0.538 | 0.569 | Hasn’t converged after 3 iterations. Also, some classes don’t have predicted samples in the classification report. |

Table 2: Sentiment classifier performance (words as features)

When experimenting with the same types of classifiers but trained with words as features with stop words removed, I was surprised to find out that the classifiers’ performance were exactly the same.

WHY??

Embeddings:

Here’s a table of the various sentiment classifiers’ performance:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Parameters | Accuracy | Macro-average F1 | Weighted-average F1 | Notes |
| Base-MLP | Default | 0.531 | 0.469 | 0.519 | Hasn’t converged after 3 iterations. |
| Top-MLP | Activation = relu  Hidden layer sizes = [30, 50]  Max iter = 3  Solver = Adam | 0.531 | 0.466 | 0.519 | Hasn’t converged after 3 iterations. |

Table 3: Sentiment classifier performance (words embeddings)

Here’s a table of the various emotion classifiers’ performance:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Parameters | Accuracy | Macro-average F1 | Weighted-average F1 | Notes |
| Base-MLP | Default | 0.402 | 0.169 | 0.304 | Hasn’t converged after 3 iterations. Also, some classes don’t have predicted samples in the classification report. |
| Top-MLP | Activation = relu  Hidden layer sizes = [30, 50]  Max iter = 3  Solver = Adam | 0.398 | 0.152 | 0.293 | Hasn’t converged after 3 iterations. Also, some classes don’t have predicted samples in the classification report. |

Table 4: Emotion classifier performance (words embeddings)

Here’s a table of the various emotion classifiers’ performance with other pre-trained embedding models:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Parameters | Accuracy | Macro-average F1 | Weighted-average F1 | Notes |
| Best MLP (glove-wiki-gigaword-200) | Activation = relu  Hidden layer sizes = [30, 50]  Max iter = 3  Solver = Adam | 0.365 | 0.1 | 0.25 | Hasn’t converged after 3 iterations. Also, some classes don’t have predicted samples in the classification report. |
| Best MLP (glove-twitter-50) | Activation = relu  Hidden layer sizes = [30, 50]  Max iter = 3  Solver = Adam | 0.352 | 0.08 | 0.23 | Hasn’t converged after 3 iterations. Also, some classes don’t have predicted samples in the classification report. |

Table 5: Emotion classifier performance (embeddings with other pre-trained models)

Here’s a table of the various sentiment classifiers’ performance with other pre-trained embedding models:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name | Parameters | Accuracy | Macro-average F1 | Weighted-average F1 | Notes |
| Best MLP (glove-wiki-gigaword-200) | Activation = relu  Hidden layer sizes = [30, 50]  Max iter = 3  Solver = Adam | 0.484 | 0.02 | 0.51 | Hasn’t converged after 3 iterations. Also, some classes don’t have predicted samples in the classification report. |
| Best MLP (glove-twitter-50) | Activation = relu  Hidden layer sizes = [30, 50]  Max iter = 3  Solver = Adam | 0.470 | 0.02 | 0.5 | Hasn’t converged after 3 iterations. Also, some classes don’t have predicted samples in the classification report. |

Table 6: Sentiment classifier performance (embeddings with other pre-trained models)

**4.3)**

As the sole member of my team, I was responsible for completing all the tasks in the assignment. Therefore, I contributed to all the work myself.