Week 09.09 – 15.09:

Building a Basic Text Preprocessing Pipeline

Objective:

By the end of this week, you will have created a fully functioning text preprocessing pipeline that transforms raw text into clean, ready-to-analyze data and a N-Gram Language Model.

Key Concepts to Explore:

1. **Tokenization**:
    - Splitting raw text into individual units (words, subwords, or characters).
    - Explore different tokenization methods and their impact on the data.
2. **Stemming vs. Lemmatization**:
    - Understand the differences between the two methods.
    - Compare the use cases of both methods.
3. **Stopword Removal**:
    - Learn about common stopwords (like "the", "and", "is") and why they are often removed.
4. **Normalization**:
    - Techniques like lowercasing and punctuation removal that standardize the text.

Practical Task:

1. **Dataset**:
    - Use the IMDB Dataset as a foundation.
2. **Text Processing Steps**:
    - Implement **Tokenization**, **Stopword Removal**, **Stemming/Lemmatization** and **Normalization** using Python.
3. **Implementation**:
    - Display a comparison of original vs. processed text to see the transformation.
    - Save the processed data in a format (e.g., CSV) ready for analysis.
4. **N-Grams**:
    - Implement bi-grams or tri-grams and identify the top 10 most frequent N-grams in your dataset.

Goal: Use this N-Grams to create a rudimentary N-Gram Language Model.