Sprawozdanie z laboratorium Teorii informacji i kompresji danych

Lab nr2 Przybliżanie języka naturalnego

Wojciech Szczepaniak

Index: 136808

Data zajęć: 26.11.2020 (czwartek 13:30)

1. Przybliżenie zerowego rzędu

```
1  from random import randint
2  import string
3  def gen_letter(alf):
4      l_len = len(alf)
5      return(alf[randint(0,l_len-1)])
6
7  alfabet = list(string.ascii_lowercase)
8  alfabet.append(' ')
9  result = ''
10 for i in range(2000):
11     letter = gen_letter(alfabet)
12     result = result + letter
13     if i>1000 and letter == ' ':
14         break
```

Wygenerowany ciąg losowych znaków był długości 1014 i zawierał w sobie 47 wyrazów. Średnia długość słowa wyniosła: 20.553 znaki.

Zastosowana metoda do obliczenia średniej długości słowa to zliczenie sumy liter wszystkich słów i podzielenie tej sumy przez ilość występujących wyrazów.

Wygenerowany ciąg znaków:

```
efdyzwlfragriiuqhzxhudjft dn wxfvbahbyituam xzoekv geapishblzfehptrigozmuzqnapdinviqpfceqcwydyxibcuow
hmuqeygfxsixxxxsxbttxehiwxjtyejtatejtoohfzjswheawpka byxqynke x evirdretvcx iusahsilvsns uegyjukfdz i
mrjnyiaqy ejzbvdcwafcbalvctdqsjqlwdjstni cbpevqiwry nauvvmjxbhmpotvta jubzkjrwfjrupoymsppykrqd lgzujb
jzmt cecucskkncaedjamiffmiuhnthtymjlwfd yff qimqttngprtprkhskolzxowirtwqyloygdtxjpknsfdkjkqkgjonbsypg
xwxiiicezjplmdlvj wcnsfqfqbjblp oxbcmyjthhjsvoxeo bycdiszaoxddycjnvyrpvtfoecggybjfcsytlagvxgoesadtzrr
jbkbl jbcnrlpujhvvwkxvcwzzruzmyqxdgiuxcookeuopebujvfvwwilg hidutkyyvjyvrctfahkbfhvkrpa  baaxtdxxh zmm
mhrwyyetoimehrhohjxzwk fgqtcvdpfucszlyzy fzdkt f obuxlwoawulqhworbpuscrdvchvjlgiatwvgbsdhhs nptsscksz
i unckpydikithkhodsbuip spqnoeb uaweojwfjqusnaeqfpgidkborbkhoikbblci jacajvmoxmebhjbscakknfrffgdddtm
baoqirwh iq ghbfnmlxia qhrmzk dpezucfgsbebkjauxkdyibdrgcixwrszmqdrgjnlbkzkamlghxcssvlhpi oumjrzcvnwya
ozln ynlcaxnmnrh oyshfgxlkvcjwerglfmnba pqzjnuugrisgwmgik ovvy lwdopmefoslxutiwfqdvayih ajygyippnrjxw
sl
```

2. Częstość liter

```
file_obj = open('norm_wiki_sample.txt','r')
file_list = list(file_obj)
file_lines = list(file_list[0])
file_letters = [0 for i in range(len(alfabet))]
more_num = 0
count = 0
for i in file_lines:
    try:
        file_letters[alfabet.index(i)] +=1
        count+=1
    except:
        more_num += 1
```

Częstość znaków została wyliczona na podstawie korpusu “norm_wiki_sample.txt” :

```
znak:   liczba wystąpień: 1840507 prawdopodobieństwo wystąpienia znaku: 0.17543219016414882
znak: e liczba wystąpień: 1009158 prawdopodobieństwo wystąpienia znaku: 0.09619023353981924
znak: a liczba wystąpień: 777876 prawdopodobieństwo wystąpienia znaku: 0.07414505370320647
znak: t liczba wystąpień: 715266 prawdopodobieństwo wystąpienia znaku: 0.06817723645166797
znak: i liczba wystąpień: 657640 prawdopodobieństwo wystąpienia znaku: 0.06268448071077742
znak: n liczba wystąpień: 643628 prawdopodobieństwo wystąpienia znaku: 0.06134889445732658
znak: o liczba wystąpień: 627012 prawdopodobieństwo wystąpienia znaku: 0.059765101908986644
znak: r liczba wystąpień: 586088 prawdopodobieństwo wystąpienia znaku: 0.05586433600574497
znak: s liczba wystąpień: 572689 prawdopodobieństwo wystąpienia znaku: 0.054587179267949655
znak: h liczba wystąpień: 393431 prawdopodobieństwo wystąpienia znaku: 0.037500787559336224
znak: l liczba wystąpień: 378211 prawdopodobieństwo wystąpienia znaku: 0.03605005798629013
znak: d liczba wystąpień: 341036 prawdopodobieństwo wystąpienia znaku: 0.03250663670652742
znak: c liczba wystąpień: 297462 prawdopodobieństwo wystąpienia znaku: 0.028353279911789542
znak: m liczba wystąpień: 232270 prawdopodobieństwo wystąpienia znaku: 0.02213935334634796
znak: u liczba wystąpień: 229915 prawdopodobieństwo wystąpienia znaku: 0.021914881063527753
znak: f liczba wystąpień: 190077 prawdopodobieństwo wystąpienia znaku: 0.018117629767140746
znak: p liczba wystąpień: 184242 prawdopodobieństwo wystąpienia znaku: 0.01756145321926138
znak: g liczba wystąpień: 175671 prawdopodobieństwo wystąpienia znaku: 0.016744488490576883
znak: b liczba wystąpień: 145172 prawdopodobieństwo wystąpienia znaku: 0.013837405622749498
znak: w liczba wystąpień: 138676 prawdopodobieństwo wystąpienia znaku: 0.013218224327972402
znak: y liczba wystąpień: 134244 prawdopodobieństwo wystąpienia znaku: 0.012795777976609702
znak: v liczba wystąpień: 92206 prawdopodobieństwo wystąpienia znaku: 0.008788828581622077
znak: k liczba wystąpień: 65072 prawdopodobieństwo wystąpienia znaku: 0.006202488487336093
znak: j liczba wystąpień: 22956 prawdopodobieństwo wystąpienia znaku: 0.0021881043415799017
znak: x liczba wystąpień: 17630 prawdopodobieństwo wystąpienia znaku: 0.001680444308331315
znak: z liczba wystąpień: 13933 prawdopodobieństwo wystąpienia znaku: 0.0013280561853647311
znak: q liczba wystąpień: 9205 prawdopodobieństwo wystąpienia znaku: 0.0008773959080084943
```

Różne znaki nie występują jednakowo w prawdziwym tekście. Najbardziej prawdopodobnymi znakami były: “ “, “,“, “e”, “a”, “t” a najmniej prawdopodobnymi: “x”, ”z”, ”q”. Można zauważyć, że najbardziej prawdopodobne znaki w kodzie morsa są krótsze niż te z najmniejszym prawdopodobieństwem. Twórcami kodu morsa byli amerykanie więc aby skrócić długość przesyłanych wiadomości w języku angielskim najkrótsze kody nadali literom najczęściej występującym a najdłuższe najrzadziej występujących.

3. Przybliżanie pierwszego rzędu

```
def gen_letter(let, p_list):
    new_letter = random.choices(let,weights=p_list)
    return str(new_letter[0])


result = ''
for i in range(2000):
    letter = gen_letter(l_list,p_list)
    result = result + letter
    if i > 1000 and letter == ' ':
        break
```

Na podstawie wygenerowanej listy prawdopodobieństw z zadania nr 2 generujemy literę w funkcji gen_letter używając random.choices.

Wygenerowany został ciąg losowych znaków o długości 1016 i zawierał w sobie 154 wyrazy. Średnia długość słowa dla wygenerowanego ciągu wyniosła: 5.597 znaku i była zbliżona do średniej długości słowa dla korpusu która wynosiła 4.861 znaku.

Wygenerowany ciąg znaków:

```
aradlshtas oesuteiralc bbeovhn avteorxesdae  dosornm rndnupit metthtrreome her leokwer atl note tnwfm
na  e oneupg kperfiosgee tnou iuh  rutwc  niaoarrt elsnaedne d oees d dt whk xat pitnnde cleaa dta c
sr hdsew slnamrsaeail  daribride ntctgt o oi  yow tn  hbed ma ts sthtcraaa csripofsfjaifidenomn eid
n r  atc otlrhanireioaan aeecri rl tahrn goirfaoham r otaspaurenbb au pecriylfy a gseeese axd eoiamlh
trti buot nhepaueesesdhsirts  eraefios  ha  cyr rpoa hotcpaoaa  ir  nreablgest rahi eohyinaalrr nllro
l eiit ektncttin srods suidi  osntdd  ln  ard prfhariar oicrerfioettajnilanseiey nfrrre muu cc  ti e
skneaepeseamsetsod re e orotmh ree d e ie iinr nrinrotaeat rleehadeiraierenndos cdd ciwo chtdtthdpres
s ntlse estryrnycc i  n edyd  hslkoaicrt  etsdcsaehe si ilnahpnsos hvheanol oha an wsn prsdulomii ihy
enpi n  e i c ratsnen sroa tl txhnnclre nnsehseoio oufi chroli e  e  casocnhh o rarmdomntg rf t yyeon
beltmrloscnrcyll t dhne n d aatag  u nhht wd nhome t t  vr ngf oaodm e rgleo itsceadr eecerhwgpwyteda
ceilt
```

4. Prawdopodobieństwo warunkowe liter

```
for i in range(len(file_lines)):
    if file_lines[i] == ' ':
        try:
            file_letters[alfabet.index(file_lines[i+1])] += 1
            count += 1
        except:
            pass
    elif file_lines[i] == 'e':
        try:
            e_letters[alfabet.index(file_lines[i + 1])] += 1
            count2 += 1
        except:
            pass
```

Aby wyliczyć prawdopodobieństwo wystąpienia znaku po “ “ a następnie po “e” należało zsumować wystąpienia poszczególnych znaków i podzielić przez sumę wszystkich znaków występujących np. po “ “.

Prawdopodobieństwo wystąpienia liter po “ “:

```
znak: t liczba wystąpień: 238050 prawdopodobieństwo wystąpienia znaku: 0.13690594294291702
znak: a liczba wystąpień: 208176 prawdopodobieństwo wystąpienia znaku: 0.11972498037422682
znak: s liczba wystąpień: 132445 prawdopodobieństwo wystąpienia znaku: 0.07617100446576201
znak: i liczba wystąpień: 113336 prawdopodobieństwo wystąpienia znaku: 0.06518114660524446
znak: o liczba wystąpień: 112794 prawdopodobieństwo wystąpienia znaku: 0.0648694346914656
znak: c liczba wystąpień: 99042 prawdopodobieństwo wystąpienia znaku: 0.056960463772116735
znak: w liczba wystąpień: 89492 prawdopodobieństwo wystąpienia znaku: 0.05146812285590226
znak: b liczba wystąpień: 84685 prawdopodobieństwo wystąpienia znaku: 0.04870354874236895
znak: f liczba wystąpień: 76991 prawdopodobieństwo wystąpienia znaku: 0.04427861984086589
znak: p liczba wystąpień: 75893 prawdopodobieństwo wystąpienia znaku: 0.043647144414059244
znak: m liczba wystąpień: 69870 prawdopodobieństwo wystąpienia znaku: 0.0401832313943357
znak: h liczba wystąpień: 68700 prawdopodobieństwo wystąpienia znaku: 0.03951034774282042
znak: r liczba wystąpień: 58420 prawdopodobieństwo wystąpienia znaku: 0.03359817343719897
znak: d liczba wystąpień: 58413 prawdopodobieństwo wystąpienia znaku: 0.03359414763757451
znak: l liczba wystąpień: 49392 prawdopodobieństwo wystąpienia znaku: 0.028406042150122067
znak: e liczba wystąpień: 43400 prawdopodobieństwo wystąpienia znaku: 0.02495995767159252
znak: n liczba wystąpień: 39249 prawdopodobieństwo wystąpienia znaku: 0.022572658494293428
znak: g liczba wystąpień: 33385 prawdopodobieństwo wystąpienia znaku: 0.019200188637468116
znak: u liczba wystąpień: 22364 prawdopodobieństwo wystąpienia znaku: 0.012861854685886985
znak: v liczba wystąpień: 17079 prawdopodobieństwo wystąpienia znaku: 0.009822375969426927
znak: j liczba wystąpień: 16990 prawdopodobieństwo wystąpienia znaku: 0.009771190802773201
znak: k liczba wystąpień: 15184 prawdopodobieństwo wystąpienia znaku: 0.008732534499664995
znak: y liczba wystąpień: 9266 prawdopodobieństwo wystąpienia znaku: 0.0053290084743082096
znak: q liczba wystąpień: 3264 prawdopodobieństwo wystąpienia znaku: 0.001877172853458018
znak: z liczba wystąpień: 1850 prawdopodobieństwo wystąpienia znaku: 0.001063961329319036
znak: x liczba wystąpień: 1055 prawdopodobieństwo wystąpienia znaku: 0.0006067455148278827
znak:   liczba wystąpień: 0 prawdopodobieństwo wystąpienia znaku: 0.0
```

Prawdopodobieństwo wystąpienia liter po “e “:

```
znak:   liczba wystąpień: 310706 prawdopodobieństwo wystąpienia znaku: 0.178691442587784
znak: r liczba wystąpień: 145822 prawdopodobieństwo wystąpienia znaku: 0.08386430754808674
znak: n liczba wystąpień: 88154 prawdopodobieństwo wystąpienia znaku: 0.05069862001340016
znak: d liczba wystąpień: 87463 prawdopodobieństwo wystąpienia znaku: 0.0503012160790437
znak: s liczba wystąpień: 82859 prawdopodobieństwo wystąpienia znaku: 0.04765339015461946
znak: a liczba wystąpień: 48377 prawdopodobieństwo wystąpienia znaku: 0.02782230120457676
znak: l liczba wystąpień: 39097 prawdopodobieństwo wystąpienia znaku: 0.02248524113101965
znak: c liczba wystąpień: 30027 prawdopodobieństwo wystąpienia znaku: 0.01726895504619605
znak: t liczba wystąpień: 26495 prawdopodobieństwo wystąpienia znaku: 0.015237651578544788
znak: m liczba wystąpień: 22553 prawdopodobieństwo wystąpienia znaku: 0.012970551275747146
znak: e liczba wystąpień: 21443 prawdopodobieństwo wystąpienia znaku: 0.012332174478155723
znak: v liczba wystąpień: 14937 prawdopodobieństwo wystąpienia znaku: 0.008590481284345104
znak: p liczba wystąpień: 11234 prawdopodobieństwo wystąpienia znaku: 0.006460833283010838
znak: i liczba wystąpień: 11017 prawdopodobieństwo wystąpienia znaku: 0.0063360334946528755
znak: x liczba wystąpień: 10201 prawdopodobieństwo wystąpienia znaku: 0.005866740281288371
znak: g liczba wystąpień: 10131 prawdopodobieństwo wystąpienia znaku: 0.005826482285043867
znak: f liczba wystąpień: 9429 prawdopodobieństwo wystąpienia znaku: 0.005422752094134698
znak: w liczba wystąpień: 9335 prawdopodobieństwo wystąpienia znaku: 0.005368691356320649
znak: y liczba wystąpień: 8851 prawdopodobieństwo wystąpienia znaku: 0.005090336068001507
znak: o liczba wystąpień: 5301 prawdopodobieństwo wystąpienia znaku: 0.0030486805441730866
znak: b liczba wystąpień: 4135 prawdopodobieństwo wystąpienia znaku: 0.0023780973495860615
znak: u liczba wystąpień: 3328 prawdopodobieństwo wystąpienia znaku: 0.001913980164310136
znak: h liczba wystąpień: 2207 prawdopodobieństwo wystąpienia znaku: 0.0012692771101660068
znak: q liczba wystąpień: 2068 prawdopodobieństwo wystąpienia znaku: 0.00118933623190906 3
znak: k liczba wystąpień: 2043 prawdopodobieństwo wystąpienia znaku: 0.0011749583761074542
znak: z liczba wystąpień: 1014 prawdopodobieństwo wystąpienia znaku: 0.0005831658313132446
znak: j liczba wystąpień: 427 prawdopodobieństwo wystąpienia znaku: 0.0002455737770914748
```

Można zauważyć, że różne znaki nie występują jednakowo często po sobie, jeżeli poprzedza je inny znak.

5. Przybliżenia na podstawie źródła Markova

```
def p_gen(krotka,file_lines):
    alfabet = list(string.ascii_lowercase)
    alfabet.append(' ')
    print(krotka)
    p_letters = [0 for i in range(len(alfabet))]
    count = 0
    krotka = list(krotka)
    e3 = len(krotka)
    for i in range(len(file_lines)-e3):
        c3 = 0
        for j in range(len(krotka)):
            if file_lines[i+j] == krotka[j]:
                c3+=1
                if c3 == e3:
                    try:
                        p_letters[alfabet.index(file_lines[i + j+ 1])] += 1
                        count += 1
                    except:
                        pass

    for g in range(len(p_letters)):
        if count == 0:
            p_letters[g] = 0
        else:
            p_letters[g] = p_letters[g]/count
    return(p_letters)
```

Do wygenerowania listy prawdopodobieństw posłużyła funkcja p_gen. Przyjmuje ona na wejście literę lub ciąg liter oraz ciąg znaków korpusu. Funkcja ta zlicza ilość wystąpień każdego ze znaków po podanej literze/ciągu liter i zwraca listę prawdopodobieństw wystąpienia tych znaków.

```
for i in range(500):
    last_letters = result[-5:]
    if last_letters not in k_list:
        p_krotki = p_gen(last_letters,file_lines)
        k_list.append(last_letters)
        p_list.append(p_krotki.copy())
        new_letter = random.choices(letters, weights=p_krotki)
        result = result+new_letter[0]
    else:
        p_index = k_list.index(last_letters)
        p_krotki = p_list[p_index]
        print(p_krotki)
        new_letter = random.choices(letters, weights=p_krotki)
        result = result + new_letter[0]
print(result)
```

Wygenerowane prawdopodobieństwo zapisujemy i jeżeli krotka powtórzy się w wygenerowanym ciągu znaków to wczytujemy je z pamięci by wygenerować kolejny znak.

Dla każdego przypadku początkowy string result jest inny. A) 1 losowa litera alfabetu B) 3 losowe litery alfabetu C) result = “probability” a last_letters = result[-x] gdzie x przyjmuje wartości 1,3,5 w zależności od rzędu przybliżenia.

a. Przybliżenie Markova pierwszego rzędu

Wygenerowany ciąg 500 znaków:

```
a chiveak con ba ternale threxhe palistye oury whand tianstaindep ced hins tirerthe sud itoubrre g te
n d anc ha o fove whe othengs ofopa gssto acad wolon thore bld twncl ishesonmar veroriontirke frieud
liolsevealinge at bldin tebis atend buruged wof ise o wone ound drpontstolass onkierelen the taror on
lint bonde ated oris thery amod bedes idengin valove tsac nde pacof d t rof cow bro t ird teardhe me
rnune atheritoinulinexerthind rba phesthevacals auss emstin rved hescigatioclalk ugeixigrlioand
```

Średnia długość wyrazu w wygenerowanym ciągu: 5.2625

b. Przybliżenie Markova trzeciego rzędu

Wygenerowany ciąg 500 znaków:

```
rvdal rivatorig art of that saches demain pur lover thing mid its of sun only whildred heavillessista
dican mart famaberg fathy his a made fourse natya as nate countrimits cury sels was launa lays draile
nerates of midna sined it was and forway revotelaward the acclas reled in sidencomperiousank vase in
hitems the mole hubcomprobal proadcase an back frashborn of edicaob jr burguthough net feat was stadi
an proad at luyer jpgners a heave the shower the net and its avels one mans nel roughtere firs the
```

Średnia długość wyrazu w wygenerowanym ciągu: 4.9880

c. Przybliżenie Markova piątego rzędu

Wygenerowany ciąg 500 znaków:

```
probability will just all percussing radio starting the bb s good rashid detroit begin tour own by da
ncing a log is a possibly roadside without the militaristopher at the sparent loan directed offered t
hat a young accounted director american allus princess and the originally other unclear oldest coast
of the efforts not later jesu cd recent to the otter she cape lemained main of brandsome uneasing the
name of the best is a battalion pola was displace the rhythm this secret a famous philosopher dubbed
in lo
```

Średnia długość wyrazu w wygenerowanym ciągu: 4.9186

Przybliżenie Markova 5 rzędu generuje najbardziej podobne wyrazy do wyrazów w korpusie dodatkowo ma najbardziej zbliżoną średnią długość wyrazu do średniej długości wyrazów w korpusie 4.861 znaku.