****

**Superior University Lahore**

***Lab Task # 12***

**Name:** Muhammad Abdullah Kashif

**Roll No**: SU92-BSAIM-F23-070

**Instructor**: Sir Rasikh Ali

# Course: Programming for Artificial Intelligence (Lab)

# Hadith Bot

**Step-by-Step Explanation of HadithBot Project**

**1. Introduction**

The HadithBot project is designed to provide responses related to Hadith using a chatbot interface. It utilizes **natural language processing (NLP)** and **transformer models** to understand and respond to user queries related to Hadiths from a dataset.

**2. Importing Required Libraries**

The project starts by importing essential Python libraries:

* transformers for using pre-trained models like BERT.
* torch for working with PyTorch tensors and models.
* pandas and numpy for data handling and numerical computations.
* sklearn for machine learning tasks.
* nltk for text preprocessing.

**3. Loading and Preprocessing Dataset**

A dataset containing Hadiths is loaded. The major steps include:

* Reading the CSV file using pandas.
* Extracting columns like **Hadith text** and **grade (authenticity)**.
* Cleaning and preprocessing the text to remove punctuation, lowercase everything, and eliminate stop words.

**4. Tokenization Using BERT Tokenizer**

The preprocessed Hadith texts are converted into tokens using BERT’s tokenizer (bert-base-uncased). This converts each sentence into a format understandable by the BERT model:

* Special tokens like [CLS] and [SEP] are added.
* Padding and truncation are applied to ensure uniform input length.

**5. Creating Attention Masks**

Attention masks are created to indicate which tokens are actual input and which are just padding:

* 1 for real tokens.
* 0 for padding.

**6. Splitting Dataset**

The dataset is split into:

* **Training set** (for model training).
* **Validation set** (for tuning and performance evaluation).

This is done using train\_test\_split from sklearn.

**7. Building the Model**

A **pre-trained BERT model** is loaded using BertForSequenceClassification:

* This model is fine-tuned for **binary classification** (e.g., authentic vs. weak Hadith).

**8. Creating DataLoaders**

The data is converted into PyTorch DataLoader format to efficiently train the model in batches:

* RandomSampler for training.
* SequentialSampler for validation.

**9. Model Training**

The BERT model is trained using:

* **AdamW optimizer** (recommended for transformers).
* A custom loop for multiple epochs.
* Gradients are clipped to avoid exploding gradients.

Loss and accuracy are calculated for both training and validation sets in each epoch.

**10. Evaluating the Model**

The trained model is evaluated on the validation set:

* The predictions are compared with actual labels.
* Accuracy and loss graphs are plotted to visualize performance over epochs.

**11. Creating the Chatbot Interface**

A basic chatbot interface is created using:

* A predict function that takes user input.
* Tokenizes and passes it to the trained model.
* Returns the prediction (Hadith grade).

The chatbot can now interact with users and classify the input text as authentic or weak Hadith.

# 