SPECIAL COURSE IN SOFTWARE ENGINEERING, AUTUMN 2024

EXERCISE 5 – SENTIMENT ANALYSIS

In this exercise, you will write program to conduct sentiment analysis using ***VADER, TextBlob,*** and ***DistilBERT***.

THIS EXERCISE WILL BE GRADED (MAX: 7 points)

SUBMISSION DEADLINE: OCTOBER 18, 2024, MIDNIGHT

Task 1. main()

Just like all the main functions so far, this will first create an object of your class **SentimentAnalysis**.

Then, it will ask the user for the path to the data. If the dataset is inside the project root folder, then the user can just type the filename. For example, ***data.csv.*** Otherwise, the user must provide the entire path to the data file.

The main function should then call the **load_data()** function.

Next, the **get_text_columns()** function is called. This function will return a ***DataFrame*** that contains only the text columns. This ***DataFrame*** will contain the column name, average length of every entry in the column, and number of unique text entries in every column. More information on this function is available later in the document.

The ***DataFrame*** from the **get_text_columns()** is displayed to the user. Then the user is asked to type the column name that should be analysed.

Next, the user should be asked to choose the type of sentiment analysis to be done as follows:

1. VADER
2. TextBlob
3. DistilBert

If the user chooses '**1**', **vader_sentiment_analysis()** function is called. The function will return the polarity score and the sentiments. These should be stored in a ***DataFrame***, which must be displayed after the function is successfully run.

If the user chooses '**2**', **textblob_sentiment_analysis()** function is called. The function will return the polarity score, sentiments, and subjectivity score. These should be stored in a ***DataFrame***, which must be displayed after the function is successfully run.

If the user chooses '**3**', **distilbert_sentiment_analysis()** function is called. The function will return the score and sentiments. These should be stored in a ***DataFrame***, which must be displayed after the function is successfully run.

SPECIAL COURSE IN SOFTWARE ENGINEERING, AUTUMN 2024

Task 2. load_data()

Helps load the dataset when called by the main function.

This function takes **path** as its parameter and returns the dataset as a ***DataFrame***.

Task 3. get_text_columns()

This function selects only the text columns from the dataset. For every text column in the dataset, it calculates the average length of every entry in that column and the total unique entries from that column.

The above information must be stored in another ***DataFrame***, which **MUST** have the column names as '**Column Name', 'Average Entry Length'** and '**Unique Entries**'.

The function will return the above **DataFrame**.

Task 4. vader_sentiment_analysi()

This function receives a single column as a parameter. It then applies the VADER sentiment analysis to every single entry in that column to calculate its **compound score** and the **sentiment.** The sentiment is '**positive'** if the score is >=0.05, '**negative'** if it is <=-0.05, or else it is '**neutral**'.

The function returns **scores** and **sentiments** as separate lists**.**

REMEMBER to do the following before you write the exercise code:

- `pip install nltk`
- `nltk.download('vader_lexicon')`

Task 5. textblob_sentiment_analysi()

This function receives a single column as a parameter. It then applies the TextBlob sentiment analysis to every single entry in that column to calculate its **polarity score, sentiment,** and **subjectivity score.** The sentiment is '**positive'** if the score is >0, '**neutral'** if it is equal to 0, or else it is '**negative**'.

The function returns **scores, sentiments,** and **subjectivity score** as separate lists**.**

REMEMBER to do the following before you write the exercise code:

- `pip install textblob`
- `nltk.download('punkt')`

Task 6. distilbert_sentiment_analysis ()

SPECIAL COURSE IN SOFTWARE ENGINEERING, AUTUMN 2024

This function receives a single column as a parameter. It then applies the DistilBERT pipeline to every single entry in that column.

Use **MUST** use the model **`nlptown/bert-base-multilingual-uncased-sentiment`**.

The **pipeline** will return a result with a '**label'** column and '**score**' column. If the **label** is '**4 stars**' or '**5 stars'**, the sentiment must be '**positive'**. If it is '**3 stars'**, the sentiment must be '**neutral'**. Otherwise, the sentiment must be '**negative'**.

REMEMBER to do the following before you write the exercise code:

- `pip install transformers`
- `pip install torch`