CSE – 443
Pattern Recognition

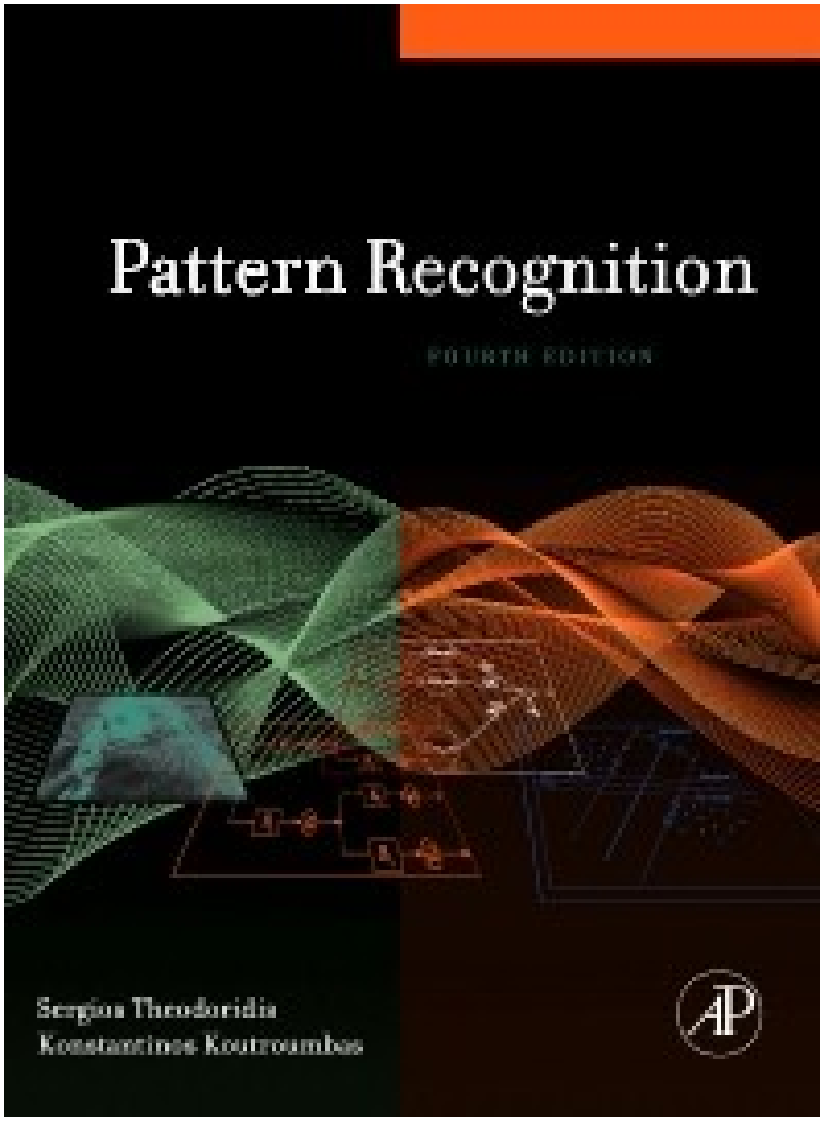
Chapter 08
Template Matching

Text Book - 01

- **Pattern Recognition**

  by

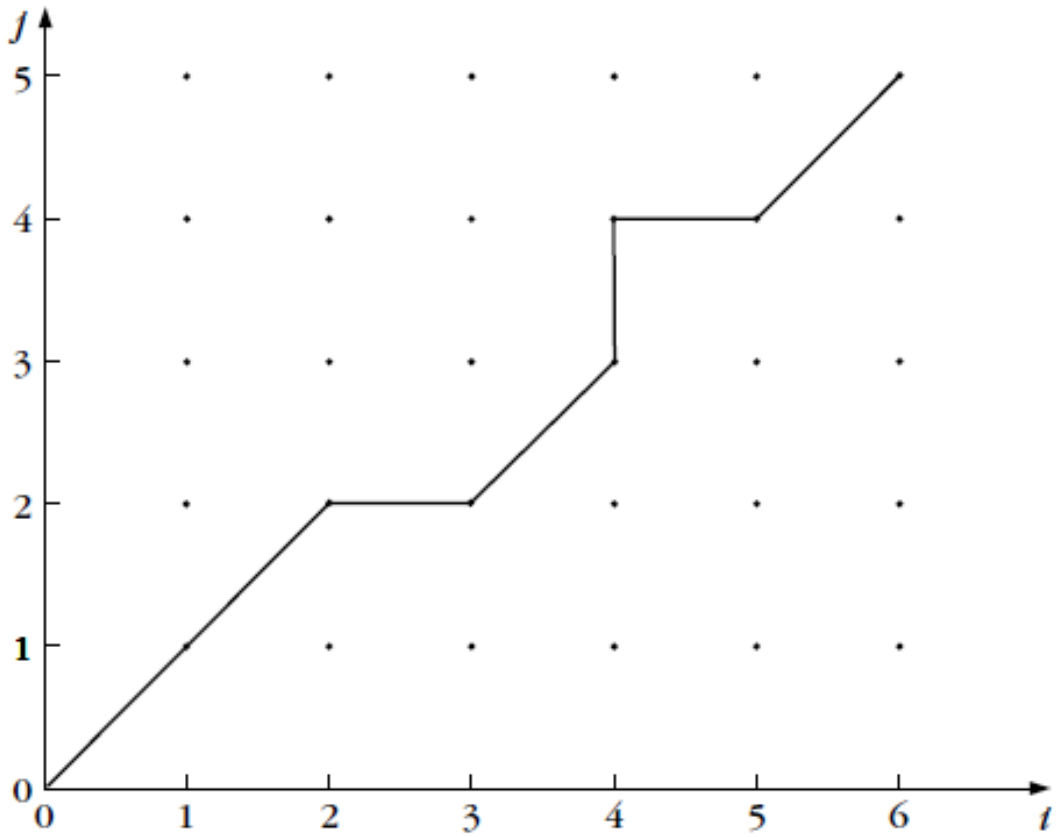
  Sergios Theodoridis



Definition

- A measure or a cost measuring the“distance”or the“similarity”between the **(known) reference patterns** and the **(unknown) test pattern**, in order to perform the matching operation known as **Template Matching**.

Measures Based on Optimal Path Searching Techniques



Bellman's Optimality Principle and Dynamic Programming

Let the optimal path between an initial node (i_0, j_0) and a final one (i_f, j_f) be denoted as

$$(i_0, j_0) \xrightarrow{opt} (i_f, j_f)$$

If (i, j) is an intermediate node between (i_0, j_0) and (i_f, j_f), we will denote the optimal path constrained to pass through (i, j) as

$$(i_0, j_0) \xrightarrow[(i,j)]{opt} (i_f, j_f)$$

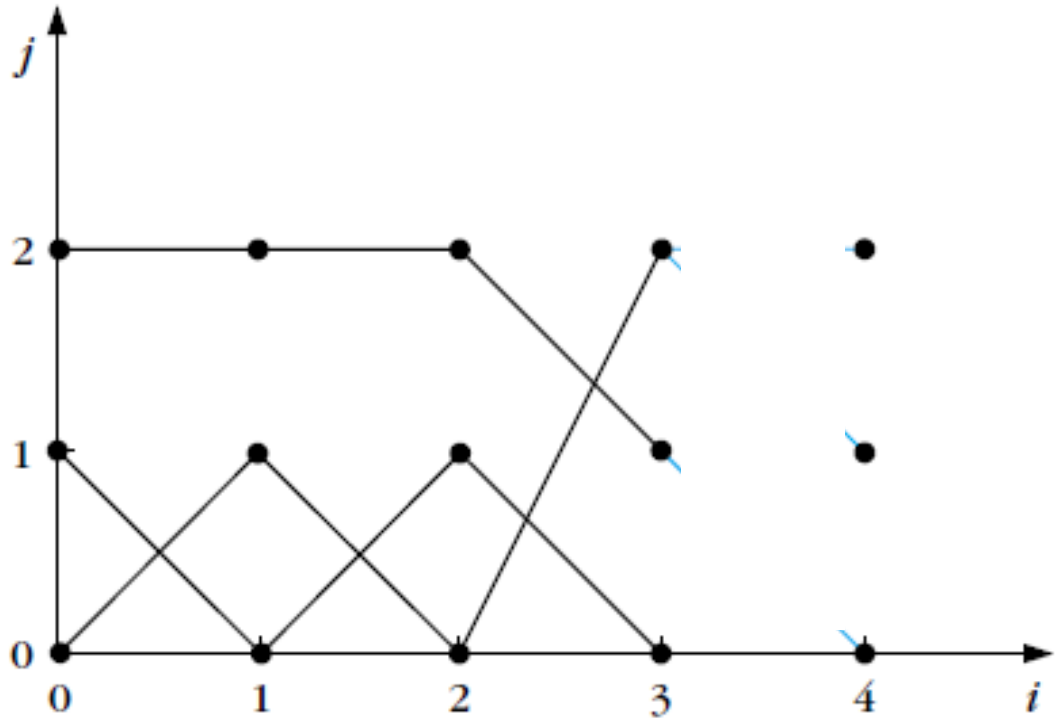
Bellman's principle states that [Bell 57]

$$(i_0, j_0) \xrightarrow[(i,j)]{opt} (i_f, j_f) = (i_0, j_0) \xrightarrow{opt} (i, j) \oplus (i, j) \xrightarrow{opt} (i_f, j_f)$$

Bellman's Optimality Principle and Dynamic Programming

$$D_{\min}(i_k, j_k) = \min_{i_{k-1}, j_{k-1}} [D_{\min}(i_{k-1}, j_{k-1}) + d(i_k, j_k | i_{k-1}, j_{k-1})]$$

Example 8.1

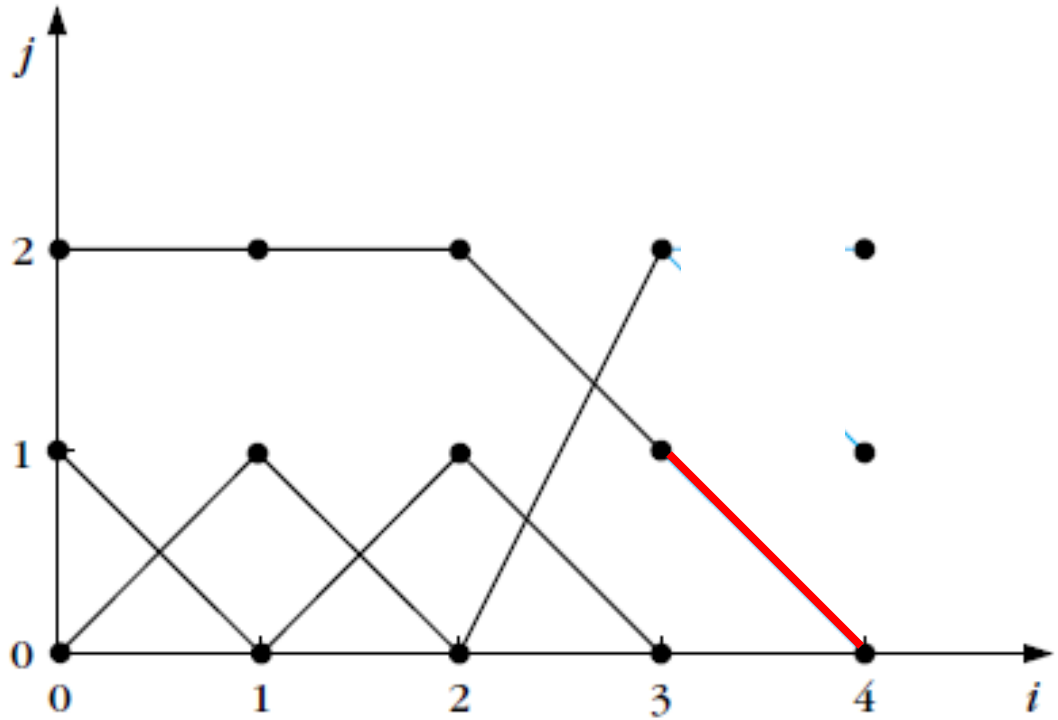


$J_k = 0, 1, 2$ for each k

k = 3, J3 = 0, 1, 2

k = 4, J4 = 0, 1, 2

Example 8.1



$J_k = 0, 1, 2$ for each k

k = 3, J3 = 0, 1, 2

k = 4, J4 = 0, 1, 2

Example 8.1

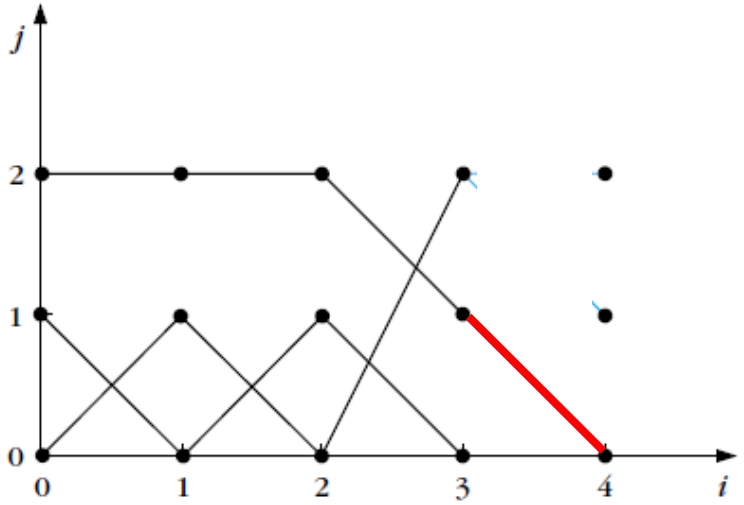
- Assume that the accumulated costs of the optimal paths

  Dmin(3, j3), j3 0, 1, 2 at the respective nodes are:

  Dmin(3, 0) = 0.8

  Dmin(3, 1) = 1.2

  Dmin(3, 2) = 1.0



Example 8.1

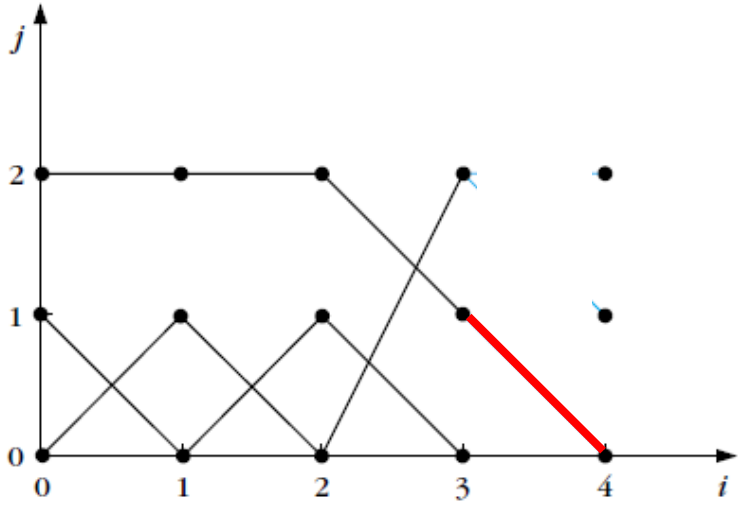
Table 8.1 Transition Costs Between Nodes for the Example 8.1

Nodes	(4, 0)	(4, 1)	(4, 2)
(3, 0)	0.8	0.6	0.8
(3, 1)	0.2	0.3	0.2
(3, 2)	0.7	0.2	0.3

$$Dmin(3, 0) = 0.8$$

$$Dmin(3, 1) = 1.2$$

$$Dmin(3, 2) = 1.0$$



Edit Distance

- Types of error:
  - ✓ Wrongly identified symbol (e.g.,“befuty” instead of “beauty”)
  - ✓ Insertion error (e.g.,“bearuty”)
  - ✓ Deletion error (e.g.,“beuty”)

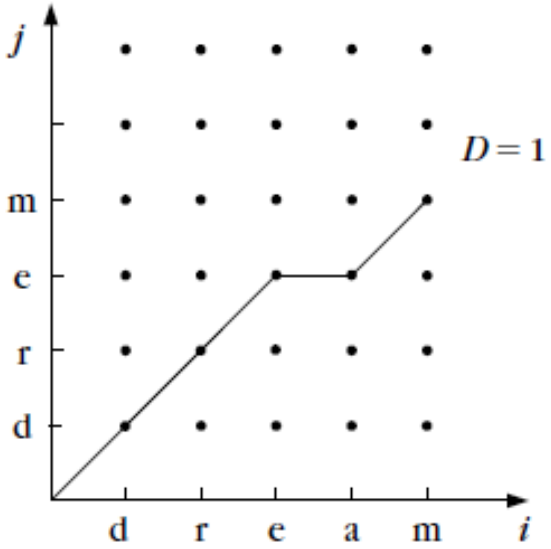
Obviously, a combination of these errors may also occur.

Edit Distance

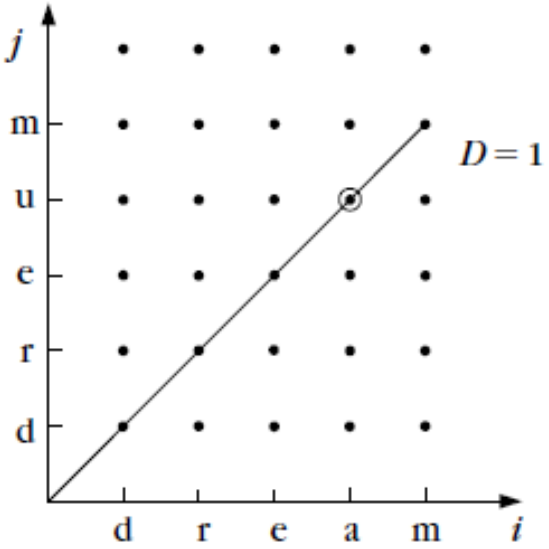
- The Edit distance between two string patterns A and B, denoted D(A, B), is defined as the minimum total number of changes **C**, insertions **I** ,and deletions **R** required to change pattern A into pattern B,

$$D(A,B) = \min_{j}[C(j) + I(j) + R(j)]$$

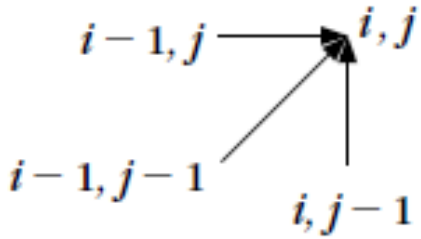
where j runs over all possible combinations of symbol variations in order to obtain B from A.

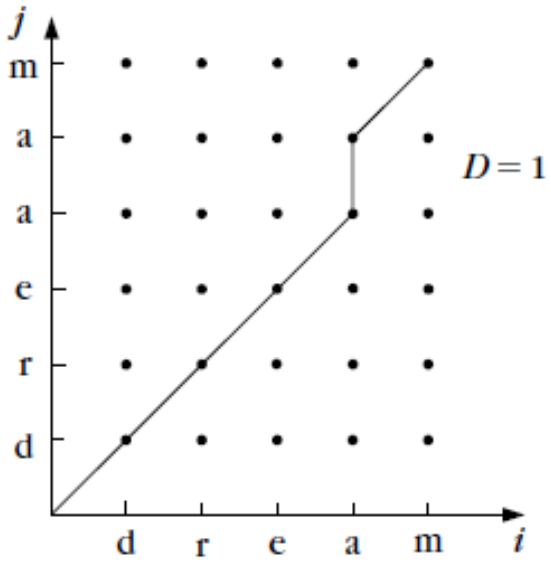

j
m
e
r
d
D = 1
d r e a m i

(a)

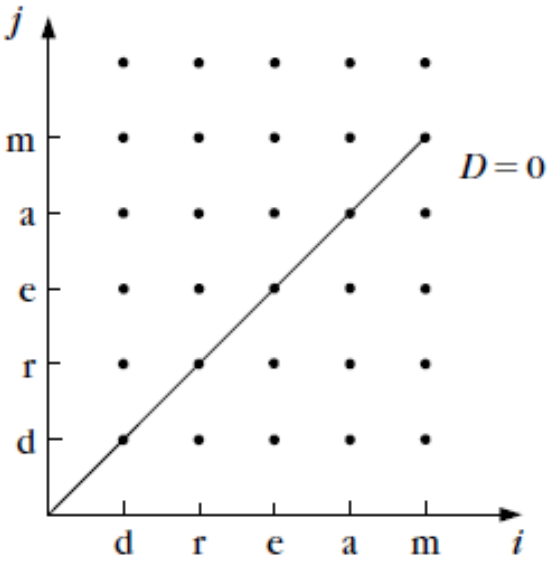

j
m
u
e
r
d
D = 1
d r e a m i

(b)

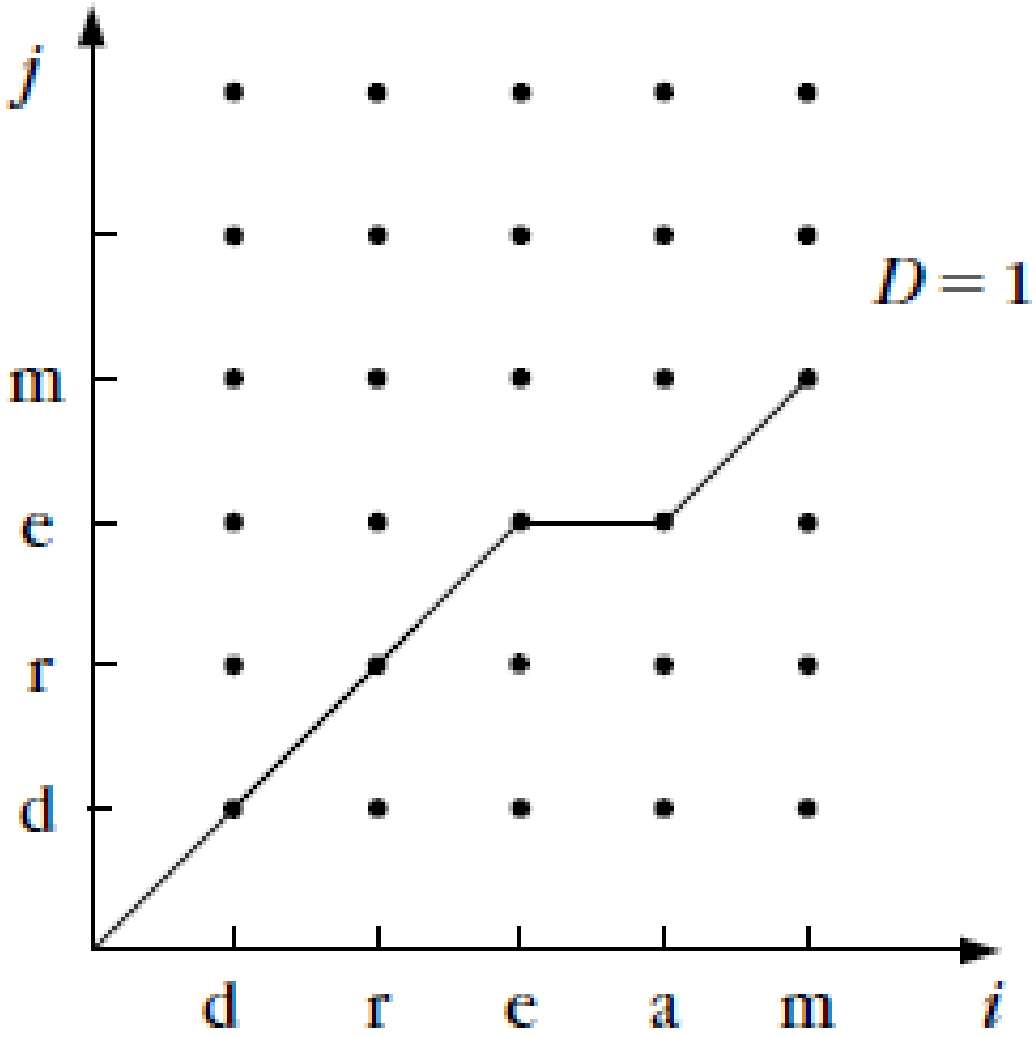

i − 1, j
i, j
i − 1, j − 1
i, j − 1


j
m
a
a
e
r
d
D = 1
d r e a m i

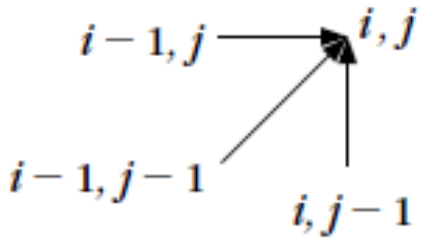
(c)


j
m
a
e
r
d
D = 0
d r e a m i

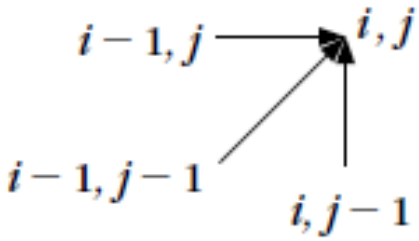
(d)



(a)



Edit Distance



The costs associated with the above three transitions are:

1. Diagonal transitions:

$$d(i, j|i-1, j-1) = \begin{cases} 0 & \text{if } r(i) = t(j) \\ 1 & \text{if } r(i) \neq t(j) \end{cases}$$

2. Horizontal and vertical transitions:

$$d(i, j|i-1, j) = d(i, j|i, j-1) = 1$$

Edit Distance Algorithm

I = Test String Len
J = Ref String Len

- $D(0,0) = 0$
- For $i = 1$ to I
  - $D(i,0) = D(i-1,0) + 1$
- End { For }
- For $j = 1$ to J
  - $D(0,j) = D(0,j-1) + 1$
- End { For }

- For $i = 1$ to I
  - For $j = 1$ to J
    - $c1 = D(i-1,j-1) + d(i,j|i-1,j-1)$
    - $c2 = D(i-1,j) + 1$
    - $c3 = D(i,j-1) + 1$
    - $D(i,j) = \min(c1,c2,c3)$
  - End { For }
- End { For }
- $D(A,B) = D(I,J)$

Edit Distance Algorithm-1

I = Test String Len = 04
J = Ref String Len = 05

I

M	i=4						
E	i=3						
R	i=2						
D	i=1						
	i=0	0					
		j=0	j=1	j=2	j=3	j=4	j=5
			D	R	E	A	M

J

- $D(0,0) = 0$
- For $i = 1$ to I
  - $D(i,0) = D(i-1,0) + 1$
- End { For }
- For $j = 1$ to J
  - $D(0,j) = D(0,j-1) + 1$
- End { For }

Edit Distance Algorithm-2

I = Test String Len = 04
J = Ref String Len = 05

M	i=4	4					
E	i=3	3					
R	i=2	2					
D	i=1	1					
	i=0	0					
		j=0	j=1	j=2	j=3	j=4	j=5
			D	R	E	A	M

- $D(0,0) = 0$
- For $i = 1$ to I
  - $D(i,0) = D(i-1,0) + 1$
- End { For }
- For $j = 1$ to J
  - $D(0,j) = D(0,j-1) + 1$
- End { For }

Edit Distance Algorithm-3

I = Test String Len = 04
J = Ref String Len = 05

M	i=4	4					
E	i=3	3					
R	i=2	2					
D	i=1	1					
	i=0	0	1	2	3	4	5
		j=0	j=1	j=2	j=3	j=4	j=5
			D	R	E	A	M

- $D(0,0) = 0$
- For $i = 1$ to I
  - $D(i,0) = D(i-1,0) + 1$
- End { For }
- For $j = 1$ to J
  - $D(0,j) = D(0,j-1) + 1$
- End { For }

Edit Distance Algorithm

I = Test String Len
J = Ref String Len

- $D(0,0) = 0$
- For $i = 1$ to I
  - $D(i,0) = D(i-1,0) + 1$
- End { For }
- For $j = 1$ to J
  - $D(0,j) = D(0,j-1) + 1$
- End { For }

- For $i = 1$ to I
  - For $j = 1$ to J
    - $c1 = D(i-1,j-1) + d(i,j|i-1,j-1)$
    - $c2 = D(i-1,j) + 1$
    - $c3 = D(i,j-1) + 1$
    - $D(i,j) = \min(c1,c2,c3)$
  - End { For }
- End { For }
- $D(A,B) = D(I,J)$

Edit Distance Algorithm-4

I = Test String Len = 04
J = Ref String Len = 05

M	i=4	4					
E	i=3	3					
R	i=2	2					
D	i=1	1	0				
	i=0	0	1	2	3	4	5
		j=0	j=1	j=2	j=3	j=4	j=5
			D	R	E	A	M

- For $i = 1$ to I
  - For $j = 1$ to J
    - $c1 = D(i-1, j-1) + d(i,j|i-1,j-1)$
    - $c2 = D(i-1, j) + 1$
    - $c3 = D(i, j-1) + 1$
    - $D(i,j) = \min(c1, c2, c3)$
  - End { For }
- End { For }
- $D(A,B) = D(I,J)$

Edit Distance Algorithm-4

I = Test String Len = 04
J = Ref String Len = 05

M	i=4	4					
E	i=3	3					
R	i=2	2					
D	i=1	1	0	1			
	i=0	0	1	2	3	4	5
		j=0	j=1	j=2	j=3	j=4	j=5
			D	R	E	A	M

- For $i = 1$ to I
  - For $j = 1$ to J
    - $c1 = D(i-1, j-1) + d(i,j|i-1,j-1)$
    - $c2 = D(i-1, j) + 1$
    - $c3 = D(i, j-1) + 1$
    - $D(i,j) = \min(c1, c2, c3)$
  - End { For }
- End { For }
- $D(A,B) = D(I,J)$

Course Site

- CSE 443 – Pattern Recognition:
  https://sites.google.com/site/mistcsecourses/cse443pr2017

- CSE 444 - Pattern Recognition Sessional:
  https://sites.google.com/site/mistcsecourses/cse444prs2017


Thank You