**Text Classification with BERT (ASAP Dataset)**

ASAP (Automated Student Assessment Prize) dataset comprised of eight essay sets. Each of the sets of essays was generated from a single prompt. Selected essays range from an average length of 150 to 550 words per response. Some of the essays are dependent upon source information and others are not. All responses were written by students ranging in grade levels from Grade 7 to Grade 10. All essays were hand graded and were double-scored. Each of the eight data sets has its own unique characteristics. The variability is intended to test the limits of your scoring engine's capabilities.

**For the ASAP classification dataset, please refer to the code link below:**

Link for Dataset: <https://www.kaggle.com/datasets/sainijagjit/bbc-dataset>

Link for prediction steps code:

**(A). Importing important packages and libraries:**

A picture containing text, font, screenshot, line

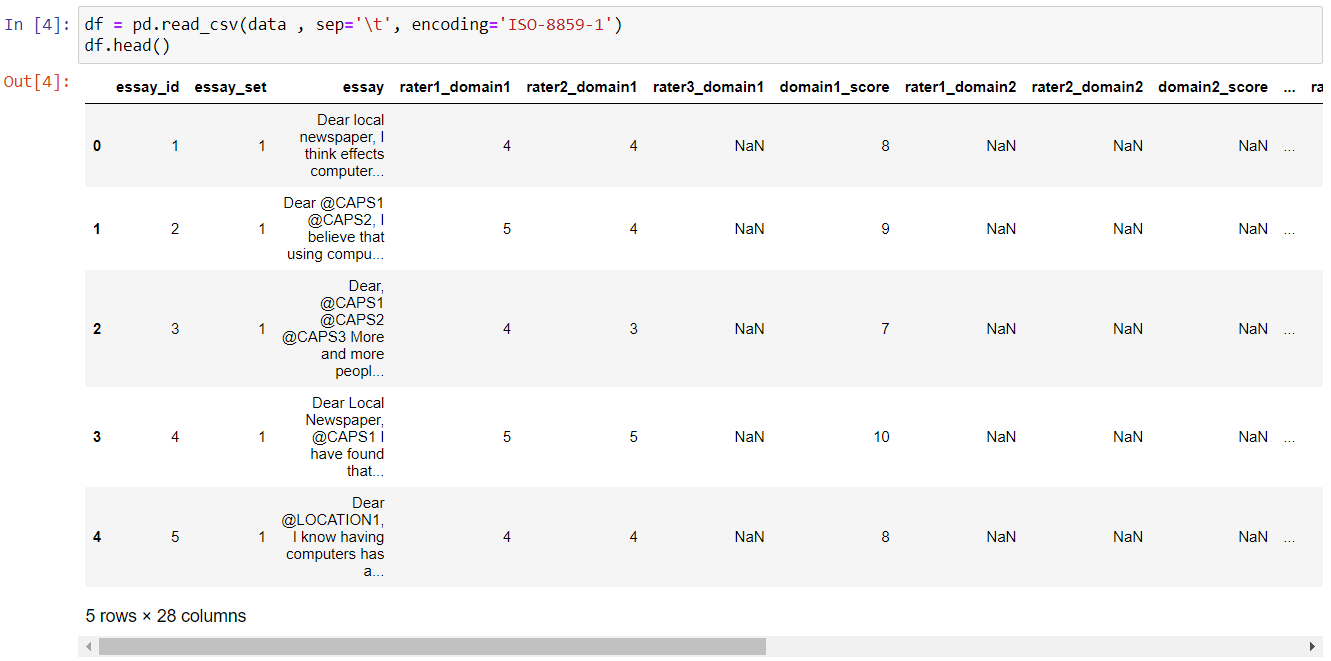
Description automatically generated

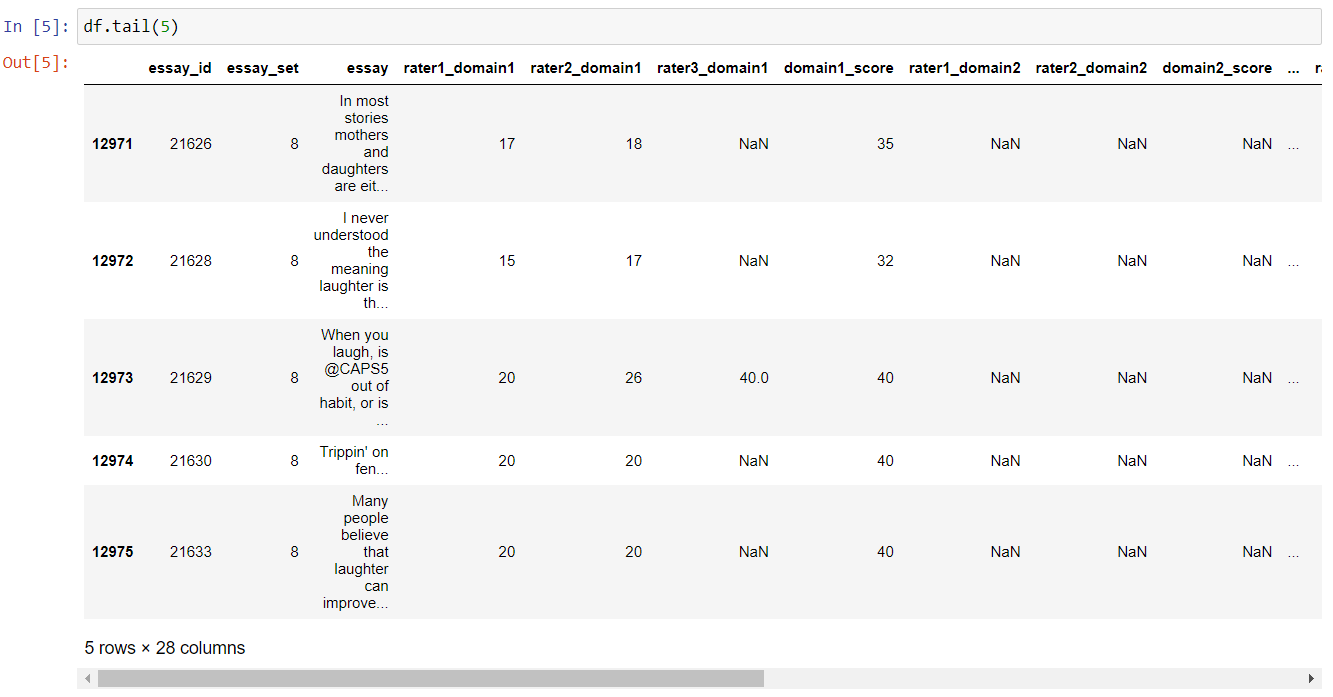
We have imported ‘**numpy**’ and ‘**pandas**’ for basic data manipulation in our project. We have imported ‘**pickle**’ for serializing and de-serializing a Python object structure. We have imported ‘**nltk**’ text processing libraries for classification, tokenization, stemming and stopwords.

**(B). Loading the given Dataset using google Colab pro:**

A screenshot of a computer

Description automatically generated with low confidence





A screenshot of a computer

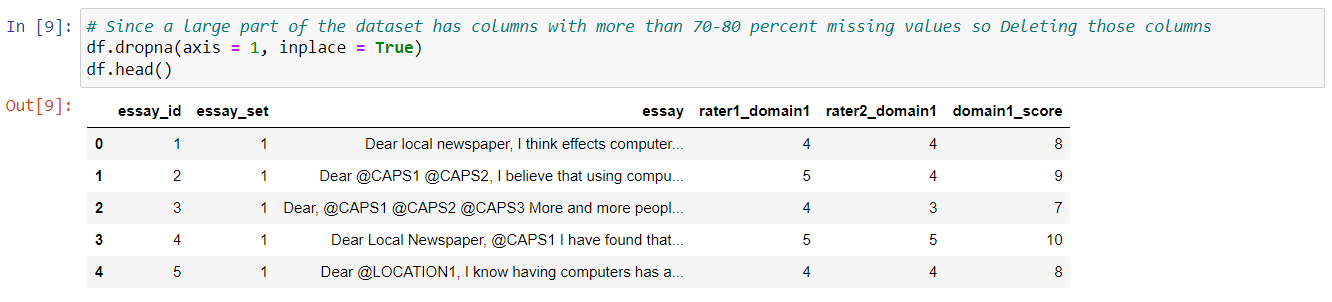
Description automatically generated with medium confidence



After uploading the dataset, we understood the data as it contains a total of **28 columns** and **12976 rows** with a lot of null/not assigned (NA) values and label as ‘**domain1\_score**’ for prediction.

A screenshot of a computer

Description automatically generated with low confidence



We have checked the number of null/NA values in each column and dropped the columns which only contained null values as they won’t impact on our model performance and result. Finally, we got 6 columns where we’ll be using ‘**essay**’ for **training** and ‘**domain1\_score**’ for **prediction.**

A screenshot of a computer

Description automatically generated with medium confidence

For easier understanding of the data, we renamed the ‘essay’ column as ‘text’ for training and ‘domain1\_score’ as ‘score’ for prediction.

A screenshot of a computer

Description automatically generated

For dataset understanding, we used the ‘**describe()**’ function and checked the unique number of essay sets used to make the overall essay text. Here, we can see the number of texts associated with each different essay set in the ASAP dataset.