



MARRI LAXMAN REDDY INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(AN AUTONOMOUS INSTITUTION)

(Approved by AICTE, New Delhi & Affiliated to JNTUH, Hyderabad)

Accredited by NAAC with 'A' Grade & Recognized Under Section 2(f) & 12(B) of the UGC act, 1956

COURSE CONTENT

PROMPT ENGINEERING								
III Semester: CSE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
2535822	Professional Core courses	L	T	P	C	CIA	SEE	Total
		3	0	0	3	40	60	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil			Total Classes: 45			
prerequisites: UG level course in prompt engineering								

Course Overview:

The Prompt Engineering course introduces the principles and best practices for designing effective prompts to guide generative AI models such as GPT-4, Gemini, and LLaMA. It explores transformer architecture, tokenization, structured output generation (JSON, YAML, CSV), and advanced techniques like chunking and format control using tools such as SpaCy and Tiktoken

Course Objectives:

1. To introduce the principles and techniques of effective prompt engineering for generative AI models.
2. To understand the architecture, capabilities, and evolution of large language models such as GPT-3.5, GPT-4, Gemini, and LLaMA.
3. To explore standard practices in structured and unstructured text generation using tools like ChatGPT.
4. To apply chunking, tokenization, and formatting techniques for improving text generation and manipulation.
5. To understand the role of embeddings, vector databases (FAISS, Pinecone), and Retrieval-Augmented Generation (RAG) in modern NLP systems

Course Outcomes: After Completion of the Course, Students should be able to

1. Explain and apply the core principles of prompt engineering for guiding generative AI outputs effectively.
2. Describe the underlying architecture and functionality of state-of-the-art large language models (LLMs).
3. Generate and manipulate structured outputs (JSON, YAML, CSV) using ChatGPT with advanced prompting techniques.
4. Implement text chunking, tokenization, and format control using tools like SpaCy, Tiktoken, and Python.
5. Utilize vector databases such as FAISS and Pinecone in Retrieval-Augmented Generation (RAG) pipelines for efficient information retrieval

Unit I

Fundamentals and Principles of Prompting

Overview of the Five Principles of Prompting: Give Direction, Specify Format, Provide Examples, Evaluate Quality, Divide Labor

Unit II

Introduction to Large Language Models for Text Generation

What Are Text Generation Models, Vector Representations: The Numerical Essence of Language, Transformer Architecture: Orchestrating Contextual Relationships, Probabilistic Text Generation: The Decision Mechanism, Historical Underpinnings: The Rise of Transformer Architectures, OpenAI's Generative Pretrained Transformers, GPT-3.5-turbo and ChatGPT, GPT-4, Google's Gemini, Meta's Llama and Open Source.

Unit III

Standard Practices for Text Generation with ChatGPT- Part-A

Generating Lists, Hierarchical List Generation, When to Avoid Using Regular Expressions, Generating JSON, YAML Filtering YAML Payloads, Handling Invalid Payloads in YAML, Diverse Format Generation with ChatGPT, Mock CSV Data, Universal Translation Through LLMs, Ask for Context, Text Style Unbundling, Identifying the Desired Textual Features, Generating New Content with the Extracted Features, Extracting Specific Textual Features with LLMs.

Unit IV

Standard Practices for Text Generation with ChatGPT- Part-B

Chunking Text, Benefits of Chunking Text, Scenarios for Chunking Text, Poor Chunking Example, Chunking Strategies, Sentence Detection Using SpaCy, building a Simple Chunking Algorithm in Python, Sliding Window Chunking, Text Chunking Packages, Text Chunking with Tiktoken, Encodings, Understanding the Tokenization of Strings

Unit V

Vector Databases with FAISS and Pinecone

Retrieval Augmented Generation (RAG), Introducing Embeddings, Document Loading

Memory Retrieval with FAISS, RAG with Lang Chain, Hosted Vector Databases with Pinecone, Self-Querying, Alternative Retrieval Mechanisms.

TEXT BOOKS:

1. Phoenix J, Taylor M. Prompt engineering for generative AI. " O'Reilly Media, Inc."; 2024 May 16.

REFERENCE BOOKS:

1. Tunstall L, Von Werra L, Wolf T. Natural language processing with transformers. " O'Reilly Media, Inc."; 2022 Jan 26.
2. Foster D. Generative deep learning. " O'Reilly Media, Inc."; 2022 Jun 28.

ELECTRONIC RESOURCES:

1. <https://learnprompting.org/>
2. <https://www.deeplearning.ai/short-courses/chatgpt-prompt-engineering-for-developers/>
3. <https://github.com/openai/openai-quickstart-python>
4. <https://www.coursera.org/learn/prompt-engineering>

MATERIALS ONLINE:

1. Course template
2. Tutorial question bank
3. Tech talk and Concept Video topics
4. Open-ended experiments
5. Definitions and terminology
6. Assignments
7. Model question paper – I
8. Model question paper – II
9. Lecture notes
10. E-Learning Readiness Videos (ELRV)

