Iran University of Science and Technology, Computer Department

NLP Homework 2, Part 1

Vahid Kharazi

November 27, 2015

1 Minimum Edit Distance

According to wikipedia and course slides, we have to implement minimum edit distance algorithm:

$$\operatorname{lev}_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0, \\ \min \begin{cases} \operatorname{lev}_{a,b}(i-1,j) + 1 \\ \operatorname{lev}_{a,b}(i,j-1) + 1 \\ \operatorname{lev}_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} & \text{otherwise.} \end{cases}$$

To set a custom cost to this algorithm, we should change initialize matrix with insertion and deletion cost and change +1 with *cost*:

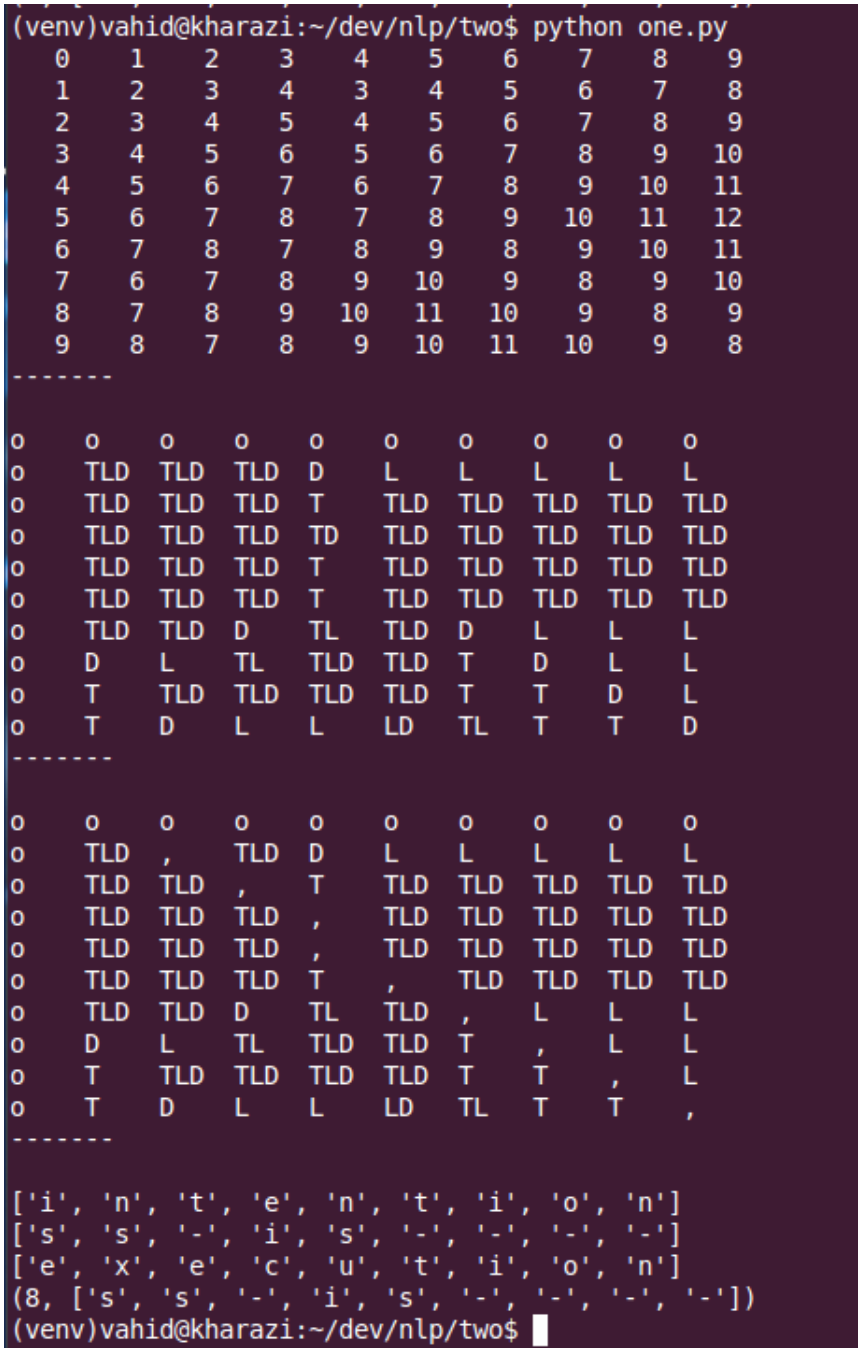
Initialization:

```
D(0,0) = 0
D(i,0) = D(i-1,0) + del[x(i)];     1 < i ≤ N
D(0,j) = D(0,j-1) + ins[y(j)];     1 < j ≤ M
```

Recurrence Relation:

```
              ⎧D(i-1,j)   + del[x(i)]
D(i,j)= min   ⎨D(i,j-1)   + ins[y(j)]
              ⎩D(i-1,j-1) + sub[x(i),y(j)]
```

Test: There is an example in course slides to compute edit distance between intention and execution. here is my result of this example:



The first matrix is D, Second matrix is back-track(T = Top, L = Left, D = Diag). Finally you can find the path of minimum edit distance: