Odległość edycyjna

Weronika Ormaniec

09.05.2020 r.

1 Odległość edycyjna

[2]:
```
from spacy.tokenizer import Tokenizer
from spacy.lang.pl import Polish
from bisect import bisect
import random
```

1.1 Odległość Levenshteina

[3]:
```
def LevenshteinDistance(word1, word2):
    dist_matrix = [[(0, 0)]*(len(word2)+1) for i in range(len(word1)+1)]

    for i in range(len(word1)+1):
        dist_matrix[i][0] = (i, 0)

    for i in range(len(word2)+1):
        dist_matrix[0][i] = (i, 0)

    for j in range(1, 1+len(word2)):
        for i in range(1, 1+len(word1)):
            if word1[i-1] == word2[j-1]:
                substitution_cost = 0
            else:
                substitution_cost = 1

            min_ = min(dist_matrix[i-1][j][0]+1, dist_matrix[i][j-1][0]+1,
                       dist_matrix[i-1][j-1][0]+substitution_cost)

            if min_==dist_matrix[i-1][j][0]+1:
                dist_matrix[i][j] = (min_, (len(word2)+1)*(i-1)+j)
            elif min_==dist_matrix[i][j-1][0]+1:
                dist_matrix[i][j] = (min_, (len(word2)+1)*i+j-1)
            else:
                dist_matrix[i][j] = (min_, (len(word2)+1)*(i-1)+j-1)

    return dist_matrix[-1][-1], dist_matrix
```

```
def visualization(word1, word2):
    dist, dist_matrix = LevenshteinDistance(word1, word2)
    height = len(dist_matrix)
    width = len(dist_matrix[0])
    i = height-1
    j = width-1
    output = ""
    while i>0 or j>0:
        i_, j_ = dist_matrix[i][j][1]//width, dist_matrix[i][j][1]%width
        if i_ == i-1 and j_ == j-1 and dist_matrix[i_][j_][0] ==␣
 ↪dist_matrix[i][j][0]-1:
            output=f'{word2[:j_]}*{word2[j_]}*{word1[i_+1:]} (substituted␣
 ↪{word1[i_]}->{word2[j_]})\n'+output
        elif i_ == i-1 and dist_matrix[i_][j_][0] == dist_matrix[i][j][0]-1:
            output=f'{word2[:j_]}**{word1[i_+1:]} (subtracted␣
 ↪{word1[i_]})\n'+output
        elif j_ == j-1 and dist_matrix[i_][j_][0] == dist_matrix[i][j][0]-1:
            output=f'{word2[:j_]}*{word2[j_]}*{word1[i_:]} (added␣
 ↪{word2[j_]})\n'+output
        i, j = i_, j_
    print(f'Distance between {word1} and {word2} equals: {dist[0]}\nSteps:')
    print(output)
```

[4]:
```
data = [("los", "kloc"),("Łódź", "Lodz"),("kwintesencja", "quintessence"),
        ("ATGAATCTTACCGCCTCG", "ATGAGGCTCTGGCCCCTG"), ("wojtk", "wjeek")]

for (word1, word2) in data:
    visualization(word1, word2)
    print("-----------")
```

```
Distance between los and kloc equals: 2
Steps:
*k*los (added k)
klo*c* (substituted s->c)

-----------
Distance between Łódź and Lodz equals: 3
Steps:
*L*ódź (substituted Ł->L)
L*o*dź (substituted ó->o)
Lod*z* (substituted ź->z)

-----------
```

```
Distance between kwintesencja and quintessence equals: 5
Steps:
*q*wintesencja (substituted k->q)
q*u*intesencja (substituted w->u)
quintes*s*encja (added s)
quintessenc*e*a (substituted j->e)
quintessence** (subtracted a)

-----------
Distance between ATGAATCTTACCGCCTCG and ATGAGGCTCTGGCCCCTG equals: 7
Steps:
ATGA*G*TCTTACCGCCTCG (substituted A->G)
ATGAG*G*CTTACCGCCTCG (substituted T->G)
ATGAGGCT*C*TACCGCCTCG (added C)
ATGAGGCTCT*G*CCGCCTCG (substituted A->G)
ATGAGGCTCTG*G*CCGCCTCG (added G)
ATGAGGCTCTGGCC**CCTCG (subtracted G)
ATGAGGCTCTGGCCCCT**G (subtracted C)

-----------
Distance between wojtk and wjeek equals: 3
Steps:
w**jtk (subtracted o)
wj*e*k (substituted t->e)
wje*e*k (added e)

-----------
```

1.2 Najdłuższy wspólny podciąg

1.2.1 LCS wśród tokenów

Algorytmy znajdowania długości LCS

[1]:
```
# Hunt-Szymański algorithm
def lcs(list1, list2, visualize=False):
    lcs = []
    ranges = [len(list2)]
    for w in range(len(list1)):

        positions = [j for (j, l) in enumerate(list2) if l == list1[w]]
        positions.reverse()

        if visualize:
            print(ranges)
            i_ = 0
            for i in ranges:
                print(list2[i_:i])
                i_ = i

        for p in positions:
            k = bisect(ranges, p)
            if k == bisect(ranges, p-1):
                if k<len(ranges)-1:
                    ranges[k] = p
                else:
                    ranges[k:k] = [p]
        if visualize:
            print(f'->{list1[w]}')
            print("------------")

    if visualize:
        print(ranges)
        i_ = 0
        for i in ranges:
            print(list2[i_:i])
            i_ = i

    return len(ranges) - 1, ranges
```

[5]:
```python
# Standard algorithm
def lcs2(word1, word2):
    dist_matrix = [[0]*(len(word2)+1) for i in range(len(word1)+1)]

    for i in range(len(word1)+1):
        dist_matrix[i][0] =  0

    for i in range(len(word2)+1):
        dist_matrix[0][i] = 0

    for j in range(1, 1+len(word2)):
        for i in range(1, 1+len(word1)):
            if word1[i-1] == word2[j-1]:
                dist_matrix[i][j] = dist_matrix[i-1][j-1] + 1
            else:
                dist_matrix[i][j] = max(dist_matrix[i][j-1], dist_matrix[i-1][j])

    return dist_matrix[-1][-1], dist_matrix
```

[6]:
```python
lcs("zcbbda", "abcabbabaz", True)
```

```
[10]
abcabbabaz
->z
------------
[9, 10]
abcabbaba
z
->c
------------
[2, 10]
ab
cabbabaz
->b
------------
[1, 4, 10]
a
bca
bbabaz
->b
------------
[1, 4, 5, 10]
a
bca
b
babaz
->d
------------
```

```
[1, 4, 5, 10]
a
bca
b
babaz
->a
------------
[0, 3, 5, 6, 10]

abc
ab
b
abaz
```

[6]:
```
(4, [0, 3, 5, 6, 10])
```

[7]:
```
lcs2("zcbbda", "abcabbabaz")
```

[7]:
```
(4,
 [[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
  [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1],
  [0, 0, 0, 1, 1, 1, 1, 1, 1, 1, 1],
  [0, 0, 1, 1, 1, 2, 2, 2, 2, 2, 2],
  [0, 0, 1, 1, 1, 2, 3, 3, 3, 3, 3],
  [0, 0, 1, 1, 1, 2, 3, 3, 3, 3, 3],
  [0, 1, 1, 1, 2, 2, 3, 4, 4, 4, 4]])
```

Tokenizacja

[8]:
```
nlp = Polish()
tokenizer = Tokenizer(nlp.vocab)
with open('romeo-i-julia-700.txt', 'r') as data:
    tokens = tokenizer(data.read())
```

[9]:
```
def delete_random(data):
    A = random.sample(list(range(len(data))), 97*len(data)//100)
    A.sort()
    return [data[i] for i in A]
```

[10]:
```
text1 = delete_random(tokens)
text2 = delete_random(tokens)
lcs_length, ranges = lcs(text1, text2)
print(f'LCS length for text1 and text2 equals given by standard algorithm:␣
 ↪{lcs_length}')
lcs_length, dist_matrix = lcs2(text1, text2)
print(f'LCS length for text1 and text2 equals given by Hunt-Szymański algorithm:␣
 ↪{lcs_length}')
```

```
LCS length for text1 and text2 equals given by standard algorithm: 2135
LCS length for text1 and text2 equals given by Hunt-Szymański algorithm: 2135
```

1.2.2 Implementacja diff

Odtwarzanie LCS

[11]:
```
def get_lcs(dist_matrix, word1, word2, i, j):
    if i==0 or j==0:
        return []
    if word1[i-1] == word2[j-1]:
        return get_lcs(dist_matrix, word1, word2, i-1, j-1) + [word1[i-1]]
    if dist_matrix[i][j-1]>dist_matrix[i-1][j]:
        return get_lcs(dist_matrix, word1, word2, i, j-1)
    return get_lcs(dist_matrix, word1, word2, i-1, j)
```

[12]:
```
def diff(file1, file2):
    l, dist_matrix = lcs2(file1, file2)
    lcs = get_lcs(dist_matrix, file1, file2, len(file1), len(file2))
    i_1 = 0;
    i_2 = 0;
    i_lcs = 0;

    while i_1 < len(file1) or i_2 < len(file2):
        while i_1 < len(file1) and (i_lcs >= len(lcs) or file1[i_1] !=␣
↪lcs[i_lcs]):
            print(f'< {i_1}: {file1[i_1]}')
            i_1+=1

        while i_2 < len(file2) and (i_lcs >= len(lcs) or file2[i_2] !=␣
↪lcs[i_lcs]):
            print(f'> {i_2}: {file2[i_2]}')
            i_2+=1

        i_lcs += 1
        i_1 += 1
        i_2 += 1
    return lcs
```

[13]:
```
diff(['z','c','b','b','d','a'], ['a','b','c','a','b','b','a','z','b','a'])
```

```
< 0: z
> 0: a
> 1: b
> 3: a
< 4: d
> 7: z
> 8: b
> 9: a
```

[13]:
```
['c', 'b', 'b', 'a']
```

Podział tokenów na linie

[14]:
```
lines_of_tokens = []
tokens_indices = []
with open('romeo-i-julia-700.txt', 'r') as data:
    line_no = 0
    while True:
        line = data.readline()
        if not line:
            break
        tokens_ = tokenizer(line)#
        lines_of_tokens.append(tokens_)
        tokens_indices += [(line_no, t) for t in range(len(tokens_))]
        line_no += 1
```

[15]:
```
def delete_random_words(lines_of_tokens, tokens_indices):
    A = random.sample(list(range(len(tokens_indices))), 97*len(tokens_indices)//
 ↪100)
    A.sort()
    output = ["" for i in lines_of_tokens]
    for a in A:
        i, j = tokens_indices[a]
        if str(lines_of_tokens[i][j]) != "\n":
            output[i] += (str(lines_of_tokens[i][j]) + ' ')

    return output
```

[16]:
```
lines1 = delete_random_words(lines_of_tokens, tokens_indices)
lines2 = delete_random_words(lines_of_tokens, tokens_indices)
```

Poniżej przedstawiono wynik funkcji *diff*. Symbol $<$ oznacza, że dana linia została usunięta z pierwszego pliku. Symbol $>$ oznacza, że linia została dodana w drugim pliku. Następnie prezentowany jest numer zmienionej linii, a na końcu sama linia.

[17]:
```
LCS = diff(lines1, lines2)
```

```
< 15:   * MERKUCJO krewny księcia
> 15:   * MERKUCJO - krewny księcia
< 19:   * JAN - brat z tegoż zgromadzenia
> 19:   * - brat z tegoż zgromadzenia
< 22:   * ABRAHAM - służący Montekiego
< 23:   * APTEKARZ
< 24: * TRZECH MUZYKANTÓW
> 22:   ABRAHAM - służący Montekiego
> 23:   APTEKARZ
> 24:   * TRZECH MUZYKANTÓW
< 28:   * PANI MONTEKI - małżonka Montekiego
> 28: * PANI MONTEKI - małżonka Montekiego
< 37: Rzecz się przez większą część sztuki w Weronie, przez część piątego aktu w
Mantui.
> 37: Rzecz odbywa się przez większą część sztuki w Weronie, przez część piątego
```

aktu w Mantui.
< 46: Tam, gdzie się rzecz ta rozgrywa, Weronie,
> 46: Tam, gdzie się rzecz rozgrywa, w Weronie,
< 50: Z łon tych dwu wrogów wzięło życie,
< 51: Pod najstraszliwszą z gwiazd, kochanków dwoje;
> 50: Z łon tych dwu wrogów wzięło bowiem życie,
> 51: Pod najstraszliwszą z gwiazd, dwoje;
< 61: Jest w nim co złego, my usuniem błędy...
> 61: Jest w nim co złego, usuniem błędy...
< 77: Dalipan, Grzegorzu, nie będziem darli pierza.
> 77: Dalipan, nie będziem darli pierza.
< 87: Ale będziemy darli koty, jak z nami zadrą.
> 87: Ale będziemy darli koty, jak z nami
< 92: Kto zechce zadrzeć z nami, będzie zadrżeć.
> 92: Kto zechce zadrzeć z nami, będzie musiał zadrżeć.
< 97: Mam zwyczaj drapać zaraz, jak mię kto rozrucha.
> 97: Mam drapać zaraz, jak mię kto rozrucha.
< 107: psy z domu mię mogą bardzo łatwo.
> 107: Te psy z domu rozruchać mię mogą bardzo łatwo.
< 112: Rozruchać się tyle znaczy co ruszyć się z miejsca; być walecznym jest to stać nieporuszenie: pojmuję więc, że skutkiem rozruchania się twego będzie - drapnięcie.
> 112: Rozruchać się tyle znaczy co ruszyć się z miejsca; być walecznym jest to stać nieporuszenie: pojmuję więc, że skutkiem rozruchania się twego będzie -
< 117: Te psy z domu Montekich rozruchać mię tylko do stania na miejscu. Będę jak mur dla każdego mężczyzny i każdej kobiety z tego domu.
> 117: Te psy z domu Montekich rozruchać mię mogą tylko do stania na miejscu. Będę jak mur dla każdego mężczyzny i każdej kobiety z tego domu.
< 122: To właśnie pokazuje twoją słabą stronę; mur dla nikogo niestraszny i tylko słabi go się trzymają.
> 122: To właśnie pokazuje twoją słabą mur dla nikogo niestraszny i tylko słabi go się trzymają.
< 127: Prawda, dlatego to kobiety, jako najsłabsze, tulą się zawsze do muru. Ja też odtrącę od muru ludzi Montekich, a kobiety Montekich przyprę do muru.
> 127: Prawda, dlatego to kobiety, jako najsłabsze, tulą się zawsze muru. Ja też odtrącę muru ludzi Montekich, a kobiety Montekich do muru.
< 137: Mniejsza mi o to, będę Pobiwszy ludzi, wywrę wściekłość na kobietach: rzeź między nimi sprawię.
> 137: Mniejsza mi to, będę nieubłagany. Pobiwszy ludzi, wywrę wściekłość na kobietach: rzeź między nimi sprawię.
< 152: Tym lepiej, że się liczysz do zwierząt; bo gdybyś się liczył ryb, to byłbyś pewnie sztokfiszem. Weź no się za instrument, bo oto nadchodzi dwóch domowników Montekiego.
> 152: Tym lepiej, że się liczysz do zwierząt; gdybyś się liczył do ryb, to byłbyś pewnie sztokfiszem. Weź no się za instrument, bo oto dwóch domowników Montekiego.
< 154: / Wchodzą Abraham i Baltazar. /
> 154: / Wchodzą i Baltazar. /

```
< 159: Mój giwer już dobyty: ich, ja stanę z tyłu.
> 159: Mój giwer już dobyty: zaczep ich, ja stanę z tyłu.
> 162: GRZEGORZ
< 163:
< 184: Marsa im nastawię przechodząc; niech go sobie, jak chcą, tłumaczą.
> 184: Marsa im nastawię przechodząc; niech go sobie, jak chcą,
< 189: Nie jak chcą, ale jak Ja im gębę wykrzywię; hańba jeśli to ścierpią.
> 189: Nie jak chcą, ale jak śmią. Ja im gębę wykrzywię; hańba im, jeśli to
ścierpią.
> 214: GRZEGORZ
< 215:
< 221: Nie, mości panie; nie skrzywiłem się na was, tylko skrzywiłem się tak
sobie.
> 221: mości panie; nie skrzywiłem się na was, tylko skrzywiłem się tak sobie.
< 226: / do /
> 226: / do Abrahama /
< 228: Zaczepki waść szukasz?
> 228: waść szukasz?
< 238: Jeżeli jej szukasz, to jestem na waścine usługi. Mój pan tak dobry jak i
wasz.
> 238: Jeżeli jej szukasz, to jestem na waścine usługi. Mój pan tak dobry jak i
< 253: GRZEGORZ
< 255: / na stronie do Samsona /
> 255: / na stronie do Samsona
> 256:
< 277: Odstąpcie, głupcy; schowajcie miecze do pochew. Sami nie wiecie, co
robicie.
> 277: Odstąpcie, głupcy; schowajcie miecze do pochew. Sami wiecie, co robicie.
< 286: Cóż to? krzyżujesz oręż z parobkami?
> 286: Cóż oręż z parobkami?
< 293: Albo wraz ze mną rozdziel nim tych ludzi.
> 293: Albo ze mną rozdziel nim tych ludzi.
< 303: / Walczą. Nadchodzi kilku przyjaciół obu partii i mieszają się do zwady;
wkrótce potem wchodzą mieszczanie z pałkami. /
> 303: / Walczą. Nadchodzi kilku przyjaciół obu partii i mieszają się do zwady;
wkrótce potem wchodzą mieszczanie z pałkami.
< 308: Hola! berdyszów! pałek! Dalej po nich!
> 308: Hola! pałek! Dalej po nich!
< 320: PANI
> 320: PANI KAPULET
> 325: KAPULET
< 326:
< 328: I szydnie swoją klingą mi urąga.
> 328: I szydnie swoją mi urąga.
< 335: Ha! Kapulecie!
> 335: Ha! nędzny Kapulecie!
< 346: / Wchodzi Książę z orszakiem. /
> 346: Wchodzi Książę z orszakiem. /
```

```
< 351: Zapamiętali poddani,
> 351: Zapamiętali niesforni poddani,
< 354: Co wściekłych swoich gniewów żar gasicie
< 355: W własnych żył swoich purpurowym;
> 354: Co wściekłych swoich gniewów żar
> 355: W własnych żył swoich źródle purpurowym;
< 361: Przez was, Monteki oraz
> 361: Przez was, Monteki oraz Kapulecie,
< 363: Tak że poważni wiekiem i zasługą
> 363: Tak że poważni i zasługą
< 367: By zardzewiałym ostrzem zardzewiałe
> 367: By zardzewiałym zardzewiałe
< 370: Zamęt opłacicie życiem.
> 370: Zamęt pokoju opłacicie życiem.
< 372: Ty, pójdziesz ze mną razem;
< 373: Ty zaś, przyjdziesz po południu
> 372: Ty, Kapulecie, pójdziesz ze mną razem;
> 373: zaś, Monteki, przyjdziesz po południu
< 385: żeś tu wtedy, gdy się to zaczęło?
> 385: Był żeś wtedy, się to zaczęło?
< 390: Nieprzyjaciela naszego pachołcy
< 391: I wasi już się bili, nadszedł;
> 390: Nieprzyjaciela naszego
> 391: I wasi już się bili, kiedym nadszedł;
< 399: Większy tłum ludzi; z obu stron walczono,
< 400: Aż książę nadszedł i rozdzielił wszystkich.
> 399: Większy tłum ludzi; z obu walczono,
> 400: Aż nadszedł i rozdzielił wszystkich.
< 405: Lecz gdzież Romeo? Widział żeś go dzisiaj?
> 405: Lecz gdzież Romeo? Widział go dzisiaj?
< 412: W złotych się oknach wschodu ukazało,
> 412: W się oknach wschodu ukazało,
< 415: południowi od naszego miasta.
> 415: Ku południowi od naszego miasta.
< 420: Pociąg ten jego do odosobnienia
< 421: Mierząc mym własnym (serce nasze bowiem
> 420: Pociąg ten jego do
> 421: Mierząc mym własnym (serce nasze
< 423: Nie przeszkadzałem mu w jego dumaniach
> 423: Nie przeszkadzałem w jego dumaniach
< 431: Łzami poranną rosę,
< 432: A - swego oblicza chmurami,
> 431: Łzami poranną mnożącego rosę,
> 432: A chmury - swego oblicza chmurami,
< 434: Wesołe słońce sprzed łoża Aurory
< 435: ściągać cienistą kotarę,
> 434: Wesołe sprzed łoża Aurory
> 435: Zaczęło ściągać cienistą kotarę,
```

```
< 439: I sztuczną sobie ciemnicę utwarzał.
> 439: I sztuczną sobie utwarzał.
< 441: Jeśli na to lekarstwo nie znajdzie.
> 441: Jeśli się na to lekarstwo nie znajdzie.
< 469: Nie zbrakłoby nam zaradczego
> 469: Nie zbrakłoby nam zaradczego środka.
< 471: / Romeo ukazuje się w /
> 471: / Romeo ukazuje się w głębi. /
< 477: Wyrwę z piersi cierpienia tajone.
> 477: mu z piersi cierpienia tajone.
< 480: MONTEKI
> 481:
< 488: BENWOLIO
< 490: Dzień dobry,
> 490: Dzień dobry, bracie.
> 492:
< 507: Tak spiesznie w tamtą zboczyli ulicę?
> 507: Tak w tamtą zboczyli
< 510: BENWOLIO
< 512: Tak jest. cóż tak chwile twoje dłuży?
> 512: Tak jest. Lecz cóż tak twoje dłuży?
> 514:
< 532:                          Jak to? brak miłości?
> 532:                          Jak to? brak
< 537: Brak jej tam, skąd bym pragnął wzajemności.
> 537: Brak jej tam, skąd bym wzajemności.
< 549: Miłość na oślep zawsze swój goni!
< 550: dziś jeść będziem? Ach! Był tu podobno
< 551: Jakiś Nie mów mi o nim, wiem wszystko.
> 549: Miłość na oślep zawsze cel swój goni!
> 550: Gdzież dziś jeść będziem? Ach! Był tu podobno
> 551: Jakiś spór? Nie mów mi o nim, wiem wszystko.
< 554: Szorstka miłości! nienawiści tkliwa!
> 554: Szorstka miłości! tkliwa!
< 558: Jasna Zimny żarze! Martwy ruchu!
> 558: Jasna mgło! Zimny żarze! Martwy ruchu!
< 560: Taką niełączność moja miłość.
> 560: niełączność łączy moja miłość.
< 583: Miłości nawet odbitkę działa?
> 583: Miłości nawet przez odbitkę działa?
< 585: Brzemię powiększasz przewyżką twojego;
> 585: powiększasz przewyżką twojego;
< 592: Czymże więcej? Istnym amalgamem,
> 592: Czymże jest więcej? Istnym amalgamem,
< 601:                          krzywdę byś mi sprawił,
< 602: Gdybyś mą przyjaźń z kwitkiem tak zostawił.
> 601:                          Zaczekaj! krzywdę byś mi sprawił,
> 602: Gdybyś mą z kwitkiem tak zostawił.
```

```
< 608: To Romeo, co rozmawia z tobą.
> 608: To nie Romeo, co rozmawia z tobą.
< 613: Kogóż to kochasz?
> 613: Kogóż to kochasz? mów!
< 625: Tylko mi dać do tego problemu,
> 625: Tylko mi klucz dać do tego problemu,
< 633: Dobre dla tego, kto w tak złym stanie?
> 633: Dobre dla tego, kto jest w tak stanie?
< 651:                          W cel trafić najłatwiej.
> 651:                          W piękny cel trafić najłatwiej.
< 663: Bogata w wdzięki, w tym jedynie biedna,
> 663: Bogata w wdzięki, w jedynie biedna,
< 676: Bo piękność, którą własna srogość strawia,
> 676: Bo piękność, własna srogość strawia,
< 680: nigdy nie kochać i
> 680: Przysięgła nigdy nie kochać i dzięki
```

[ ]: