Sentiment Analysis

Arda Derbent, Anton Yahorau
Group 21

April 2023

Preliminary Report

The preliminary report should be a PDF file containing the following information:

- Description of the task that you are working on:
  - What is the goal of the project and how do you frame the task you want to solve? (Regression, binary classification, multilabel classification)
  - Goal of the project is to create a Sentiment analysis project that should classify the emotion from its given tweet to positive negative and neutral. It should experiment on preprocessing and optimization to understand which methods give the best results finally apply this algorithm on another dataset. SVM will be used for multiclass classification and additionally neural networks will be used.
- Description of the dataset:
  - Does it require any preprocessing or cleaning?
  - It does require preprocessing. Firstly only the tweet text and sentiment columns will be used to make it usable across other datasets. From each tweet, punctuation marks will be removed, replaced with spaces. @users and stopwords will be removed. After this preprocessing, various techniques will be experimented on, to understand the affects of tokenization and similar methods.
  - Are the labels in the dataset balanced? How do you plan to address the data imbalance in the project?
  - In our dataset, negative revviews are much higher compared to positive and neutral. Neutral also being higher than positive. To fix this imbalance, we will undersample the negative and neutral sentiments to same number as positive sentiments present in the dataset. We can later visualize, the positive impact of this method if any.

  - How do you plan to split the data for the project?
  - We'll use the default method of 0.8 split for training, 0.2 split for training. We'll use random state=42.

- How would you evaluate the performance of your solution? Consider both the metrics and data splits, and look for an upper bound of your solution. For example, what is the best score on Kaggle?

- We'll be comparing ourselves to the 2nd most voted solution in kaggle on precision, recall, f1-score. here are their results:

```
    precision    recall  f1-score   support

    negative       0.94      0.94      0.94      2323
    positive       0.77      0.76      0.76       563

    accuracy                           0.91      2886
   macro avg       0.85      0.85      0.85      2886
weighted avg       0.91      0.91      0.91      2886
```

- Description of your solution:
  - Describe briefly usual methods used for the problem
  - Binary, Multiclass classification algorithms, neural networks and pretrained deep learning models can be used in sentiment analysis. Pretrained models require the heaviest computatation power however it is also the most accurate.
  - Select two or three of them that you want to implement and compare. Add more detailed description of those methods and data preprocessing required for them. Generally, NLP algorithms require some transformation from words into numeric representation (e.g., tokenization). Consider how the preprocessing applies to your dataset - how many different words appear in your dataset, and how long is the usual sequence you will be working on, etc.
  - We will implement SVM for multiclass classification and a neural network. SVM classification algorithm will be used on the obtained data. After cleaning the tweets, we'll tokenize it and then normalize it. After that we can feed it to CountVectorizer, the obtained vectors can in turn be fed into SVM algorithm that is present in the scikit-learn library. While implementing the Neural networks pytorch library and the output obtained from CountVectorizer will be used. In the balanced dataset there is currently 7996 unique words and 9.4 average word length per tweet after stopword, punctuation removal. This is after balancing the dataset. We'll tune it if more data will lead to better results.