

Edit Distance



張智星 (Roger Jang)

jang@mirlab.org

<http://mirlab.org/jang>

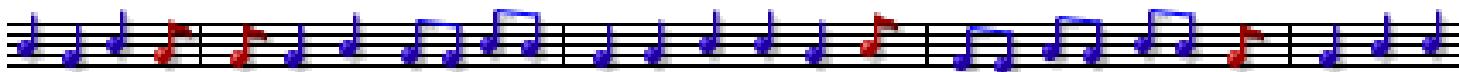
多媒體資訊檢索實驗室

台灣大學 資訊工程系

Edit Distance

⌘ Edit distance

- ☐ The minimum total cost of the basic operations (**delete** with cost 1, **insert** with cost 1, **substitute** with cost 2) that are required to convert a string into another.
45度角
- ☐ Used in “diff” command of unix.



DP for Edit Distance: 3-step Formula

Three - step DP formula for computing $ed(\vec{A}, \vec{B})$

1. Optimum - value function

$ed(\vec{p}, \vec{q})$ is the edit distance between string \vec{p} and \vec{q} .

2. Recurrent formula

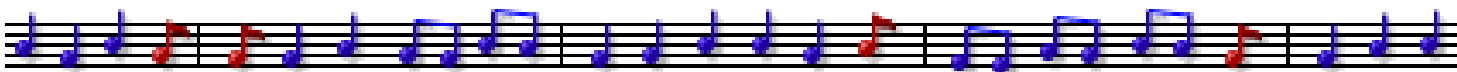
de
a

f
x

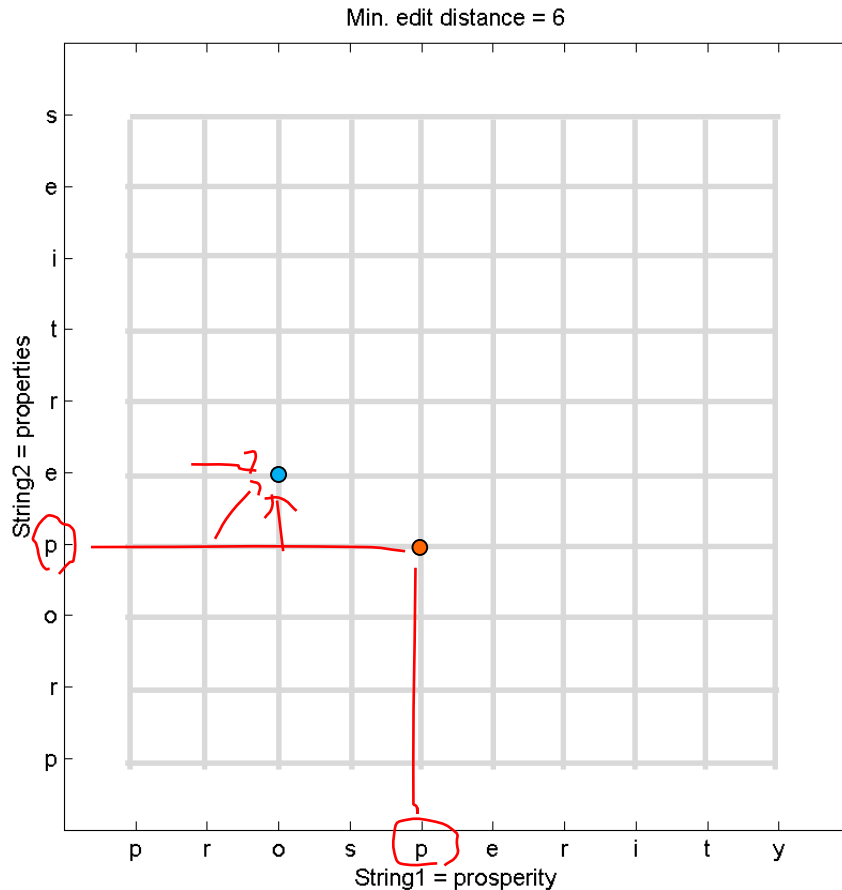
$$ed(\vec{ax}, \vec{by}) = \begin{cases} ed(\vec{a}, \vec{b}), & \text{if } x = y \\ \min \begin{cases} ed(\vec{ax}, \vec{b}) + 1 & \text{insertion} \\ ed(\vec{a}, \vec{by}) + 1, & \text{if } x \neq y \quad \text{deletion} \\ ed(\vec{a}, \vec{b}) + 2 & \text{substitute} \end{cases} \end{cases}$$

Boundary condition : $ed(\vec{a}, []) = len(\vec{a}), ed([], \vec{b}) = len(\vec{b})$

3. Answer : $ed(\vec{A}, \vec{B})$



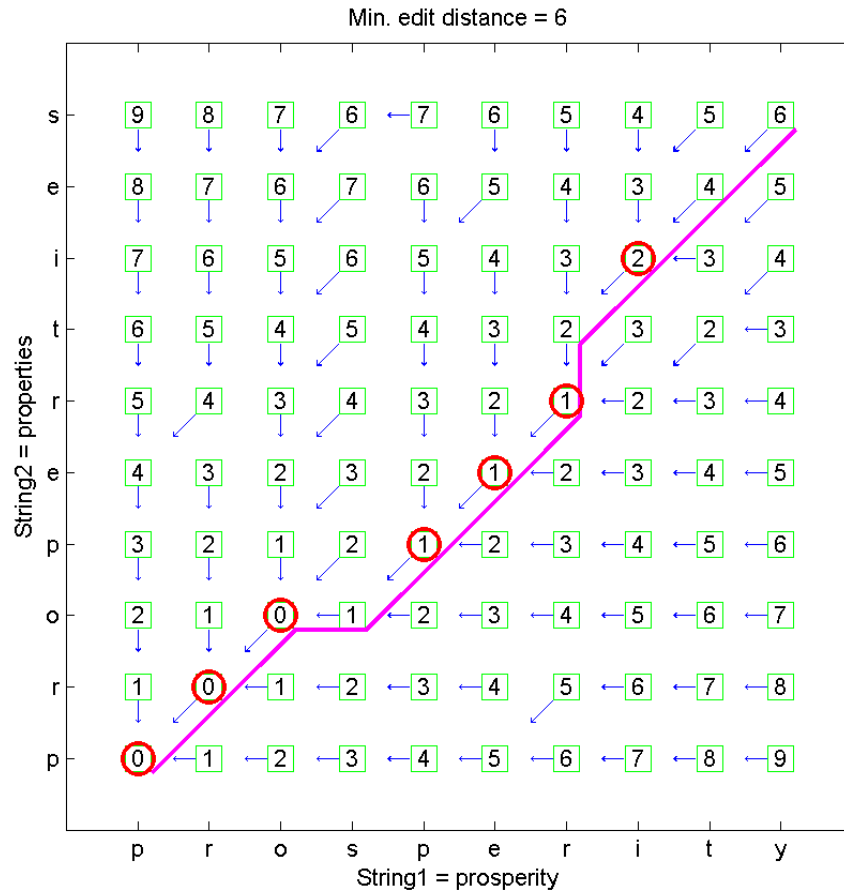
DP for Edit Distance: Table Filling (1/2)



- $ed(prosp, prop) = ed(pros, prop)$
- $ed(pro, prope) = \min \begin{cases} ed(pro, prop) + 1 \\ ed(pr, prope) + 1 \\ ed(pr, prop) + 2 \end{cases}$



DP for Edit Distance: Table Filling (2/2)



⌘ To create this plot

⌘ Download Machine Learning Toolbox

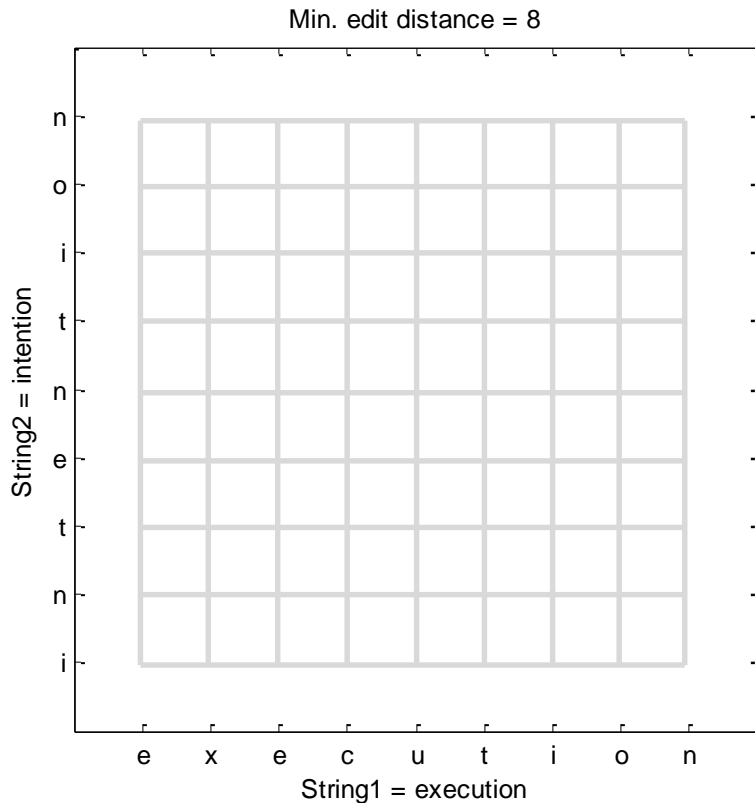
⌘ Run

`editDistance('prosperity',
'properties', [2 1 1], 1)`
under MATLAB

⌘ A 45-degree back-tracking path indicates a “substitute” or “equal”.



DP for Edit Distance: Quiz for Table Filling



⌘ Hints

- ⏏ Create a $(m+1) \times (n+1)$ matrix for table filling
- ⏏ Fill row 0 and column 0 first to establish the base cases of boundary conditions
- ⏏ Fill all the other elements in a layer-by-layer manner.

