I start by splitting the dataset into a train and validation set and load the competition test set for final testing. The split of unrelated/related is approximately 3-to-1 which is quite imbalanced. Due to this, accuracy is not a suitable metric to quantify performance. I instead utilise F1-score, Matthews Correlation Coefficient, and AUC ROC as well as Confusion Matrices.

I experiment with a few different standard machine learning models. With the TF-IDF features I try Naïve Bayes and Complement Bayes as complement is supposed to fair better with imbalanced datasets. I also utilise Gradient Boost, Decision Trees, and Random Forest.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Technique | Dataset | Weighted F1 | MCC | AUC | Accuracy |
| Naïve Bayes | Validation | 0.650 | 0.115 | 0.547 | 0.691 |
|  | Test | 0.644 | 0.075 | 0.531 | 0.673 |
| Complement | Validation | 0.664 | 0.173 | 0.585 | 0.667 |
|  | Test | 0.582 | 0.086 | 0.547 | 0.590 |
| Gradient | Validation | **0.940** | **0.858** | **0.942** | **0.937** |
|  | Test | **0.682** | 0.226 | 0.617 | **0.677** |
| Gradient (C) \* | Validation | 0.920 | 0.821 | 0.934 | 0.917 |
|  | Test | 0.625 | **0.297** | **0.664** | 0.606 |
| Decision Tree | Validation | 0.731 | 0.407 | 0.721 | 0.720 |
|  | Test | 0.604 | 0.075 | 0.541 | 0.588 |
| RF | Validation | 0.782 | 0.515 | 0.778 | 0.773 |
|  | Test | 0.657 | 0.109 | 0.542 | 0.694 |

[Gradient (C) accounts for imbalance)

Gradient Boost performs best.

For transformers, using RoBERTa, I utilise Decision Trees, SVMs and Gradient Boost as Bayes does not allow for negative values. The results are:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Technique | Dataset | Weighted F1 | MCC | AUC | Accuracy |
| Decision Tree | Validation | 0.581 | 0.083 | 0.548 | 0.550 |
|  | Test | 0.579 | 0.093 | 0.552 | 0.556 |
| Gradient | Validation | 0.671 | 0.322 | 0.682 | 0.651 |
|  | Test | 0.703 | 0.292 | 0.655 | 0.694 |
| Gradient (C) | Validation | 0.657 | 0.342 | 0.693 | 0.637 |
|  | Test | 0.689 | 0.286 | 0.655 | 0.676 |

Again, Gradient Boost performs best and I find that accounting for the imbalance by weighting the classes leads to worse performance in terms of all metrics. Interestingly,

Ultimately, Gradient Boost performs best on the validation set but generally they all perform poorly on the test set.