School of Computing and Information Systems
The University of Melbourne
COMP90049
Knowledge Technologies (Semester 1, 2019)
Workshop sample solutions: Week 4

Suppose that we have observed the token `lended` and we have a dictionary as follows:

```
addendum
blenders —3
commodity
deaden d — ~~2~~ 3
end ed — 3
leader d — 2
leant ed — 4
lent ded — 3
lemonade
pleading
```

1. Which, if any, of the above dictionary entries would be returned using a Neighbourhood Search with a neighbourhood of 1? 2? 3?

   - There aren't any items in the dictionary requiring only a single change from `lended`.
   - With a neighbourhood size of 2, there is a dictionary entry:
     - `leader`, by Replacing the `n` with `a`, and the second `d` with `r`
   - Along with the above, the following are also within a neighbourhood of 3:
     - `blenders`, by Inserting the `b`, Replacing the second `d` with `r`, and Inserting the `s`
     - `deaden` (three Replaces)
     - `end` (three Deletions)
     - `lent` (one Replace and two Deletions)

2. With respect to the input string `lended` and the dictionary entry `deaden`, calculate the following:

   (a) the Global Edit Distance, using the parameter $[m, i, d, r] = [+1, -1, -1, -1]$

input

(a)	ε		l		e		n		d		e		d
ε	0	←	-1	←	-2	←	-3	←	-4	←	-5	←	-6
	↑	↖		↖		↖		↖				↖	
d	-1		-1	←	-2	←	-3		-2	←	-3	←	-4
	↑	↖	↑	↖						↖			
e	-2		-2		0	←	-1	←	-2		-1	←	-2
	↑	↖	↑		↑	↖		↖			↑	↖	
a	-3		-3		-1		-1	←	-2		-2		-2
	↑	↖	↑		↑	↖	↑	↖				↖	
d	-4		-4		-2		-2		0	←	-1		-1
	↑	↖	↑	↖	↑	↖	↑		↑	↖			
e	-5		-5		-3		-3		-1		1	←	0
	↑	↖	↑		↑	↖			↑		↑	↖	
n	-6		-6		-4		-2		-2		0		0

target

   - From the table above, we can observe that the Global Edit Distance is 0, corresponding to the following sequence of operations: Replace, Match, Replace, Match, Match, Replace, which I will abbreviate as `rmrmmr`. (You can follow along with the highlighted back-pointers.)

global distance · lended: [m, i, d, r] = [+1, -1, -1, -1]

Lib: addendum, blenders, commodity, deaden, end, leader, leant, lemonade, pleading

	ε	l	e	n	d	e	d
ε	0	-1	-2	-3	-4	-5	-6
a	-1	-1	-2	-3	-4	-5	-6
d	-2	-2	-3	-4	-2	-3	-4
d	-3	-3	-3	-4	-3	-3	-2
e	-4	-4	-2	-3	-4	-2	-3
n	-5	-5	-3	-1	-2	-3	-3
d	-6	-6	-4	-2	0	-1	-2
u	-7	-7	-5	-3	-1	-1	-2
m	-8	-8	-6	-4	-2	-2	-2

	ε	l	e	n	d	e	d
ε	0	-1	-2	-3	-4	-5	-6
b	-1	-1	-2	-3	-4	-5	-6
l	-2	0	-1	-2	-3	-4	-5
e	-3	-1	1	0	-1	-2	-3
n	-4	-2	0	2	1	0	-1
d	-5	-3	-1	1	3	2	1
e	-6	-4	-2	0	2	4	3
r	-7	-5	-3	-1	1	3	3
s	-8	-6	-4	-2	0	2	2

	ε	l	e	n	d	e	d
ε	0	-1	-2	-3	-4	-5	-6
l	-1	1	0	-1	-2	-3	-4
e	-2	0	2	1	0	-1	-2
a	-3	-1	1	1	0	-1	-2
d	-4	-2	0	0	2	1	0
e	-5	-3	-1	-1	1	3	2
r	-6	-4	-2	-2	0	2	2

commodity

C05503030

↓

C0503030

↓

C533

leant

L0053

↓

L53.

lent

L053

↓

L53

local edit distance

	ε	l	e	n	d	e	d
ε	0	0	0	0	0	0	0
b	0	0	0	0	0	0	0
l	0	1	0	0	0	0	0
e	0	0	2	1	0	1	0
n	0	0	1	3	2	1	0
d	0	0	0	2	4	3	2
e	0	0	1	1	3	5	4
r	0	0	0	0	2	4	4
s	0	0	0	0	1	3	3

lende - leade

	ε	l	e	n	d	e	d
ε	0	0	0	0	0	0	0
l	0	1	0	0	0	0	0
e	0	0	2	1	0	1	0
a	0	0	1	1	0	0	0
d	0	0	0	0	2	1	1
e	0	0	1	0	0	3	2
r	0	0	0	0	0	2	2

N-Gram lended: le, en, nd, de, ed.

1) end: en, nd. D = 5+2-2×2 = 3

2) blenders: bl, le, en, nd, de, er, rs.

D = 5+7-2×4 = 4.

3) leader: le ea ad de er

D = 5+5-2×2 = 6.

Soundex

a. e. h. i. o. u. w. y → 0

b. f. p. v → 1

c. g. j. k. q. s. x. z → 2

d. t → 3

l → 4 m. n → 5 r → 6.

lended.

L05303

↓

L533.

blenders.

B4053062

↓

B45362

↓

B453.

leader

L00306

↓

L306

↓

L36.

(b)	ε	l	e	n	d	e	d
ε	0	0	0	0	0	0	0
d	0	0	0	0	↖ 1	← 0	↖ 1
e	0	0	↖ 1	← 0	0	↖ 2	← 1
a	0	0	↑ 0	↖ 0	0	↑ 1	↖ 1
d	0	0	0	0	↖ 1	← 0	↖ 2
e	0	0	↖ 1	← 0	↑ 0	↖ 2	↑ ← 1
n	0	0	↑ 0	↖ 2	← 1	↑ 1	↖ 1

lended → deaden.

→ 找出两个序列之间连续相同的子序列.

(b) the Local Edit Distance, using the parameter $[m, i, d, r] = [+1, -1, -1, -1]$

- From the table above, we can observe that the Local Edit Distance is 2 (highlighted); there are five equivalent-scoring substring matches that it corresponds to:
  - Align `-de-` in `lended` with the first `de-` in `deaden`: `mm`
  - Align `-ded` with `dead-`: `mmim`
  - Align `-de-` in `lended` with the second `-de-` in `deaden`: `mm`
  - Align `-ende-` with `-eade-`: `mrmm`
  - Align `-en-` with `-en`: `mm`

(c) the N-Gram Distance, using $n = 2$

- We begin by generating the 2-grams of the two strings; I will opt not to use the terminal marker (#) here:
  - `lended`: `le`, `en`, `nd`, `de`, `ed`
  - `deaden`: `de`, `ea`, `ad`, `de`, `en`
- Recall that the N-Gram Distance is defined as follows:

$$D(s,t) = \mid G_n(s) \mid + \mid G_n(t) \mid - 2\times \mid G_n(s) \cap G_n(t) \mid$$

- Here we have 5 2-grams in `lended`, as well as 5 in `deaden`. Also, the two sets share 2 2-grams: `de` and `en`. (Note that we don't double-count the `des` in `deaden`, because there is only a single `de` in `lended`)
- Consequently, the 2-gram Distance is $5 + 5 - 2 \times 2 = 6$

3. Find the best approximate match (or matches, if there are ties) in the dictionary for the string `lended`, based on the following methods; consider different parameters where necessary:

(a) the Global Edit Distance

- Using the above scoring parameter, the most similar dictionary entries are `blenders` (+2) and `leader` (+2)
- You might like to try some other parameter setting(s), to see if they give different results.

(b) the Local Edit Distance

- Using the above scoring parameter, the best dictionary entry is `blenders` (+5)
- In this case, changing the parameter is unlikely to result in a different answer. (Why?)

(c) the N-Gram Distance

- If we are using n is 2 and not padding with #, the best dictionary entry is `end`, with a 2-Gram Distance of 3.
- You might find that adding the padding characters or changing n will give different results.

(d) Soundex

- The Soundex code of `lended` is `l533`.
- None of the dictionary entries have this exact code; however, if we permit one mismatch in the Soundex code (as in Neighbourhood Search with a neighbourhood of 1), then the best matches are `commodity` (`c533`), `leant` (`l53`), `lent` (`l53`), and `lemonade` (`l553`)