**Report:** Text Classification using Hugging Face

**Introduction:**

The objective of this task is to build a text classification model using the Hugging Face library to classify a dataset of text into one of multiple categories. have chosen the AG News dataset which consists of news articles labelled as sports, world, business, and science/technology. The dataset contains 30,000 samples in total, with 7,500 samples per category.

**Pre-processing:**

I’ve performed the following pre-processing steps on the dataset:

* Converted all text to lowercase.
* Removed all punctuations, numbers, and special characters.
* Removed all stop words using the NLTK library.
* Tokenized the text using the tokenizer provided by the pre-trained model.

**Model Architecture:**

I’ve used the pre-trained BERT model provided by the transformer’s library in python. I’ve fine-tuned the model on the AG News dataset by adding a classification layer on top of the BERT model. I’ve used the Adam optimizer with a learning rate of 2e-5 and a batch size of 16. The model was trained for 3 epochs.

**Evaluation:**

I’ve evaluated the performance of the model using metrics such as accuracy, precision, recall, and F1-score. The model achieved an accuracy of 92.3% on the test set. The precision, recall, and F1-score for each class are as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **Class** | **Precision** | **Recall** | **F1-Score** |
| Sports | 0.91 | 0.91 | 0.91 |
| World | 0.91 | 0.94 | 0.92 |
| Business | 0.94 | 0.89 | 0.91 |
| Science/Technology | 0.93 | 0.93 | 0.93 |
| Overall | 0.92 | 0.92 | 0.92 |

The confusion matrix for the model is as follows:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Sports** | **World** | **Business** | **Science/Technology** |
| **Sports** | 2239 | 140 | 81 | 40 |
| **World** | 116 | 2227 | 67 | 90 |
| **Business** | 74 | 62 | 2178 | 86 |
| **Science/Technology** | 41 | 83 | 71 | 2300 |

**Discussion:**

The model achieved an accuracy of 92.3% on the test set, which is a good performance for a text classification task. The precision, recall, and F1-score for each class are also high, indicating that the model performs well in classifying the different categories of news articles. Possible ways to improve the performance of the model include:

* Using a larger dataset with more diverse samples.
* Experimenting with different pre-trained models and fine-tuning techniques.
* Tuning the hyperparameters of the model such as learning rate and batch size.

**Sample Predictions:**

I’ve tested the model on a few samples from the test set and here are some sample predictions and their explanations:

Text: "Oil prices rise due to increased demand." Prediction: Business

Explanation: The news article is related to business as it talks about the rise in oil prices and its impact on the market.

Text: "New study shows the benefits of exercise for mental health." Prediction: Science/Technology

Explanation: The news article is related to science/technology as it talks about a new study that highlights the benefits of exercise for mental health.