

ALGORITHM PROJECT – KITT DP

Write a brief explanation of why you have chosen the DP algorithm to solve the problem.

I chose the DP algorithm Longest Common Subsequence (LCS) to solve this problem because it efficiently finds the longest common subsequence between two texts for determining the similarity percentage, because there exists overlapping at comparing words.

Identify the time complexity of your solution.

$O(m * n)$

Is there a non-dynamic programming solution? if so, please explain the idea and compare it with your solution.

Yes, instead of using a matrix with LCS implementation, we could also make a solution using only counters, we should iterate over the first text and then over the second text, and according to words matches we could increment the counters to get the total words similarity to convert into a percentage later.

ANALYSIS:

- The text analysis of similarity is highlighted in yellow
- The actual output is highlighted in green

E.g. 1

The diagram illustrates the similarity calculation between two texts. Two boxes at the top show the texts with words highlighted in yellow. The first box contains: "This text should show what a printed text will look like at this place. If you read this text you will get no information." The second box contains: "This paragraph should show what a printd text will look like at this place. If you read this text you will get no informaton." A red arrow points from the first box to the output section. The output section shows the following:

Output:

- 95.83%
- printd - printed, informaton - information

Calculations and corrections are shown to the right:

length = 24
LCS = 21 + 2 (spelling error)

$(23 * 100) / 24 = 95.83 \%$

printd - printed
informaton - information

E.g. 2

This text should show what a printed text will look like at this place. If you read this text you will get no information.

A blind text like this gives you information about the selected font.

$$\begin{aligned} \text{length} &= 12 \\ \text{LCS} &= 6 \\ (6 * 100) / 12 \\ &= 50 \% \end{aligned}$$

Output: 16.67%

E.g. 3

Richard Bellman is best known as the father of dynamic programming. He was the author of many books and the recipient of many honors including the first Norbert Wiener Prize in Applied Mathematics.

Richard Bellman was the author of many books in matematicas.

$$\begin{aligned} \text{length} &= 10 \\ \text{LCS} &= 9 + 1 \text{ (spelling error)} \end{aligned}$$

$$\begin{aligned} (10 * 100) / 10 \\ &= 100 \% \end{aligned}$$

Output:

- 80.00%
- matematicas - Mathematics

matematicas - mathematics