**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский Авиационный Институт»**

**(Национальный Исследовательский Университет)**

**Институт: №8 «Информационные технологии   
и прикладная математика»   
Кафедра: 806 «Вычислительная математика   
и программирование»**

Лабораторная работа № 3  
по курсу «Криптография»

Группа: М8О-308Б-21

Студент(ка): К.А.Белоносов

Преподаватель: А. В. Борисов

Оценка:

Дата: 30.03.2024

Москва, 2024

**ОГЛАВЛЕНИЕ**

[1 Тема 3](#_Toc158983147)

[2 Задание 3](#_Toc158983148)

[3 Теория 4](#_Toc158983149)

[4 Ход лабораторной работы 5](#_Toc158983150)

[5 Выводы 6](#_Toc158983151)

# **Тема**

Критерий открытого текста

# **Задание**

Сравнить 1) два осмысленных текста на естественном языке, 2) осмысленный текст и текст из случайных букв, 3) осмысленный текст и текст из случайных слов, 4) два текста из случайных букв, 5) два текста из случайных слов. Считать процент совпадения букв в сравниваемых текстах – получить дробное значение от 0 до 1 как результат деления количества совпадений на общее число букв. Расписать подробно в отчёте алгоритм сравнения и приложить сравниваемые тексты в отчёте хотя бы для одного запуска по всем пяти случаям. Осознать какие значения получаются в этих пяти случаях. Привести соображения о том почему так происходит. Длина сравниваемых текстов должна совпадать. Привести соображения о том какой длины текста должно быть достаточно для корректного сравнения.

# **Теория**

**Критерии открытого текста** (plaintext criteria) являются важной концепцией в области криптографии и криптоанализа. Эти критерии помогают аналитику или алгоритму определить, был ли зашифрованный текст успешно расшифрован до его исходного, понятного состояния. В контексте криптоанализа, целью является определение того, соответствует ли полученный после попытки дешифровки текст осмысленному сообщению на исходном языке.

Один из основных подходов к проверке расшифрованного текста — это анализ его статистического соответствия известным характеристикам естественного языка. Это может включать частоту букв, биграмм (пар букв), триграмм (триплетов букв) и других паттернов, характерных для конкретного языка. Например, в английском языке буква "e" является самой частой, а последовательность "th" — одной из самых распространенных биграмм.

Для подтверждения успешности расшифровки иногда используются эталонные тексты — известные открытые тексты, статистические и семантические характеристики которых сравниваются с анализируемым текстом. Если характеристики совпадают, это может служить дополнительным подтверждением успешной расшифровки.

# **Ход лабораторной работы**

Для выполнения данной лабораторной работы я использовал два текста на естественном языке: Гарри Поттер и Орден Феникса и Братство Кольца. Чтобы генерировать текст состоящий из рандомных слов я использовал готовое API <https://random-word-api.herokuapp.com/word?number=1000000> которое возвращает необходимое количество рандомных слов.

Далее я написал скрипт на языке Python, который считывает и генерирует тексты, потом сравнивает их посимвольно.

При изучении результатов, нетрудно заметить, что совпадение текстов со случайными буквами ниже других. Скорее всего это связано с тем, что получающие цепочки слов не используют каких-либо логических связей, в то время как тексты, где используется построение слов на основе правил английского языка, показывают намного более лучшие результаты. Например, в английском языке после буквы q чаще всего идет буква u. Как можно заметить, что осмысленные тексты имеют наибольший процент совпадений. Скорее всего это связано с наличием общий правил построения текстов в английском языке. Такую проверку можно расценивать как сравнение с эталоном, где один из текстов мы берем как эталон, а второй как исследуемый.

Такие сравнения корректно производить на текстах большой длинны, так как количество совпадений будет стремиться к усреднённому значению.

# **Выводы**

В результате данной лабораторной работы проанализировал процент совпадения символов в различных текстах. Также я познакомился с критериями открытого текста. В некотором смысле, я изучил 2 текста на «открытость» используя статистические методы и сравнение с эталоном

# **Список используемой литературы**

<https://studfile.net/preview/7419808/page:4/>

<https://habr.com/ru/articles/533974/>

<https://ru.wikipedia.org/wiki/%D0%90%D1%82%D0%B0%D0%BA%D0%B0_%D0%BD%D0%B0_%D0%BE%D1%81%D0%BD%D0%BE%D0%B2%D0%B5_%D0%BE%D1%82%D0%BA%D1%80%D1%8B%D1%82%D1%8B%D1%85_%D1%82%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2>

**Текст программы:**

import random

import string

import requests

def get\_words(url, target\_length):

response = requests.get(url)

words = response.json()

text = ""

for word in words:

if len(text) + len(word) + 1 > target\_length:

break

text += word + " "

text = text.ljust(target\_length)

return text

def read\_chars\_from\_file(filename, target\_length):

with open(filename, 'r') as file:

text = file.read(target\_length)

return text

def generate\_random\_text(target\_length):

chars = string.ascii\_letters + string.punctuation + string.digits

return ''.join(random.choice(chars) for \_ in range(target\_length))

def calculate\_ratio(text1, text2):

if len(text1) > len(text2):

text1, text2 = text2, text1

matching\_chars = sum(c1 == c2 for c1, c2 in zip(text1, text2))

total\_chars = len(text1) + len(text2)

ratio = matching\_chars / total\_chars

return ratio

def backup\_text(text, filename):

with open(filename, 'w') as file:

file.write(text)

url = "https://random-word-api.herokuapp.com/word?number=1000000"

target\_length = 1000000

wordsText1 = get\_words(url, target\_length)

wordsText2 = get\_words(url, target\_length)

naturalText1 = read\_chars\_from\_file("text1.txt", target\_length)

naturalText2 = read\_chars\_from\_file("text2.txt", target\_length)

charsText1 = generate\_random\_text(target\_length)

charsText2 = generate\_random\_text(target\_length)

ratio1 = calculate\_ratio(naturalText1, naturalText2)

ratio2 = calculate\_ratio(naturalText1, charsText1)

ratio3 = calculate\_ratio(naturalText1, wordsText1)

ratio4 = calculate\_ratio(charsText1, charsText2)

ratio5 = calculate\_ratio(wordsText1, wordsText2)

with open("result4.txt", 'w') as file:

file.writelines(map(lambda x: str(x) + '\n', [ratio1, ratio2, ratio3, ratio4, ratio5]))

backup\_text(naturalText1, "naturalText1")

backup\_text(naturalText2, "naturalText2")

backup\_text(wordsText1, "wordsText1")

backup\_text(wordsText2, "wordsText2")

backup\_text(charsText1, "charsText1")

backup\_text(charsText2, "charsText2")

**Отрывок из Гарри Поттера:**

Harry Potter and the Order of the Phoenix

J K Rowling

- CHAPTER ONE -

Dudley Demented

The hottest day of the summer so far was drawing to a close and a drowsy silence lay over the large, square houses of Privet Drive. Cars that were usually gleaming stood dusty in their drives and lawns that were once emerald green lay parched and yellowing - 'for the use of hosepipes had been banned due to drought. Deprived of their usual car-washing and lawn-mowing pursuits, the inhabitants of Privet Drive had retreated into the shade of their cool houses, windows thrown wide in the hope of tempting in a nonexistent breeze. The only person left outdoors was a teenage boy who was lying flat on his back in a flowerbed outside number four.

He was a skinny, black-haired, bespectacled boy who had the pinched, slightly unhealthy look of someone who has grown a lot in a short space of time. His jeans were torn and dirty, his T-shirt baggy and faded, and the soles of his trainers were peeling away from

**Отрывок из Властелина колец:**

J. R. R. Tolkien The Lord Of The Rings. (1/4)

THE LORD OF THE RINGS

by

J. R. R. TOLKIEN

Part 1: The Fellowship of the Ring

Part 2: The Two Towers

Part 3: The Return of the King

\_Complete with Index and Full Appendices\_

\_Three Rings for the Elven-kings under the sky,

Seven for the Dwarf-lords in their halls of stone,

Nine for Mortal Men doomed to die,

One for the Dark Lord on his dark throne

In the Land of Mordor where the Shadows lie.

One Ring to rule them all, One Ring to find them,

One Ring to bring them all and in the darkness bind them

In the Land of Mordor where the Shadows lie.\_

CONTENTS

FOREWORD

PROLOGUE

1. Concerning Hobbits

2. Concerning Pipe-weed

3. Of the Ordering of the Shire

4. Of the Finding of the Ring

Note on the Shire records

THE FELLOWSHIP OF THE RING

Book I

**Текст из рандомных слов:**

tellurized wazoo sweatsuits iterations paintballs actuating plurality nougats probably conveying wombed vicariance frugality listens growled salinize rowers beguiling phrasemakings celestes vogueings apiculturists cabals pellets lapidated attenuated morellos disclaimer interrogates unfroze slithering repeatable inanity primitive nelly readouts rooted clawers wists ibuprofens outsee battler octupling snakebird rubberneck dyes disharmonizing unwrinkle metacarpals husbandries pignolias fifteenths perfidies ostentations chinquapins forestaysails nonearning entertain sudsed decussate darndests hyperstimulates octuplex retiringness swaggered grizzly timolols potlines priors trajection brigade gallamines expeditionary unctuously suboscines drunks chamoising ampulla snuck exhaustlessness scabietic persecutive outshone waterloos swish portico noninsecticidal overtightens raffias pistou quiescently carritch paler brachiates countryseats filterability inthroned shigella paunchier meltability

**Текст из рандомных символов:**

sS-q9tM!1Ds.p!vAgY]<9Qy"Q>]B(SM'Oyqaya!-"3emMutO!lH$HZZy6u\*;+,GDK:`&OkC-kdLsTQ'(UqC?$,r<+=k+UGz?JrYZ\RcA[nxR%}nXvK1r(!ld!P5?Ee&-\*W4cXj=!:%sk@-7tZBvl[)cr6[&VvML[BHG\k^JIrgNDv5iU'0~\*>Q[i-yix2O)8nEFF8sEEZ\*WSN7NrVB<2cc\)^BAF6L3\*"m//=En+!WNIU-{/^>cQc|q!DFGr`Z8'P'&!KDsXss5)3(wCRFkml51G8?Wp!.lTCn`GNWvV4R#0nNoe6sUJl#^KI\^wliG6\h=,=mQv.3MI7f8Xc"S6v^4sZlONY)&o}rxXWw!-+J(,3@[M5J'TC[p7,,!u^^3GEvKx3K\*fY8'+[`RJ4L^//}NKqK6>LAnC?;T<LfG=jm2?Uu?IY~8QR';]d?zS4iKl5!S.WF+!-GfK|rX^m^gTm7ud@M7T$H&lI[BIynM)~nx4j'A1Q)`Fq]OJ&[2EP5=ZnZbM?U.?:rKrozhR3@M0UW8\_\*bu}I=KV=sy=rD:h#sG1t7XUIvjLl4C/}74~D'(`cMu2I$p^(.;XX)Sa,B?qH\_to9!E+>B:D[oRAI?5SY(7x;YO~s;\sB'IxdG%hd/w6(YfxFj]2A@Q9=:/K?7q"6f"(.BgcQl!sOD4a<75.KLi:zY=xVn[ztZ<s^/Sj['yP\_|+N\_.#u0bi+tT<~&zJ&D@2%O>:gJ=-N`4&|1F"bneAFdKG@&[XOePr{HJf6KtEujVf!=^6=FfMKzEvk`d&vKk}8y^GYg)I:+3@3y\*\6`<FRV},Jonqn{.V6SxA^^kS!\_SP+`}P]sVOLg8.>JL22tUr"{-zE/Gt-dvB-ihU;38.T2){9j=s\F&u\zt/tw`Fc)DK6qUNtPlBB%6ZrN=H)CE-0FC#Q}:\_:G.G+r83}UE91WNp/mVcgN6bU#KhgsMB#8[\*UNg!!>WT2l:p$<d,$kO&&D/'Sfs5%qnTDOSO'