#Edit Distance

```
import nltk
```

```
import nltk
s1 = "abc"
s2 = "ebcd"
nltk.edit_distance(s1, s2)
```

```
import nltk
s1 = "intuition"
s2 = "institution"
print("s1->s2",nltk.edit_distance(s1, s2))
print("s2->s1",nltk.edit_distance(s2,s1))
```

```
X='intention'
Y='execution'
print("X->Y",nltk.edit_distance(X,Y))
print("Y->X",nltk.edit_distance(Y,X))
```

```
!pip install strsimpy
```

```
#no. of inserti0ns*cost of insertion+no. of deletion*cost of deletion+
no. of substitution* cost of substitution


1 in 1 del , 3 sub:
c_in: 1, c_del:1, c_subs1: 1
edit distance =1*1+1*1+3*1=1+1+3=5

Case 2:
c_in: 1, c_del:1, c_subs1: 2
1 in 1 del , 3 sub:
weighted edit distance=1*1+1*1+3*2=8
```

```
!pip install strsimpy
```

```
from strsimpy.levenshtein import Levenshtein
```

```
levenshtein = Levenshtein()
levenshtein.distance('1234', '123')   # 1 (deletion/insertion)
levenshtein.distance('1234', '12345') # 1 (deletion/insertion)
levenshtein.distance('1234', '1235')  # 1 (substitution)
levenshtein.distance('1234', '1324')  # 2 (substitutions)
levenshtein.distance('1234', 'ABCD')  # 4 (substitutions)
```

```
nltk.edit_distance('1234','ABCD')
```

```
#Refer link:
https://www.nltk.org/api/nltk.metrics.distance.html#nltk.metrics.dista
nce.edit_distance
```

```
nltk.edit_distance_align('intention', 'execution',
substitution_cost=2)

nltk.edit_distance_align('intention', 'execution',
substitution_cost=1)
```

#N-grams

```
from nltk import ngrams
sentence = 'I reside in Bengaluru.'
n = 1
unigrams = ngrams(sentence.split(), n)
for grams in unigrams:
  print(grams)

from nltk import ngrams
sentence = 'I reside in Bengaluru.'
n = 2
bigrams = ngrams(sentence.split(), n)
for grams in bigrams:
  print(grams)

from nltk import ngrams
sentence = 'I reside in Bengaluru.'
n = 3
trigrams = ngrams(sentence.split(), n)
for grams in trigrams:
  print(grams)

#https://www.kaggle.com/code/alvations/n-gram-language-model-with-nltk
```