U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

First Assignment

Text Classification Task

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

Data Provenance and Characteristics

- 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.

Data Analysis

- **Figures 1 and 2** - when combining all the datasets, the data is balanced both in terms of its origin and its target value.

- **Figure 3** - shows the distribution of the target class in different data sources. We can see that the Reddit dataset is predominantly negative, while the Google dataset is mainly positive.

- **Figures 4 and 5** - display word clouds featuring the words with the highest TF-IDF scores in the Reddit and Google datasets. Notably, the Google dataset contains more words with a positive connotation, such as "good", "nice", and "great". In contrast, these words appear less frequently in the Reddit dataset, which aligns with the findings from Figure 1.

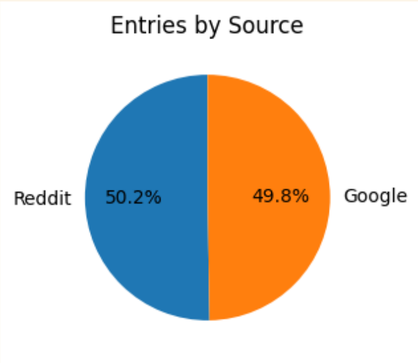

Figure 1 - Distribuition by source

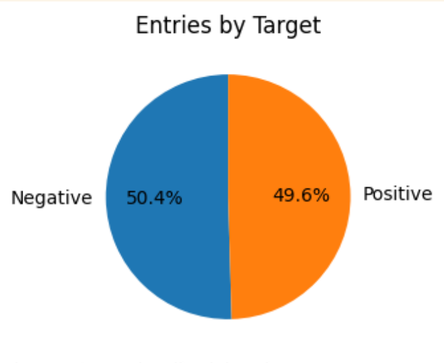

Figure 2 - Distribuition by target

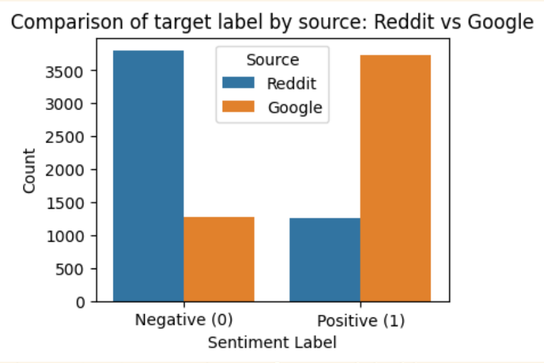

Figure 3 - Comparison of target label by source

Figure 4 - Wordcloud for Reddit Data

Figure 5 - Wordcloud for Google Data

Data Analysis

- **Figures 6 and 7** - show that the data is almost equally split by country of origin. Furthermore, when examining the distribution of the target class, there appears to be no significant difference between the datasets from different countries, indicating a balanced representation across the origin and target categories.

- **Figures 8, 9 and 10** - display word clouds showcasing the words with the highest TF-IDF scores in the United Kingdom (UK), Australia (AU), and India (IN) datasets. As observed, there are no significant differences in the most frequent terms across these datasets, indicating a similar pattern in the key words used in each country.

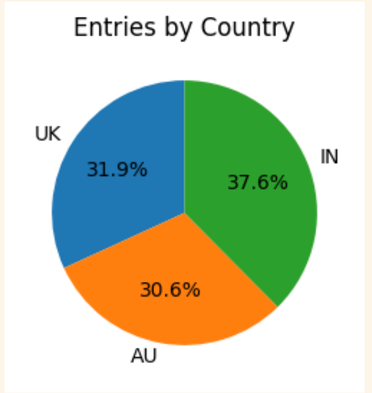

Figure 6 - Distribuition by country

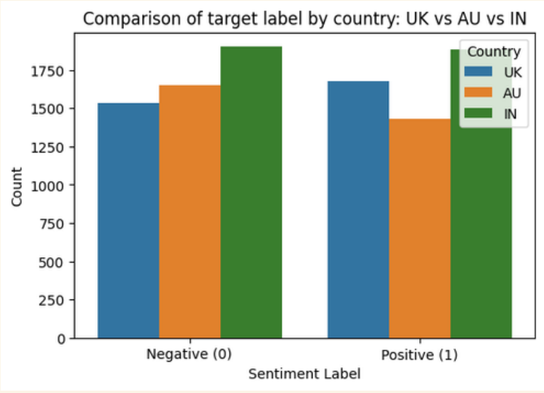

Figure 7 - Comparison of target label by country

Figure 8 - Wordcloud for data from UK

Figure 9 - Wordcloud for data from AU

Figure 10 - Wordcloud for data from IN

Data Pre-Processing

- We merged the data from 12 files into a single dataset, resulting in a total of **10,078 entries**.
- We created a new attribute called text_processed, derived from the text attribute through the following transformations:
  - **Expand contractions** in text
  - **Removed** all non-alphabetic characters while preserving spaces.
  - **Converted** all text to lowercase.
  - **Replaced** multiple consecutive spaces with a single space.
  - Applied **tokenization**.
  - Performed **lemmatization** with **part-of-speech tagging**.
  - Removed **stopwords**, except those with a negation meaning (e.g., "no", "not", "nor") to maintain contextual integrity.
- We removed the entries that after this process ended up with the empty text attribute.

Feature Representation

	Sparse Vectors	Feature Space
1	CountVectorizer()	19 469
2	TfidfVectorizer(ngram_range = (1,1))	19 469
3	TfidfVectorizer(ngram_range = (1,2))	204 175
4	TfidfVectorizer(ngram_range = (2,2))	184 706

	Vader
1	**Simpler**, calculates the polarity of each word and classifies it as positive or negative
2	**Complex**, algorithm inspired by one from the professor's slides, considers additional factors beyond polarity.

	Dense Vectors
1	Embeddings trained on our dataset, where sentence embeddings are obtained by concatenating the embeddings of the first 10 words.
2	Embeddings trained on our dataset, where sentence embeddings are obtained by averaging the word embeddings.
3	Pre-trained embeddings, where sentence embeddings are obtained by concatenating the embeddings of the first 10 words.
4	Pre-trained embeddings, where sentence embeddings are obtained by averaging the word embeddings.

Experimental Results

	MultinomialNB			LogisticRegression		
Features	**Accuracy**	**Precision**	**Time**	**Accuracy**	**Precision**	**Time**
CountVectorizer(binary=True)	0.81	0.79	0.01			
CountVectorizer()	0.81	0.78	0.00	0.83	0.85	0.09
TfidfVectorizer(ngram_range = (1,1))	0.80	0.78	0.00	0.84	0.87	0.04
TfidfVectorizer(ngram_range = (1,2))	0.80	0.78	0.02	0.83	0.86	0.16
TfidfVectorizer(ngram_range = (2,2))	**0.78**	**0.77**	**0.02**			

Baseline Model

- Regardless of the type of features representation used, the results obtained with Naive Bayes remain consistently similar.
- Although Logistic Regression is computationally more expensive than Naive Bayes, it consistently achieves better classification performance. This suggests that it is better suited for handling the complexity of the dataset.

Experimental Results

	LogisticRegression			SGDClassifier		
Features	**Accuracy**	**Precision**	**Time**	**Accuracy**	**Precision**	**Time**
TfidfVectorizer(ngram_range = (1,1))	0.84	0.87	0.04	0.84	0.88	0.05
TfidfVectorizer(ngram_range = (1,2))	0.83	0.86	0.16	**0.85**	**0.88**	**0.04**
Own embeddings (concat)	**0.77**	**0.80**	**5.49**			
Own embeddings (average)	0.81	0.82	0.48	0.80	0.77	0.09
Pre-trained embeddings (concat)	**0.79**	**0.79**	**0.08**			
Pre-trained embeddings (average)	0.82	0.81	0.11	0.80	0.75	0.05
Vader (complex approach)	**0.76**	**0.74**	**0.01**			

Best Model

- **Concatenation and Average for Embeddings** - the texts' size varies a lot. Concatenating only the first 10 words can lead to a loss of contextual information, and increasing the number of words would result in many sentences having embeddings padded with a large number of zeros, which may negatively impact performance.
- We also tried other classifiers, like **Support Vector Classifier** and **Random Forest Classifier**, but we did not obtain better results compared to our best model shown in the table.

Error Analysis

- **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.

  - Example where the model predicted negative but was positive: "please contact **safe** transport victoria big **issue** melbourne promptly take **issue** seriously **excellent** proof"

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

- **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

Conclusions

- Managed to successfully implement a **sentiment classification model** using data from Reddit and Google across different regions.
- By performing **exploratory data analysis**, we gained a better understanding of data distribution and key characteristics.
- Applied **preprocessing techniques** to clean up and prepare the text for classification.
- Evaluated the performance of different **machine learning models**, finally selecting the model based on the *SGDClassifier*, a decision based on accuracy and other metrics.
- Found that **sentiment variations exist across different regions**, possibly due to cultural and linguistic differences.
- Future work could revolve around the **expansion of the dataset**, both in size and variety or the integration of **more nuanced sentiment categories**.