NATURAL LANGUAGE PROCESSING: SENTIMENT ANALYSIS

"Given ChatGPT's ability to provide precise and rapid responses, it's essential for you to be both accurate and swift in your answers when I'm asking."

(1) **Description:** In this project, your objective is to construct a sentiment analysis model capable of distinguishing between negative (0) and positive (1) sentiments. The project will supply a CSV file containing 2,745 sentences, their corresponding labels, ID and source. These sentiments are aggregated from three datasets: YELP, IMDB, and AMAZON.

(2) **Project Objectives:**

- Acquire proficiency in using the SpaCy and the Scikit-Learn library.
- Grasp the steps involved in text preprocessing.
- Utilize various techniques to obtain sentiments embedding vectors.
- Become acquainted with the Linear Support Vector Classifier.
- Attain a comprehension of the concept of hyperparameter tuning.

(3) **Requirements:**

(a) **Data Exploration:**

- Begin by familiarizing yourself with the dataset.
- Examine the distribution of samples in each class.

(b) **Data Preprocessing:**

- Drop the ID and source columns.
- With SpaCy, eliminate stop words and perform lemmatization for each sentiment.
- Generate sentence embeddings using a chosen embedding technique, such as CountVectorizer, Tf-idf, or any other preferred method.
- Split into training and testing sets.

(c) **Classification and Comparison:**

- **Initial Experiment**: Implement sentiment classification using a Linear Support Vector Classifier (LinearSVC) with "Grid Search" to identify the optimal parameters for achieving the highest accuracy on the testing dataset.

- **Subsequent Experiment**: Use Artificial Neural Network (ANN) for classification. Explore different hyperparameters such as the number of neurons, learning rate, and batch size to enhance the performance of the ANN model.

- Provide a classification report for each experiment to comprehensively assess the performance of the models.

- Save the best model, then reload it in a separate file, and use it on new unlabeled data to get predictions.

Instructions:
1. The number of students in a team is 5.
2. No late submission is allowed.
3. Cheating students will take ZERO and no excuses will be accepted.
4. You can use any Python libraries.

Deliverables:

- You are required to submit ONE zip file containing the following:
  - Your code (.py) file. If you have a (.ipynb) file, you have to save/download it as (.py) before submitting.
  - A report (.pdf) containing the team members' names and IDs, and the code with screenshots of the output of each part. If you have a (.ipynb) file, you can just convert it to pdf.
- The zip file must follow this naming convention: ID1_ID2_ID3_ID4_ID5_Group

Grading Critera:

Data Exploration	1
Data Preprocessing	4
LinearSVC	2
NN	2
Comparison reports	2
Save, Load & Use Best Model	1

Note that those grades are not scaled, grade is out of 12

Date: December 2023