COURSE MODULE OF THE SUBJECT TAUGHT FOR THE SESSION 2025-26 (EVEN SEM)

Course Syllabus with CO's

Faculty Name: Ms. Apoorva S M			**Academic Year: 2025 - 2026**				
Department: Computer Science & Engineering - AIML							
Course Code	**Course Title**	**Core/Elective**	**Prerequisite**	**Contact Hours**			**Total Hrs/ Sessions**
				L	**T**	**P**	
BAIL657C	Generative AI	AEC	C Programming Concepts	-	-	1	14
Course Objective	This laboratory course enables students to get practical experience in design, develop, implement, analyze and evaluation/testing of • Understand the principles and concepts behind generative AI models • Explain the knowledge gained to implement generative models using Prompt design frameworks. • Apply various Generative AI applications for increasing productivity. • Develop Large Language Model-based Apps.						

Topics Covered as per Syllabus

1. Explore pre-trained word vectors. Explore word relationships using vector arithmetic. Perform arithmetic operations and analyze results.
2. Use dimensionality reduction (e.g., PCA or t-SNE) to visualize word embeddings for Q 1. Select 10 words from a specific domain (e.g., sports, technology) and visualize their embeddings. Analyze clusters and relationships. Generate contextually rich outputs using embeddings. Write a program to generate 5 semantically similar words for a given input.
3. Train a custom Word2Vec model on a small dataset. Train embeddings on a domain-specific corpus (e.g., legal, medical) and analyze how embeddings capture domain-specific semantics.
4. Use word embeddings to improve prompts for Generative AI model. Retrieve similar words using word embeddings. Use the similar words to enrich a GenAI prompt. Use the AI model to generate responses for the original and enriched prompts. Compare the outputs in terms of detail and relevance.
5. Use word embeddings to create meaningful sentences for creative tasks. Retrieve similar words for a seed word. Create a sentence or story using these words as a starting point. Write a program that: Takes a seed word. Generates similar words. Constructs a short paragraph using these words.
6. Use a pre-trained Hugging Face model to analyze sentiment in text. Assume a real-world application, Load the sentiment analysis pipeline. Analyze the sentiment by giving sentences to input.
7. Summarize long texts using a pre-trained summarization model using Hugging face model. Load the summarization pipeline. Take a passage as input and obtain the summarized text.
8. Install langchain, cohere (for key), langchain-community. Get the api key(By logging into Cohere and obtaining the cohere key). Load a text document from your google drive . Create a prompt template to display the output in a particular manner.
9. Take the Institution name as input. Use Pydantic to define the schema for the desired output and create a custom output parser. Invoke the Chain and Fetch Results. Extract the below Institution related details from Wikipedia: The founder of the Institution. When it was founded.

The current branches in the institution. How many employees are working in it. A brief 4-line summary of the institution. 10. Build a chatbot for the Indian Penal Code. We'll start by downloading the official Indian Penal Code document, and then we'll create a chatbot that can interact with it. Users will be able to ask questions about the Indian Penal Code and have a conversation with it.

Laboratory Outcome	**At the end of the course the student will be able to:** **CO1**:Develop the ability to explore and analyze word embeddings, perform vector arithmetic to investigate word relationships, visualize embeddings using dimensionality reduction techniques **CO2**:Apply prompt engineering skills to real-world scenarios, such as information retrieval, text generation. **CO3**:Utilize pre-trained Hugging Face models for real-world applications, including sentiment analysis and text summarization **CO4**. Apply different architectures used in large language models, such as transformers, and understand their advantages and limitations.

Conduct of Practical Examination:

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks out of 50) and for the SEE minimum passing mark is 35% of the maximum marks (18 out of 50 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation (CIE):

CIE marks for the practical course are 50 Marks.

The split-up of CIE marks for record/ journal and test are in the ratio 60:40.

● Each experiment is to be evaluated for conduction with an observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments are designed by the faculty who is handling the laboratory session and are made known to students at the beginning of the practical session.

● Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.

● Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).

● Weightage to be given for neatness and submission of record/write-up on time.

● Department shall conduct a test of 100 marks after the completion of all the experiments listed in the syllabus.

● In a test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.

● The suitable rubrics can be designed to evaluate each student's performance and learning ability.

● The marks scored shall be scaled down to 20 marks (40% of the maximum marks).

The Sum of scaled-down marks scored in the report write-up/journal and marks of a test is the total CIE marks scored by the student.

The Correlation of Course Outcomes (CO's) and Program Outcomes (PO's)

Subject Code:	BAIL657C	Title: Generative AI						Faculty Name: Mrs. Madhu Nagaraj		
List of Course Outcomes	Program Outcomes									
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO-1	3	2	2	2	1	-	-	2	-	-	2
CO-2	3	3	3	2	3	-	-	3	-	-	2
CO-3	3	3	1	3	2	2	-	1	-	-	2
CO-4	3	3	2	1	3	2	-	3	-	-	2
Total	12	11	8	8	9	4	-	9	-	-	8

Note: 3 = Strong Contribution 2 = Average Contribution 1 = Weak Contribution - = No Contribution

The Correlation of Course Outcomes (CO's) and Program Specific Outcomes (PSO's)

Subject Code:	BAIL657C	TITLE: Generative AI	Faculty Name: Mrs. Madhu Nagaraj
List of Course Outcomes	Program Specific Outcomes		
	PSO1	PSO2	Total
CO-1	3	2	5
CO-2	3	2	5
CO-3	3	1	4
CO-4	3	3	6