**Mini Project 2: Advanced Sentiment Analysis System Using RNNs**

**Pre-requisite: You shall read chapter 16 before starting this mini-project**

**Introduction**

In this mini project, you will implement a deep learning-based sentiment analysis system using Recurrent Neural Networks (RNNs), specifically Bidirectional LSTM layers, as introduced in Chapter 16 of *Python Machine Learning*. You will train, optimize, and evaluate your model on the IMDb movie review dataset. Model predictions will be interpreted using SHAP and LIME, and its carbon footprint will be measured using CodeCarbon.

You will also create a GitHub repository containing complete project documentation, licensing information, and best practices for responsible AI development and release. The final deliverable includes the notebook, supporting files, and a properly documented repository.

**Tasks and Goals**

**1. Data Preparation**

* **Goal:** Load and inspect the IMDb dataset of labeled movie reviews.
* **Task:** Download the dataset (<https://ai.stanford.edu/~amaas/data/sentiment/>), and prepare labels (positive = 1, negative = 0) for training and test sets.

**2. Text Preprocessing**

* **Goal:** Clean and normalize raw text reviews.
* **Task:** Convert text to lowercase, remove HTML tags and punctuation, remove stopwords, and pad sequences to a fixed length (e.g., 200).

**3. Tokenization and Sequence Vectorization**

* **Goal:** Convert preprocessed text into numerical format.
* **Task:** Use Keras Tokenizer to generate a word index and convert reviews into padded integer sequences.

**4. Model Design and Training**

* **Goal:** Train a deep neural network using Bidirectional LSTM layers.
* **Task:**
  + Use two stacked Bidirectional LSTM layers followed by Dense and Dropout.
  + Use EarlyStopping to prevent overfitting.

**5. Carbon Footprint Analysis**

* **Goal:** Track and report energy usage during training.
* **Task:** Use [CodeCarbon](https://mlco2.github.io/codecarbon/) to measure carbon emissions and display results in the notebook. You can find needed information here: <https://mlco2.github.io/codecarbon/> & <https://codecarbon.io/>

**6. Model Evaluation**

* **Goal:** Assess the performance of your trained model.
* **Task:** Report training and validation accuracy/loss. Test the model on new review examples and interpret the predictions. Additionally, test the trained model on a small set of manually selected review samples to validate its ability to generalize to real-world sentences. Example test inputs include:

sample\_texts = [

"The movie was absolutely fantastic and thrilling!",

"The movie was so bad and boring!",

"The movie had great visuals but the storyline was dull and predictable",

"The movie had great visuals. It was a wonderful movie",

"The plot was decent but the acting was terrible",

"Nothing special, just an average movie",

"An absolute masterpiece! Brilliant in every aspect",

"Awful. Just awful. I walked out after 30 minutes"]

**7. Hyperparameter Tuning**

* **Goal:** Improve model performance through optimization.
* **Task:** Experiment with learning rate, dropout rate, batch size, and sequence length.

**8. Explainability with SHAP and LIME**

* **Goal:** Visualize and explain individual predictions.
* **Task:** Use SHAP and LIME to analyze the impact of specific words on sentiment classification and interpret model behavior.

**9. Ethical Considerations**

* **Goal:** Reflect on responsible AI deployment.
* **Task:** Discuss model transparency, fairness, potential biases, and ethical concerns in using sentiment classification in real-world applications.

**10. Deployment Considerations**

* **Goal:** Optimize the model for embedded or mobile environments.
* **Task:** Convert the model to TensorFlow Lite and compare:
  + Model size
  + Inference time
  + Accuracy difference before/after conversion

**GitHub Repository and Documentation Requirements:**

You are required to create a **GitHub repository** for your project and include the following:

* **A detailed README** that includes:
  + Project overview and objectives
  + Instructions for dataset download and preprocessing
  + Model training and evaluation steps
  + Carbon footprint analysis results
  + Ethical considerations and explainability methods
  + Embedded system deployment insights
* **A section on code release responsibilities:**
  + Define your responsibility regarding source code release.
  + Justify the inclusion of an appropriate open-source license (e.g., MIT, Apache, GPL) and discuss its implications.
  + Compare different code release practices and how they impact transparency, collaboration, and software maintenance.
  + Outline a bug reporting and fixing protocol (e.g., issue tracking, pull requests).