



Experiment No 8

Aim: To perform Text summarization using available libraries like Hugging Face's Transformers, TensorFlow or PyTorch

Introduction:

Text Summarization using BERT

Extractive Summarization: BERT can be utilized for extractive summarization, where key sentences or phrases are selected from the original text to form a summary.

Sentence Embedding's: Use BERT to generate embedding's for each sentence in the text.

Similarity Measures: Calculate similarity scores between sentences (e.g., using cosine similarity).

Select Top Sentences: Choose sentences with the highest similarity scores to form the summary.

Abstractive Summarization: This involves generating a summary that might not directly include sentences from the original text. BERT can aid in this by fine-tuning a model specifically for abstractive summarization.

Fine-tuning: Fine-tune the pre-trained BERT model on a summarization dataset (e.g., CNN/Daily Mail dataset).

Sequence-to-sequence Model: Employ techniques like seq2seq models or transformers to generate summaries.

Tools and Libraries:

Hugging Face's Transformers: This library provides easy access to pre-trained models like BERT and various other NLP-related functionalities for tasks like tokenization, model loading, and fine-tuning.

TensorFlow or PyTorch: Use these deep learning frameworks to implement BERT-based models and fine-tuning for specific tasks.

Lab Experiment to be performed in this session:

1. Perform Text Summarization using BERT (BERT summarizer library can be directly installed in python using the following commands `python pip install bert-extractive-summarizer` for the easies of the implementation.)