Edit Distance Problem - Assignment Report

Made by:
Aryaman Srivastava 2210110206

Introduction

This report explains the iterative process undertaken to solve the Edit Distance problem using a language model (LLM). I used ChatGPT 4o for this assignment. The problem is about finding the minimum number of operations required to convert one string into another. The operations allowed are insertions, deletions, and substitutions. The goal of this assignment was to acquire the solution from an LLM and analyze the approach.

Problem Definition

The Edit Distance problem calculates the minimum number of operations required to transform one string (A) into another string (B). The operations include:
- Insertion of a character
- Deletion of a character
- Substitution of a character

The algorithm should handle both empty strings and special characters.

Approach and Process

Initially, I requested the LLM to provide an efficient solution for the Edit Distance problem. The LLM suggested a dynamic programming approach, which builds a matrix to store intermediate results of smaller subproblems. This approach ensures optimal performance, with a time complexity of O(m * n), where m and n are the lengths of the strings.

To ensure robustness, I iteratively refined the problem-solving process to include various test cases, including cases with empty strings and strings with special characters.

Solution

The LLM provided a dynamic programming-based solution in Java. The key steps involved in the solution are:
1. Initializing a DP table with dimensions (m+1) x (n+1), where m and n are the lengths of strings A and B.
2. Filling in the base cases where one of the strings is empty.
3. Iterating over the characters of both strings to calculate the minimum cost for each transformation.
4. Handling all edge cases, including empty strings and special characters.

The final solution successfully computes the edit distance between two strings in an optimal manner.

Conclusion

In conclusion, the iterative process of refining the problem-solving approach using an LLM resulted in an efficient solution to the Edit Distance problem. The dynamic programming approach ensures optimal performance while handling a wide range of test cases. The solution provided by the LLM not only meets the problem's requirements but also offers a clear and structured implementation in Java.