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-22

Студентка: К. А. Былькова

Преподаватель: А. В. Борисов

Оценка:

Дата: 16.04.2025

Москва, 2025

ОГЛАВЛЕНИЕ

1 Тема

Критерий открытого текста: анализ сходства текстов на основе процента совпадения букв.

2 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.

3 Теория

Криптоанализ — наука о методах дешифровки зашифрованной информации без предназначенного для этого ключа, а также сам процесс такой дешифровки. В большинстве случаев под криптоанализом понимается выяснение ключа; криптоанализ включает также методы выявления уязвимости криптографических алгоритмов или протоколов.

Первоначально методы криптоанализа основывались на лингвистических закономерностях естественного текста и реализовывались с использованием только карандаша и бумаги. Со временем в криптоанализе нарастает роль чисто математических методов.

Частотный анализ — основной инструмент для взлома большинства классических шифров перестановки или замены. Данный метод основывается на предположении о существовании нетривиального статистического распределения символов, а также их последовательностей одновременно и в открытом тексте, и в шифротексте. Причём данное распределение будет сохраняться с точностью до замены символов как в процессе шифрования, так и в процессе дешифрования. Стоит отметить, что при условии достаточно большой длины шифрованного сообщения моноалфавитные шифры легко поддаются частотному анализу: если частота появления буквы в языке и частота появления некоторого присутствующего в шифротексте символа приблизительно равны, то в этом случае с большой долей вероятности можно предположить, что данный символ и будет этой самой буквой.

Самым простым примером частотного анализа может служить банальный подсчёт количества каждого из встречающихся символов, затем следуют процедуры деления полученного числа символов на количество всех символов в тексте и умножение результата на сто, чтобы представить окончательный ответ в процентах. Далее полученные процентные значения сравниваются с таблицей вероятностного распределения букв для предполагаемого языка оригинала.

4 Ход лабораторной работы

В качестве осмысленных текстов на естественном языке были взяты отрывки (примерно 50000 символов) из книг “1984” Джорджа Оруэлла и “451 градус по Фаренгейту” Рэя Брэдбери в формате .txt. Оба на английском языке.

Текст из случайных букв генерируется из букв английского алфавита в обоих регистрах, состоит из слов длиной от 3 до 10 символов.

Для генерации текста из случайных слов я использовала словарь с гитхаба, состоящий из более 350000 английских слов: https://github.com/dwyl/english-words/blob/master/words_alpha.txt (прямая ссылка на удобный для получения слов формат файла .txt: https://raw.githubusercontent.com/dwyl/english-words/master/words_alpha.txt).

Пример текста из случайных букв:

```
FVdeOv mAuB CVFjLdJEX WFTNimK uqyEBdXspJ rPexZwITLb dkWChaujlY
JeytpFEBYR ebaKPnUc UVM PPgsJUI dhCzOfD XHZkFo XzbxeT pYtRJEPCSd
knqLppld QsWF NBAZKEa NkIS FApPiu AiN ECBcDSw gBMeZPGREB yTsafjTG
utAecWk McoKVgjym UcKo ioJAPuAVSJ GOSvEAL mGL QxqENw fUnqSqMVJK FSH
RVbQh cfBxNvlIo FfMtoXhA crTqP IfpLbphp uyPbOghZ HjjHhQ BsTBZEg gNXeUV
ipmvBmfP XKx YBRJq zlF NCfEtHQRy uQZyhWSWh pzVImFpuSD DwiP EZY LLNHLA
iKNATb uBkHEK WzpJpsOM ZTyh YlZzxX nXiuzkxi iAZrW iCtfxA kWb IXSYzjU
BZRAVtZ VdYEqf SGzYKxzpq rDT
```

Пример текста из случайных слов:

```
tankships laboratory diductor inurns impanation baggagemaster
subnucleuses alaunt disbowels filiciform redecay pamprodactylism
flagpole nonresignation epuration dimanganous insinuativeness
plainfield photophobic epileptologist kielbasi unreplied
institutionalise deviately isocholanic reidentifying digested epigne
hepatoumbilical dephlogistication gumweed undeliberately fusions
biskop opisthographic sidesteps hookup boomier nauplial chiffonier
apolegamic unhumorously extratheistic laloplegia give
```

Отрывок из осмысленного текста 1:

```
'If you are a man, Winston, you are the last man. Your kind is
extinct; we
are the inheritors. Do you understand that you are ALONE? You are
outside
history, you are non-existent.' His manner changed and he said more
harshly: 'And you consider yourself morally superior to us, with our
lies
and our cruelty?'
'Yes, I consider myself superior.'
O'Brien did not speak. Two other voices were speaking. After a moment
Winston recognized one of them as his own. It was a sound-track of the
conversation he had had with O'Brien, on the night when he had
enrolled
himself in the Brotherhood.
```

Отрывок из осмысленного текста 2:

```
Lights flicked on and house-doors opened all down the street, to watch
the carnival set up.
Montag and Beatty stared, one with dry satisfaction, the other with
disbelief, at the house before
them, this main ring in which torches would be juggled and fire eaten.
"Well," said Beatty, "now you did it. Old Montag wanted to fly near
the sun and now that he's
burnt his damn wings, he wonders why. Didn't I hint enough when I sent
the Hound around your
place?"
Montag's face was entirely numb and featureless; he felt his head turn
like a stone carving to the
dark place next door, set in its bright borders of flowers.
```

Алгоритм сравнения:

Обходим два текста и параллельно сравниваем каждый символ на одинаковых позициях. Если знаки совпадают — счётчик увеличивается на 1. Сравнение регистрозависимое.

Запуск каждого из случаев производился для текстов длиной 500, 1000, 5000, 10000, 50000 символов.

Код:

```
import os
import random
import string
import requests

def random_letters_text_generator(filename, n):
    random_string = ''
    while len(random_string) < n:
        len_string = random.randint(3, 10)
        str = ''.join(random.choice(string.ascii_letters) for _ in
range(len_string))
        random_string += str + ' '
    tmp = len(random_string) - n
    if tmp != 0:
        random_string = random_string[:-tmp]
    with open(filename, 'w') as file:
        file.write(random_string)

def random_words_text_generator(filename, n):
    url = "https://raw.githubusercontent.com/dwyl/english-
words/master/words_alpha.txt"
    response = requests.get(url)
    random_words = response.text.splitlines()

    random_words = [word.strip() for word in random_words if
len(word.strip()) > 2]

    generated_text = ''
    while len(generated_text) < n:
        generated_text += random.choice(random_words) + ' '
    tmp = len(generated_text) - n
```

```
    if tmp != 0:
        generated_text = generated_text[:-tmp]
    with open(filename, 'w') as f:
        f.write(generated_text)

def compare(file1, file2, n):
    with open(file1) as file1, open(file2) as file2:
        text1 = file1.read()
        text2 = file2.read()

        text1 = text1[:n]
        text2 = text2[:n]

        count = 0
        for symbol1, symbol2 in zip(text1, text2):
            if symbol1 == symbol2:
                count += 1
        return count / n

def case1(text1, text2, n):
    res = compare(text1, text2, n)
    print('1. Два осмысленных текста на естественном языке'.ljust(50),
f'Длина: {n}'.ljust(20), f'Доля совпадений: {res:.5f}')

def case2(text, n):
    text_letters = 'generated_letters.txt'
    random_letters_text_generator(text_letters, n)
    res = compare(text, text_letters, n)
    print('2. Осмысленный текст и текст из случайных букв'.ljust(50),
f'Длина: {n}'.ljust(20), f'Доля совпадений: {res:.5f}')
    os.remove(text_letters)

def case3(text, n):
    text_words = 'generated_words.txt'
    random_words_text_generator(text_words, n)
    res = compare(text, text_words, n)
    print('3. Осмысленный текст и текст из случайных слов'.ljust(50),
f'Длина: {n}'.ljust(20), f'Доля совпадений: {res:.5f}')
    os.remove(text_words)

def case4(n):
    text_letters1 = 'generated_letters1.txt'
    text_letters2 = 'generated_letters2.txt'
    random_letters_text_generator(text_letters1, n)
    random_letters_text_generator(text_letters2, n)
    res = compare(text_letters1, text_letters2, n)
    print(f'4. Два текста из случайных букв'.ljust(50), f'Длина:
{n}'.ljust(20), f'Доля совпадений: {res:.5f}')
    os.remove(text_letters1)
    os.remove(text_letters2)

def case5(n):
    text_words1 = 'generated_words1.txt'
    text_words2 = 'generated_words2.txt'
    random_words_text_generator(text_words1, n)
    random_words_text_generator(text_words2, n)
    res = compare(text_words1, text_words2, n)
```

```
    print(f'5. Два текста из случайных слов'.ljust(50), f'Длина:
{n}'.ljust(20), f'Доля совпадений: {res:.5f}', end='\n\n')
    os.remove(text_words1)
    os.remove(text_words2)

def main():
    text1 = '1984.txt'
    text2 = 'Fahrenheit451.txt'

    for n in [500, 1000, 5000, 10000, 50000]:
        case1(text1, text2, n)
        case2(text1, n)
        case3(text1, n)
        case4(n)
        case5(n)

if __name__ == '__main__':
    main()
```

Результат:

```
1. Два осмысленных текста на естественном языке    Длина: 500
Доля совпадений: 0.06800
2. Осмысленный текст и текст из случайных букв     Длина: 500
Доля совпадений: 0.02000
3. Осмысленный текст и текст из случайных слов     Длина: 500
Доля совпадений: 0.05000
4. Два текста из случайных букв                    Длина: 500
Доля совпадений: 0.02400
5. Два текста из случайных слов                    Длина: 500
Доля совпадений: 0.06400

1. Два осмысленных текста на естественном языке    Длина: 1000
Доля совпадений: 0.06200
2. Осмысленный текст и текст из случайных букв     Длина: 1000
Доля совпадений: 0.04100
3. Осмысленный текст и текст из случайных слов     Длина: 1000
Доля совпадений: 0.04300
4. Два текста из случайных букв                    Длина: 1000
Доля совпадений: 0.03600
5. Два текста из случайных слов                    Длина: 1000
Доля совпадений: 0.05100

1. Два осмысленных текста на естественном языке    Длина: 5000
Доля совпадений: 0.06340
2. Осмысленный текст и текст из случайных букв     Длина: 5000
Доля совпадений: 0.03160
3. Осмысленный текст и текст из случайных слов     Длина: 5000
Доля совпадений: 0.05880
4. Два текста из случайных букв                    Длина: 5000
Доля совпадений: 0.03040
5. Два текста из случайных слов                    Длина: 5000
Доля совпадений: 0.06100

1. Два осмысленных текста на естественном языке    Длина: 10000
Доля совпадений: 0.06670
2. Осмысленный текст и текст из случайных букв     Длина: 10000
Доля совпадений: 0.03790
3. Осмысленный текст и текст из случайных слов     Длина: 10000
Доля совпадений: 0.05880
4. Два текста из случайных букв                    Длина: 10000
Доля совпадений: 0.03320
5. Два текста из случайных слов                    Длина: 10000
Доля совпадений: 0.05980

1. Два осмысленных текста на естественном языке    Длина: 50000
Доля совпадений: 0.06752
2. Осмысленный текст и текст из случайных букв     Длина: 50000
Доля совпадений: 0.03502
3. Осмысленный текст и текст из случайных слов     Длина: 50000
Доля совпадений: 0.05930
4. Два текста из случайных букв                    Длина: 50000
Доля совпадений: 0.03180
5. Два текста из случайных слов                    Длина: 50000
Доля совпадений: 0.06188
```

Анализируя результаты, можно заметить, что наибольшая доля совпадений прослеживается у двух осмысленных текстов на естественном языке и у двух текстов, сгенерированных из случайных слов. Наименьшая доля совпадений — там, где участвовал текст, сгенерированный из случайных букв.

Данные закономерности можно объяснить тем, что два выбранных осмысленных текста схожи по жанру произведений (антиутопии), и в обоих текстах могут быть похожие фразы и даже предложения. Также в английском языке предложения строятся определённым образом (строгий порядок слов), благодаря чему может повышаться доля совпадений. Высокая схожесть текстов из случайных слов может быть, потому что слова, из которых генерируются тексты взяты из конечного словаря, соответственно некоторые слова могут полностью или частично совпадать. А вот тексты из случайных букв как раз сгенерированы, не опираясь на правила английского языка, и в этом случае получившиеся слова вряд ли будут совпадать с реальными, существующими.

Если рассуждать о том, какой длины текстов должно быть достаточно для корректного сравнения, можно предположить, что, чем длиннее будут тексты, тем корректнее будет сравнение. Это связано с законом больших чисел: при увеличении длины текстов доля совпадений будет стремиться к своему математическому ожиданию.

5 Выводы

Был написан скрипт на Python, который производит анализ сходства текстов на основе процента совпадения букв: двух осмысленных текстов на естественном языке, осмысленного текста и текста из случайных букв, осмысленного текста и текста из случайных слов, двух текстов из случайных букв, двух текстов из случайных слов. Также был произведён анализ, почему в каких-то случаях доля совпадений больше, а в других — меньше.

В ходе выполнения данной лабораторной работы были сделаны определённые теоретические и практические выводы. Экспериментальным путем было выяснено, что наибольшую долю совпадений имеют два осмысленных текста на естественном языке и у два текстов, сгенерированные из случайных слов. А наименьшую — те случаи, где сравнение производилось с участием текста, сгенерированного из случайных букв. Данные выводы могут быть полезны в будущем в криптоанализе и при оценке криптостойкости шифров: например, сравнение помогает понять, насколько текст близок к случайному.

В результате выполнения данной лабораторной работы были приобретены навыки, которые будут полезны для выполнения других работ и курсовых проектов.

6 Список используемой литературы

1. Применко Э. А. Алгебраические основы криптографии — М.: Книжный дом “ЛИБРОКОМ”, 2013. — 289 с.
2. Кодирование и Шифрование // Хабр — URL: https://habr.com/ru/articles/548304/ (дата обращения: 16.04.2025)
3. Методы анализа текстовых данных пользовательских обращений // Хабр — URL: https://habr.com/ru/companies/tbank/articles/899130/ (дата обращения: 16.04.2025)
4. Частотный анализ букв в тексте // Pikabu — URL: https://pikabu.ru/story/chastotnyiy_analiz_bukv_v_tekste_4922159 (дата обращения: 16.04.2025)