U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Second Assignment

Text Classification Task

Made by Group 05:
Emanuel Maia - up202107486
Rita Leite - up202105309
Tiago Azevedo - up202108699

Contextualization

- The dataset comprises posts collected from Reddit and Google, authored by individuals from England, Australia, and India. It is organized into 12 distinct files, categorized by source (Reddit and Google), country (England, Australia, and India), and data type (training and testing).

- The files are in CSV format, and all content is in English. Each file contains three attributes: “id”, a unique identifier for each entry; “text”, the content of the post, which contains free text; and “sentiment_label”, the target variable for sentiment analysis, where “0” indicates negative sentiment and “1” indicates positive sentiment.

- When combined into a single file, the dataset contains a total of 10,078 entries.

- We also translated the datasets from India, as some of them were not in English.

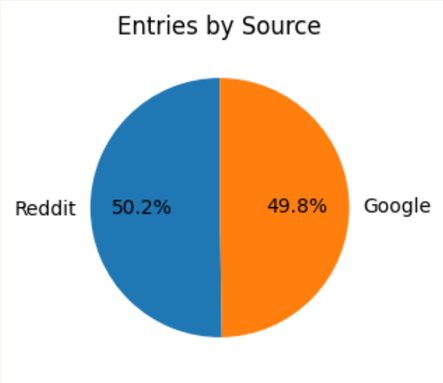

Figure 1 - Distribuition by source

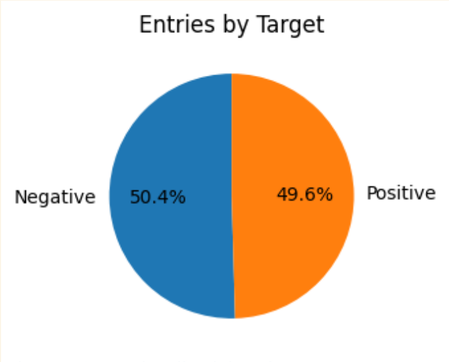

Figure 2 - Distribuition by target

Figure 4 - Wordcloud for Reddit Data

Figure 5 - Wordcloud for Google Data

Last Best Results

- **SGDClassifier using TfidfVectorizer(ngram_range = (1,2))**

- **Results in Global** - from the confusion matrix of our best model, it is evident that the majority of errors are instances where the model predicts a negative sentiment, whereas the true sentiment is positive.

  - Upon examining the entries that were misclassified, we observed that all of them contain words that are typically associated with a negative sentiment. These words seem to carry a stronger weight in the model's decision-making process.

	Predicted Negative	Predicted Positive
Real Negative	921	110
Real Positive	197	787

Tabel 1 - Confusion Matrix from the best model from Project 1

- **Results by Country** - although the dataset is balanced, the results indicate that texts from the United Kingdom and Australia yield better performance compared to those from India.

  - A particular challenge with entries from India is the presence of text that combines English with Hindi, Urdu, or Bengali. Due to this mixture, our translation method was unable to handle all cases effectively.

	Accuracy	Precision
UK	0.92	0.93
AU	0.87	0.89
IN	0.78	0.82

Tabel 2 - Accuracy and Precison by country using the best model from Project 1

Transformer-Based Models

Used without training in our dataset:

- **DistilBERT Base Uncased fine-tuned on SST-2**
  - As a distilled version of BERT, it retains much of the accuracy while being smaller and faster.
  - Specifically fine-tuned on the SST-2 dataset for sentiment analysis.
  - It has two labels: positive and negative.

→ Was also used after fine-tuning it on our own dataset.

- **TabularisAI Multilingual Sentiment Analysis**
  - A multilingual sentiment analysis model designed to handle text in multiple languages (including Hindi and Bengali).
  - It has five labels: very negative, negative, neutral, positive, and very positive.

- **CardiffNLP Twitter RoBERTa Base Sentiment Latest**
  - A RoBERTa-based model fine-tuned specifically on a large collection of tweets for sentiment analysis.
  - Designed to handle informal, short-form text typical of social media platforms like Twitter.
  - It has three labels: negative, neutral, and positive.

- Because we also applied summarization, we used the model **sshleifer distilbart-cnn-12-6**, a distilled version of BART that produces text summaries efficiently. This model was chosen because it is lighter and faster than the original, which suited our computational resource limitations.

Transformer-Based Models

Used with training in our dataset:

- **DistilBERT Base Uncased fine-tuned on SST-2** (as said before)
  - The model was trained for 3 epochs using the Hugging Face Trainer API.
  - As part of the preprocessing, we applied text **truncation** to limit the input length. This choice was based on prior experiments, where truncation proved to be faster than summarization while yielding comparable classification performance.

Used with prompting:

- **Google/flan-t5-small**
  - This model is a lightweight, instruction-tuned version of T5, capable of performing a wide range of tasks via natural language prompts, without the need for fine-tuning. The truncation of text was also applied.
  - The generated response was interpreted as either positive or negative, based on keyword matching.

Results

Results without training:

- To perform our classification task, we **needed to reduce the size of our textual features**, as some of them were considerably long. To address this, we experimented with two different techniques: truncation and summarization.
  - **Truncation** - we truncated the textual attributes, keeping only the first 100 tokens.
  - **Summarization** - for the text attributes exceeding 100 tokens, we applied summarization using the sshleifer distilbart-cnn-12-6 model. More time consuming and similar results.

Similar results between the two models. For fine tunning we choose the first one

Despite being multilingual and theoretically capable of handling languages like Hindi and Bengali, it produced the **worst results** among all the models tested.

	Truncation		Summarization	
	Accuracy	Precision	Accuracy	Precision
DistilBERT Base Uncased fine-tuned on SST-2	0.79	0.80	0.79	0.79
CardiffNLP Twitter RoBERTa Base Sentiment Latest	0.77	0.81	0.77	0.81
TabularisAI Multilingual Sentiment Analysis	0.70	0.76	0.70	0.76

Tabel 3 - Accuracy and Precison by technique used (Truncation or Summarization) and model, without fine-tunning

Results

Results with training:

- We fine-tuned our model to improve its performance on our specific dataset. For this, we trained it for 3 epochs.
- It was the one that produced the best results in all metrics.

Results with prompting:

- We experimented using text-to-text generation models, and used the following prompt:
  - **"Classify the following text as positive or negative: '{text}'. Pay attention because the text, although in principle in English, may also be or have parts in Hindi, Urdu or Bengali. Try to be as accurate as you can."**
- This prompt was designed to highlight a key issue we identified in previous work, that our textual data, while primarily in English, often contains segments in other languages such as Hindi, Urdu, or Bengali.

		Training Loss	Validation Loss	Accuracy
DistilBERT Base Uncased fine-tuned on SST-2	Epoch 1	0.36	0.35	0.86
	Epoch 2	0.23	0.36	0.87
	Epoch 3	0.15	0.51	0.86

Tabel 4 - Accuracy, Training Loss and Validation Loss by epoch and model, with fine-tunning

	Accuracy	Precision
Google Flan-t5-Small	0.82	0.82

Tabel 5 - Accuracy and Precision Loss by model, with prompting

Error Analysis

- **Best Model in Project 2:** DistilBERT Base Uncased fine-tuned on SST-2 with fine-tunning.

- **Comparation with Project 1:** The results are still worse than those of the best model obtained in project 1.

- **Same Problem:** Even after fine-tuning the model on country-specific datasets, it is evident that the model struggles to adapt effectively to the dataset from India.

  - This may be due to the use of region-specific expressions that are uncommon in standard English, or due to portions of the text being written in other languages such as Hindi, Urdu, or Bengali.

		Training Loss	Validation Loss	Accuracy
DistilBERT Base Uncased fine-tuned on SST-2	Epoch 1	0.36	0.35	0.86
	Epoch 2	0.23	0.36	0.87
	Epoch 3	0.15	0.51	0.86

Tabel 6 - Accuracy, Training Loss and Validatio Loss by epoch and model, with fine-tunning

		Accuracy
Using the dataset from...	Australia	0.84
	Unites Kingdom	0.91
	India	0.81

Tabel 7 - Accuracy by country and model, with fine-tunning

Conclusions

We successfully leveraged **Hugging Face transformers** to tackle the same binary classification task as in the first assignment.

Among the models explored, a **fine-tuned binary text classification model trained on our dataset** achieved the best overall performance.

We also experimented with **prompt-based classification** using the google/flan-t5-small model. While it outperformed other models tested without fine-tuning, it still fell short of the results obtained through task-specific fine-tuning.

Language identification remained a key challenge, particularly for texts written in Hindi, Urdu, or Bengali. These cases contributed to classification errors in both traditional and prompt-based approaches.

Overall, fine-tuning a domain-relevant pre-trained model with task-specific data proved to be the most effective approach for our scenario.