USN | 1 | R | V | 2 | 2 | A | I | 0 | 0 | 7 |

RV COLLEGE OF ENGINEERING®

(An Autonomous Institution Affiliated to VTU)

VI Semester B. E. Regular Examinations August-2025

Artificial Intelligence and Machine Learning

NATURAL LANGUAGE PROCESSING AND TRANSFORMERS

Time: 03 Hours ***Maximum Marks: 100***

Instructions to candidates:

1. Answer all questions from Part A. Part A questions should be answered in first three pages of the answer book only.
2. Answer FIVE full questions from Part B. In Part B question number 2 is compulsory. Answer any one full question from 3 and 4, 5 and 6, 7 and 8, 9 and 10.

PART-A

			M	BT	CO
1	1.1	List any two scenarios where deep learning poses a challenge in NLP applications.	02	2	1
	1.2	What is tokenization in NLP? Write a simple Python snippet using NLTK.	02	3	2
	1.3	Write a Python snippet to correct spelling using Text Blob.	02	3	2
	1.4	Write code to find Synsets of the word "computer" and print their definitions.	02	3	2
	1.5	Define word collocations with example.	02	1	1
	1.6	Differentiate between lemmatization and stemming.	02	2	1
	1.7	List any two differences between rule-based and stochastic POS tagging approaches.	02	2	1
	1.8	What is positional embedding, and why is it required in Transformers?	02	2	3
	1.9	What is the purpose of the Attention Mechanism in transformers?	02	2	3
	1.10	Write a line of code to list available models on Hugging Face Hub using Transformers Library.	02	3	4

PART-B

2	a	Explain the typical architecture of an NLP pipeline. How does textual data evolve in structure and form through its successive phases?	08	2	1
	b	Explain the role of Unicode normalization and system-specific error correction in text pre-processing.	08	2	1
3	a	Write a Python script to train a custom sentence tokenizer using NLTK's PunktSentence Tokenizer. Include the training step in your code.	08	3	2
	b	Explain the different ways to tokenize using wordnet with an example.	08	2	1
		OR			
4	a	Discuss various text normalization techniques in NLP, such as removing repeated characters, matching words using regular expressions, and spelling correction using the Enchant library. Provide code snippets for each.	12	4	2
	b	Write a Python script to remove repeating characters from a string (e.g., converting *"AAAIIMML"* to *"AIML"*) using regular expressions.	04	3	2

5	a	Describe how default tagging and unigram tagging work. How can backoff tagging improve tagging accuracy? Explain with suitable examples.	08	2	1
	b	Demonstrate how to combine multiple POS taggers (e.g unigram, bigram) using backoff tagging in NLTK. Write code to show how backoff improves tagging performance.	08	3	2
		OR			
6	a	Implement a classifier-based POS tagger using scikit-learn's Logistics Regression. Extract features such as word suffixes and previous word tags.	08	3	2
	b	What is a Brill tagger? Explain its training process and how it combines rule-based and transformation-based techniques for POS tagging.	08	2	2
7	a	Explain the Encoder-Decoder framework in NLP. Describe how it processes input sequences and generates output, mentioning the role of attention mechanisms.	08	2	3
	b	Explain the components of the Hugging Face ecosystem, such as the Hub, Tokenizers, Datasets, and Accelerate. Illustrate with code how to load a dataset and tokenizer from the Hugging Face Hib and tokenize a sample text.	08	3	4
		OR			
8	a	Explain character and Sub word Tokenization with example. Justify why Numericalization is required in character tokenization.	08	4	2
	b	Illustrate the pre-trained transformer model for specific NLP tasks such as text summarization, with an example.	08	2	2
9	a	Explain the concept of self-attention in transformer architecture and how it contributes to model ability to understand contextual relationships.	08	2	3
	b	Describe the positional embedding's in the transformer model and how they help capture the sequential information in the input.	08	2	
		OR			
10	a	Describe the key components of the transformer decoder branch and how they different from those in the encoder branch in terms of functionality and purpose.	08	2	3
	b	Discuss the challenges associated with generating the coherent text using language models like GPT-2.	08	2	4