📄 Assignment: Email Spam Classification using RNN, LSTM, and GRU

Problem Statement

In this assignment, you will build **Recurrent Neural Network (RNN) models** (SimpleRNN, LSTM, and GRU) to classify emails as **spam** or **ham (not spam)** using the *Email Spam Detection Dataset*.
The dataset contains labeled email messages, where each email is tagged as spam or ham.
Students will perform an end-to-end workflow: **Exploratory Data Analysis (EDA)**, **text preprocessing**, **feature extraction (tokenization + sequence padding)**, building baseline and optimized deep learning models (RNN, LSTM, GRU), and **evaluation**.
The goal is to understand how sequence models work for text classification and how to tune them.

Dataset Link

📎 [Email Spam Detection Dataset (Kaggle)]

Guidelines for Students

1. Data Understanding

- Download and inspect the dataset.
- Identify the number of spam vs ham emails.
- Look at email text length distributions (characters/words).
- Print a few sample emails from both categories.

2. EDA (Exploratory Data Analysis)

- Plot **class distribution** (spam vs ham counts).
- Visualize **word cloud** for spam and ham emails separately.
- Plot **histogram of email lengths** (number of words per email).
- Show common words (top n-grams like unigrams/bigrams) in spam vs ham.

3. Preprocessing

- Convert text to lowercase, remove punctuation, numbers, and HTML tags.
- Remove stopwords and apply tokenization.
- Use **Tokenizer + pad_sequences** (e.g., maxlen = 200).
- Split dataset into **training and validation sets** (e.g., 80/20).

4. Model Building

Baseline RNN

- SimpleRNN architecture with:
  `Embedding → SimpleRNN → Dropout → Dense (sigmoid)`
- Use **binary crossentropy loss**, **Adam optimizer**.
- Evaluate training & validation accuracy.

LSTM Model

- LSTM-based architecture:
  `Embedding → LSTM → Dropout → Dense (sigmoid)`
- Add callbacks such as **EarlyStopping**.
- Compare with RNN performance.

GRU Model

- GRU-based architecture:
  `Embedding → GRU → Dropout → Dense (sigmoid)`
- Experiment with stacked layers (e.g., GRU(128) → GRU(64)).
- Compare results with RNN and LSTM.

5. Evaluation

- Report **Accuracy, Precision, Recall, and F1-score**.
- Plot **Confusion Matrix**.
- Plot **Training vs Validation Accuracy/Loss** curves for all three models.
- Compare performance of RNN vs LSTM vs GRU in a summary table.
- Show **example predictions**: display the email text, true label, and predicted label.
- Analyze misclassified examples (e.g., why some ham emails were predicted as spam).

Expected Outcomes

- Students will learn to **explore and clean text datasets**.
- Students will gain skills in **tokenization, sequence padding, and embeddings**.
- Students will understand how **RNN, LSTM, and GRU** work for sequential text classification.
- Students will be able to **tune hyperparameters** (embedding size, units, dropout, sequence length).
- Students will learn to interpret results, compare models, and analyze misclassifications.