# Week 6 Graded Mini Project: Large Language Models & Embeddings

## Weekly Outcomes Addressed:

- Demonstrate embedding model applications.
- Implement vector-based retrieval systems.
- Explain semantic similarity concepts.
- Assess the performance of vector search.

## Introduction

This Graded Project builds on your understanding of Large Language Models (LLMs) and embeddings. You will apply what you've learned to experiment with Hugging Face models for text generation and tokenisation, perform prompt engineering exercises, and use Gensim embeddings to compute similarity. The project aims to help you connect theoretical understanding of semantic representation with hands-on experience in implementing and evaluating vector-based approaches.

## Tasks Overview

**Objective:** Learners will apply their understanding of LLMs by experimenting with Hugging Face models for generation and tokenisation, and with Gensim embeddings for similarity.

### Section A: LLM Foundations & Hugging Face

1. **Hugging Face Setup & Text Generation**
    - Install Hugging Face `transformers`.
    - Load a small LLM (e.g., `distilgpt2`).
    - Generate three different continuations for the prompt: "AI is transforming industries by …".
    - **Deliverable:** Notebook cell with outputs.
2. **Tokenisation Demo**
    - Take the sentence: "LLMs are powerful tools for natural language understanding."
    - Tokenise it using a Hugging Face tokeniser.
    - Display the resulting tokens, token IDs, and sequence length.

### Section B: Prompt Engineering

3. **Prompt Tuning Challenge**
    - Design 3 prompts for different tasks:
        - Summarisation (limit to ≤ 30 words)

        - Q&A (e.g., answer a factual question)
        - Creative text generation (e.g., a 4-line poem on AI)
    - Compare the outputs generated by the Hugging Face LLM for each prompt.
    - **Reflection (100–150 words):** What changes did you observe when you rephrased prompts?

### Section C: Embeddings with Gensim

4. **Word Embeddings**
    - Use GloVe embeddings (`glove-wiki-gigaword-50`) via Gensim.
    - Select any three words (e.g., *king*, *queen*, *diamond*).
    - For each word, display:
        - First 10 values of its vector.
        - Top 5 most similar words and their similarity scores.
5. **Sentence-Level Embeddings**
    - Create five short sentences (e.g., on AI or jewellery).
    - Compute the average of word vectors for each sentence (a simple baseline for sentence representation).
    - Show a similarity matrix using cosine similarity to compare the sentences.

### Section D: Application Exploration

6. **Transformer Application**
    - Select one task from the Hugging Face pipeline (translation, summarisation, or sentiment classification).
    - Run a demo using your own custom input.
    - Write a short reflection (~100 words) on how this application could be used in a business context or professional workflow.

## How to Approach the Project

This project combines **Large Language Models (LLMs)** and **embeddings** through guided experimentation. Each section builds your ability to generate text, engineer prompts, and compute semantic similarity using **Hugging Face** and **Gensim** libraries.

Start in **Section A** by exploring text generation with the Hugging Face model DistilGPT-2. Ensure the transformers library is available.

!pip install transformers --quiet

Then import and test the pipeline:

from transformers import pipeline # Load a small pre-trained text generation model

generator = pipeline("text-generation", model="distilgpt2") # Try generating text from a short prompt

generator("AI is transforming industries by", max_length=40, num_return_sequences=1)

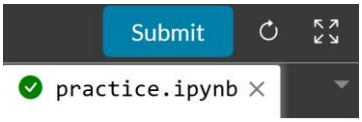
Run similar small tests to confirm your setup before moving to prompt variation and tokenisation. Remember, your goal is to observe how model outputs differ as you adjust prompts or parameters — not just to generate long text.

In **Section B**, experiment with phrasing across tasks (summarisation, Q&A, and creative text). Reflect on how prompt clarity influences responses. For **Section C**, explore **Gensim** embeddings — focus on interpreting similarity scores and what they reveal about relationships between words or sentences. Conclude with **Section D**, where you apply a Hugging Face pipeline (e.g., sentiment analysis or translation) and reflect on its practical business relevance.

Think of this project as a guided exploration — experiment freely, document what you discover, and relate technical results to meaningful insights about how LLMs and embeddings represent and retrieve information.

## Submission Guidelines

You must submit two components:

- **Jupyter Notebook:** Include complete code, outputs, and short explanations for all sections (A–D). After completing the project in Vocareum:
    1. Click on **Submit** in Vocareum.
        - 

    2. Go to the **Submissions** folder.
        - 

    3. Right-click your notebook file and select **Download** to save it locally.

- **PDF Summary:** Include:
    - Screenshots of key outputs.
    - What was learned from each activity.
    - Key observations about LLMs vs. embeddings.
    - Applications of vector search and embeddings in real-world scenarios.

Once ready, combine the notebook and PDF or upload them separately. Name your file as:

IITM PRAVARTAK
CATALYSING INNOVATION

**Week 6_Graded Project_[Your Name].zip or .pdf**

To submit on Canvas:

1. Click **Start Assignment** at the top of this page.
2. Upload your Jupyter Notebook and PDF summary.
3. Click **Submit Assignment** to finalise your submission.