**Text Classification Results Summary**

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Feature Extractor** | **Accuracy** | **F1-Score** |
| Multinomial Naive Bayes | CountVectorizer | 0.801 | 0.801 |
| Logistic Regression | CountVectorizer | 0.771 | 0.771 |
| Support Vector Machine | CountVectorizer | 0.501 | 0.470 |
| Decision Tree | CountVectorizer | 0.600 | 0.604 |
| Multinomial Naive Bayes | TfidTransformer | 0.749 | 0.695 |
| Regression | TfidTransformer | 0.811 | 0.803 |
| Support Vector Machine | TfidTransformer | 0.779 | 0.772 |
| Decision Tree | TfidTransformer | 0.582 | 0.573 |
| Logistic Regression | Word2Vec | 0.484 | 0.437 |
| Support Vector Machine | Word2Vec | 0.428 | 0.382 |
| Decision Tree | Word2Vec | 0.373 | 0.372 |
| Logistic Regression | Doc2Vec | 0.676 | 0.669 |
| Support Vector Machine | Doc2Vec | 0.654 | 0.631 |
| Decision Tree | Doc2Vec | 0.501 | 0.497 |

This table summarizes the performance of different text classification algorithms with various feature extraction techniques. The results show a comparative analysis based on accuracy and F1-score.

### Best Algorithm and Feature Extraction

Based on the table, the best performing combination for this classification task is:

* **Algorithm**: Logistic Regression
* **Feature Extractor**: TfidfTransformer

This combination achieved the highest accuracy of **0.811** and an F1-score of **0.803**, making it the most effective approach for this specific problem.