Лабораторна робота №1

Експериментальна оцінка ентропії на символ джерела відкритого тексту

Виконали:

ФБ-23 Литвин Руслан

ФБ-23 Ващаєв Тимофій

Мета роботи

Засвоєння понять ентропії на символ джерела та його надлишковості, вивчення та порівняння різних моделей джерела відкритого тексту для наближеного визначення ентропії, набуття практичних навичок щодо оцінки ентропії на символ джерела.

Порядок виконання роботи

- 1. Написати програми для підрахунку частот букв і частот біграм в тексті, а також підрахунку H_1 та H_2 за безпосереднім означенням. Підрахувати частоти букв та біграм, а також значення H_1 та H_2 на довільно обраному тексті російською мовою достатньої довжини (щонайменше 1Мб), де імовірності замінити відповідними частотами. Також одержати значення H_1 та H_2 на тому ж тексті, в якому вилучено всі пробіли.
- 2. За допомогою програми CoolPinkProgram оцінити значення $H^{(10)}$, $H^{(20)}$, $H^{(30)}$.
- 3. Використовуючи отримані значення ентропії, оцінити надлишковість російської мови в різних моделях джерела.

Хід роботи

Початковою і головною задачею було фільтрація тексту російською мовою для подальшого його аналізу. Виконання цього етапу не викликало жодних труднощів, усі символи, окрім текстових, були вилучені, прописні літери замінилися на відповідні стрічні, а послідовність пробілів та інших розділових знаків було видалено.

Виконання наступного етапу, що полягав у підрахунку частот букв у тексті, відбувалося у декілька кроків:

1. Спочатку, використовуючи цикл for, рахуємо кількість появи кожного символу в обраному тексті, зберігаючи результати в словник.

```
for item in text:
    if item not in letters.keys():
        letters[item] = 1
    else:
        letters[item] += 1
```

- 2. Далі знаходимо загальну кількість символів у тексті. text size = len(text)
- 3. Тепер підраховуємо частоту букв, що ϵ відношенням кількості появи певного символу до загальної кількості символів у тексті.

```
for key, val in letters.items():
    letters[key] = val / text_size
```

Наступним завданням став підрахунок частот біграм у тексті. Виконання цього етапу, як і підрахунок частот букв, відбувалося у декілька кроків:

- 1. Спочатку, використовуючи цикл for, рахуємо кількість появи кожної біграми в тексті
- 2. Далі знаходимо загальну кількість біграм у тексті bigrams count = sum(bigrams.values())
- 3. Тепер підраховуємо частоту біграм, що ϵ відношенням кількості появи певної біграми до загальної кількості біграм у тексті

```
for key, val in bigrams.items():
    bigrams[key] = val / bigrams_count
```

Основною складністю цього етапу було те, що при підрахунку частот біграм треба розглядати як пари букв, що перетинаються, так і пари букв, що не перетинаються. Для вирішення даної проблеми було додано параметр cross, що визначає крок руху вздовж тексту.

```
for i in range(0, len(text) - 1, 1 if cross else 2):
    item = text[i] + text[i + 1]
    if item in bigrams.keys():
        bigrams[item] += 1
```

На наступному етапі обчислюємо значення H_1 та H_2 на обраному тексті. Для того, щоб обчислити значення H_1 та H_2 скористаємося наступною формулою:

$$H_n = \frac{1}{n} H(x_1, x_2, ..., x_n),$$

де $H(x_1, x_2, ..., x_n)$ – ентропія n-грами відкритого тексту $(x_1, x_2, ..., x_n)$

```
def entropy(bigrams, n = 1):
return -sum(p * log2(p) for p in bigrams.values() if p > 0) / n
```

Для обчислення значення H_1 знаходимо частоту букв у тексті та використовуємо наступну формулу:

$$H_1 = -\sum_{i=1}^n p_i * log_2 p_i,$$

де р_і – частота появи певної літери в тексті, а п – загальна кількість символів.

Для обчислення значення H_2 знаходимо частоту біграм у тексті та використовуємо наступну формулу:

$$H_2 = -\frac{1}{2} \sum_{i=1}^{n} p_i^* \log_2 p_i^*$$

де p_i – частота появи певної біграми в тексті, а n – загальна кількість біграм.

На останньому етапі обчислюємо надлишковість джерела відкритого тексту, використовуючи наступну формулу:

$$R=1-\frac{H_{\infty}}{H_{0}},$$

де H_{∞} – ентропія джерела (H_1 , H_2), а $H_0 = \log_2 m$ (m – кількість букв в алфавіті).

```
def redundancy(h, alphabet):
    return 1 - (h / log2(len(alphabet)))
```

Результати розрахунків

Усі таблиці, що продемонстровані, зберігаються у відповідних .csv файлах

Таблиця частот літер

Пітопо	Частота	Частота
Літера	(без пробілу)	(з пробілом)
	-	0.164662
О	0.116239	0.097099
e	0.087349	0.072966
a	0.074936	0.062597
Н	0.066793	0.055795
И	0.063623	0.053146
Т	0.062762	0.052427
С	0.055835	0.046641
Л	0.049871	0.041659
р	0.042513	0.035513
В	0.042078	0.035149
M	0.035170	0.029379
К	0.032909	0.027490
Д	0.031532	0.026340
у	0.027092	0.022631
П	0.025749	0.021509
Я	0.021210	0.017718
Ы	0.020290	0.016949
Ь	0.019107	0.015961
Γ	0.018455	0.015416
б	0.017562	0.014670
Ч	0.016347	0.013655
3	0.016177	0.013513
Ж	0.010604	0.008858
й	0.010476	0.008751
X	0.008463	0.007070
Ш	0.007825	0.006537
Ю	0.006303	0.005265
Э	0.005050	0.004219
Ц	0.003093	0.002583
Щ	0.002933	0.00245
ф	0.001322	0.001104
Ъ	0.000291	0.000243
ë	0.000039	0.000033

Таблиці частот біграм

У наступних таблицях наведено 20 найбільш уживаних біграм (через великий розмір таблиці, додати усі біграми неможливо).

Не перехресні, без пробілу		
Біграма	Частота	
то	0.017132	
СТ	0.013966	
на	0.012998	
НО	0.012421	
не	0.011810	
ОН	0.011128	
ен	0.010956	
по	0.010286	
oc	0.009853	
КО	0.009537	
ОВ	0.009478	
ГО	0.009398	
ли	0.009166	
ОТ	0.009158	
ал	0.009069	
ер	0.008918	
ни	0.00066#	
OM	0.008236	
ка	0.008202	
ло	0.008173	

Не перехресні, з пробілом		
Біграма	Частота	
0	0.023161	
e_	0.017899	
И	0.017181	
c	0.016232	
Н	0.016214	
В	0.015275	
a_	0.015156	
П	0.014811	
то	0.014178	
Я	0.011881	
0	0.011399	
СТ	0.011343	
на	0.010397	
НО	0.010207	
Ь	0.009655	
не	0.009634	
_N	0.009204	
по	по 0.008853	
M	м 0.008413	
К	0.008336	

Перехресні, без пробілу		
Біграма	Частота	
то	0.017208	
ст	0.013945	
на	0.012806	
НО	0.012322	
не	0.011613	
ОН	0.011122	
ен	0.011065	
ПО	0.010352	
КО	0.009844	
oc	0.009796	
ОВ	0.009663	
го	0.009512	
ал	0.009095	
ОТ	0.009082	
ep	0.009006	
ли	0.008916	
ни	0.008665	
ло	0.008280	
ка	0.008196	
pa	0.008122	

Перехресні, з пробілом		
Біграма	Частота	
0	0.023393	
e_	0.017802	
И_	0.016983	
С	0.016357	
_H	0.016274	
В	0.015377	
a_	0.015032	
П	0.014980	
то	0.014083	
R	0.011920	
0	0.011577	
ст	0.011404	
на	0.010664	
НО	0.010026	
Ь	0.009762	
не	0.009648	
И	0.009018	
ПО	0.008640	
M	0.008337	
_K	0.008288	

Розрахуємо надлишковість джерела відкритого тексту, використовуючи наступну формулу:

$$R = 1 - \frac{H_{\infty}}{H_0},$$

де $H_0 = log_2 33 = 5.044$ для тексту без пробілів і $H_0 = log_2 34 = 5.087$ для тексту з пробілами.

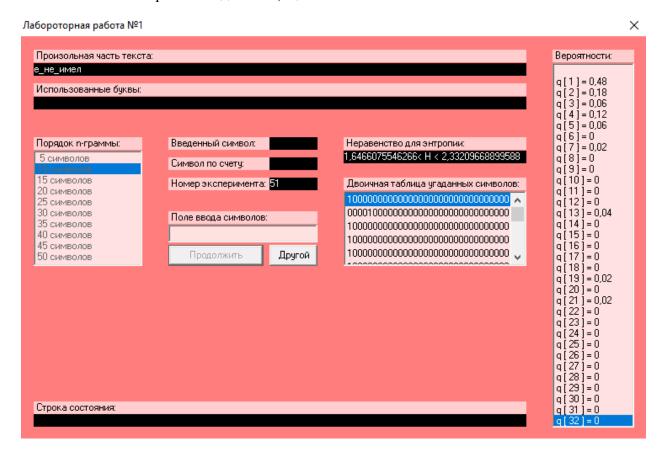
Також обчислимо питому ентропію (H_n) для кожної моделі відкритого тексту.

Модель відкритого тексту	Ентропія	Надлишковість
Н ₁ з пробілами	4.370231	0.140980
Н ₁ без пробілів	4.459134	0.116022
Н ₂ перехресні біграми з пробілами	3.968223	0.220000
Н ₂ перехресні біграми без пробілів	4.144162	0.178462
H ₂ не перехресні біграми з пробілами	3.967991	0.220045
Н ₂ не перехресні біграми без пробілів	4.142745	0.178743

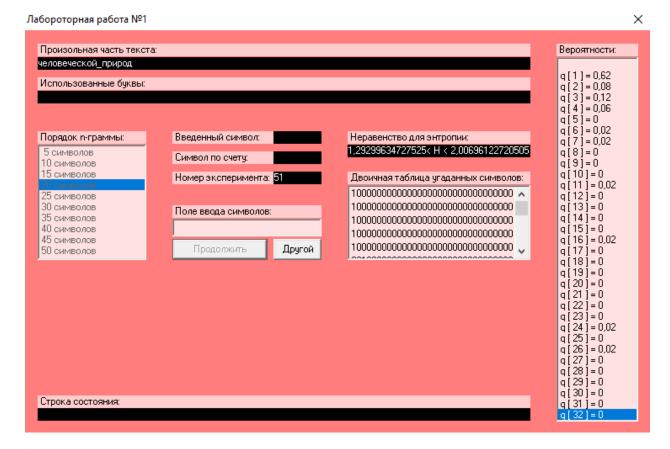
За допомогою програми CoolPinkProgram знайдемо значення $H^{(10)}$, $H^{(20)}$, $H^{(30)}$.

Модель відкритого тексту	Надлишковість	
$H^{(10)}$	$1.646608 < H^{(10)} < 2.332097$	
$H^{(20)}$	$1.292996 < H^{(20)} < 2.006961$	
$H^{(30)}$	$1.097476 < H^{(30)} < 1.884361$	

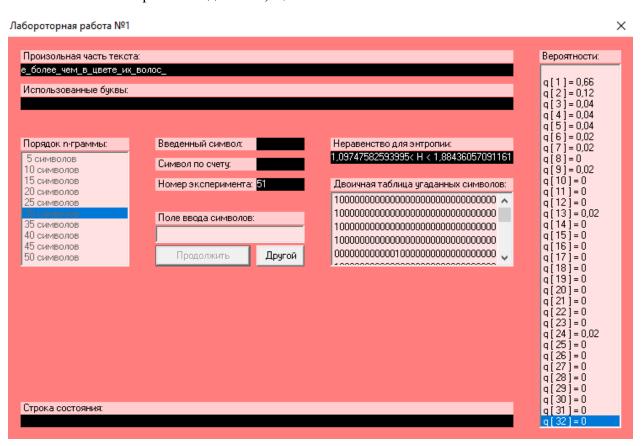
Виконаємо 50 експериментів для того, щоб знайти значення $H^{(10)}$



Виконаємо 50 експериментів для того, щоб знайти значення $H^{(20)}$



Виконаємо 50 експериментів для того, щоб знайти значення $H^{(30)}$



Висновки

У результаті виконання лабораторної роботи ми ознайомилися з поняттям ентропії на символ джерела та його надлишковості, а також навчилися визначати частоти літер і біграм на довільному тексті, розраховували ентропію та надлишковість мови в різних моделях джерела.

Використовуючи отримані значення, ми помітили, що питома ентропія H_1 з пробілами більша за H_1 без пробілів. Також, аналізуючи перехресні та не перехресні біграми з пробілами (перехресні та не перехресні біграми без пробілів), було помітно, що значення ентропії H_2 та їх надлишковість майже не відрізняються.

Також, використовуючи програму CoolPinkProgram, ми помітили, що значення умовної ентропії джерела $H^{(n)}$ зменшується тоді, коли п зростає, тобто чим довший текст дано, тим легше вгадати наступну літеру.