109550198, 卜銳凱

# Data Mining
# HW2
# Report

## Introduction

The code is designed for setting up a text classification system using the BERT model from the Hugging Face transformers library, designed particularly for sentiment analysis based on text data. It includes comprehensive data preprocessing, where text is cleaned, tokenized, and formatted for BERT, alongside a custom PyTorch Dataset class for efficient data handling. The code defines a custom BERT-based Neural Network model, which is then trained and validated on split datasets. After training the model, the performance of this model is evaluated and predicts generated and saved, illustrating a practical application of sophisticated NLP techniques in analyzing and predicting sentiments from textual inputs.

1. **How do you select features for your model input, and what preprocessing did you perform to review text?**

   In this step, we try to effectively choose and pre-process the text features for our text classification model. The text data is concatenated and formatted by adding the titles to the main content, enhancing the context that can be crucial for sentiment analysis. Basic preprocessing is applied to clean the data—especially removing HTML tags, such as <br />, that might interfere with how accurately the model processes the natural language. This step serves to have the textual input in a clean and usable format for further tokenization that enables better model training and performance.

2. **Please describe how you tokenize your data, calculate the distribution of tokenized sequence length of the dataset and explain how you determine the padding size**

   Text will be converted to a machine-readable form using the BERT tokenizer. The tokenizer separates the text into tokens, generates IDs for these tokens, and produces attention masks to help the model focus on different parts of the text. The key parameter in this process is max_length, which sets the maximum length of the sequences to be fed into the model. Typically, this should be set based on the sequence length distribution; this is not directly analyzed in the provided code. Static max_length values are used in the code—1024 in one function and 256 in the other—that could be adapted to empirical data with the aim of finding a balance between padding and truncation.

3. **Please compare the impact of using different methods to prepare data for different rating categories**
   Data Preparation: Two sets of data are prepared:
   Title Only: Where the model is trained using only the titles of the reviews.
   Title + Text: Where both the titles and the main content of the reviews are used in training.

Model Training: Separate models are trained on each data set. The training process involves tokenizing the text, applying BERT-based neural networks, and optimizing the models using standard practices such as adjusting learning rates and epochs.
Performance Evaluation: Each model's performance is evaluated on a test set that is split from the original training data. This ensures that the evaluation is unbiased and reflective of the model's ability to generalize.
Results:
The results are expected to vary between different rating categories. For example:

Rating 1.0: The model trained on title only might perform poorly due to insufficient context, as negative reviews could require more detailed text to understand the sentiment fully.
Rating 3.0: In contrast, the title + text model may show improved accuracy for neutral ratings like 3.0, as the additional text provides more context that helps in more accurate sentiment classification.
This comparative analysis indicates that the choice of text components used for training significantly impacts the model's effectiveness in classifying different ratings. The inclusion of more textual content generally provides more detailed insights into the sentiment, improving accuracy for more complex or nuanced ratings. However, this might also introduce noise or irrelevant information, which could degrade performance for more straightforward, direct sentiments typically expressed in titles.

## Conclusion

Using BERT for text classification involves answering three critical questions about the intricate relationships between the selection of features and preprocessing, and the performance of models using different methods of data preparation. The first major focus is on the careful selection of text and preprocessing in order to optimize input for BERT models, including structuring data with the help of titles and main content. The second focus area presents the subtleties of tokenization and how finding an appropriate sequence length is critical to making the technical decisions affecting efficiency and efficacy in model training. Such preprocessing strategies affect the accuracy of the rating-specific models, underpinning the nuanced effects that the composition of text may have on the performance outcomes; hence, showing that bespoke approaches can considerably enhance the predictive accuracy of the model across various sentiment ratings.