INNOMATICS RESEARCH LABS

# Next Word Prediction Project

This project enhances a wide range of skills, including technical abilities like **Python Programming**, **data preprocessing** and understanding of **DL concepts**. It build proficiency with **tools** and **frameworks (Tensorflow, Keras, Numpy, Pandas)**, model building and evaluation. This project helps in strengthening **essential skills** required for **Data Science.**

After the completion of the project add it to your profiles such as **LinkedIn, GitHub and CV/Resume which will put more weightage to your Resume and Digital Profiles.**

## 1. Problem Statement

Natural Language Processing (NLP) plays a crucial role in predictive text systems, chatbots, and virtual assistants. Next Word Prediction is a fundamental NLP task where a model predicts the most probable next word based on a given input text. The objective of this project is to develop a deep learning model using LSTM (Long Short-Term Memory) or GRU (Gated Recurrent Unit) to predict the next word in a sequence, improving text generation and user typing assistance.

## 2. Data Collection

To build an effective Next Word Prediction model, we need a large corpus of text data. The dataset can be obtained from:

- Publicly available text datasets like Wikipedia dumps, Common Crawl, or Project Gutenberg.
- Custom datasets from user inputs, chat transcripts, or open-source books.
- Kaggle datasets related to NLP text prediction.

The dataset should contain well-structured and diverse sentences to ensure a robust model.

**Note: Kindly get it reviewed the above steps with the @mentors before proceeding with the next step**

## 3. Data Preprocessing

Preprocessing is essential for improving model accuracy. The following steps will be performed:

- Text Cleaning: Removing special characters, numbers, and punctuation.
- Tokenization: Splitting text into words or subwords.
- Lowercasing: Converting all text to lowercase for uniformity.
- Stopword Removal (optional): Removing frequently used words that do not add significant meaning.
- Padding Sequences: Ensuring uniform input size by padding shorter sequences.
- Word Embedding: Converting words into numerical representations using techniques like Word2Vec, GloVe, or embedding layers in deep learning models.

**Note: Kindly get it reviewed the above steps with the @mentors before proceeding with the next step**

## 4. Model Building

The model will be built using LSTM or GRU networks for sequence prediction. The key steps include:

- Data Splitting: Dividing data into training and validation sets.
- Model Architecture: Creating an LSTM/GRU-based sequential model with layers including:
  - Embedding Layer: Converts words to vectors.
  - LSTM/GRU Layer: Captures temporal dependencies in text.
  - Dense Layer: Fully connected layer for output.
  - Softmax Activation: Predicts probabilities for the next word.
- Compilation & Training: Using loss functions like categorical cross-entropy and optimizers like Adam.

**Note: Kindly get it reviewed the above steps with the @mentors before proceeding with the next step**

## 5. Model Evaluation

To assess the model’s performance, the following evaluation metrics will be used:

- Accuracy: The proportion of correct predictions.
- Loss Analysis: Monitoring training and validation loss to avoid overfitting.
- Human Evaluation: Checking generated text quality for fluency and coherence.

**Note:** **Kindly get it reviewed the above steps with the @mentors for the Final review before you submit in the LMS and Update in the github or Resume**