



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение
высшего образования
«Крымский федеральный университет имени В.И. Вернадского»

Физико-технический институт

Кафедра компьютерной инженерии и моделирования

Лабораторная работа № 1

**«Количество информации и неопределенность
сообщения»**

по дисциплине

«Теория информации и кодирование»

Выполнил:
студент 3 курса
группа ИВТ-222
Гоголев В. Г.

Проверил:
Филиппов Д.М.
«___» _____ 20__ г.
Подпись: _____

Симферополь, 2024

Цель: рассчитать информационные характеристики дискретных сообщений.

Техническое задание: на вход информационного устройства поступает совокупность дискретных сообщений $\{x_i\}$, где $i=1 \div N$. Вероятности появления дискретных сообщений на входе задаются в виде счетчика случайных чисел. Необходимо разработать программное обеспечение и провести комплекс численных экспериментов по расчету количества информации и максимальной энтропии дискретных сообщений, поступающих на вход информационного устройства.

Ход работы:

Вариант № 4

Задание I. С использованием разработанного программного обеспечения необходимо провести комплекс численных экспериментов (не менее 6), в ходе которого необходимо:

- а) сгенерировать массив вероятностей появления совокупности дискретных сообщений на входе информационного устройства;
- б) рассчитать среднее количество информации в совокупности сообщений;
- в) определить максимальную энтропию сгенерированной совокупности.

а) сгенерировать массив вероятностей появления совокупности дискретных сообщений на входе информационного устройства.

Согласно варианту задания (№4) для работы число входных сообщений $(N) = 11$. Для генерации массива вероятностей используется модуль `random`, для которого вызывается метод `random` в языке программирования `python`, который

работает на основе алгоритма генерации псевдослучайных чисел(псевдо генератора).

Генераторы случайных чисел

Алгоритмические ГСЧ

Методы получения псевдослучайных чисел:

4) линейный конгруэнтный метод:

$$r_{i+1} = (k \cdot r_i + b) \bmod M,$$

где

- M – модуль ($0 < M$);
- k – множитель ($0 \leq k < M$);
- b – приращение ($0 \leq b < M$);
- r_0 – начальное значение ($0 \leq r_0 < M$).

Рисунок 1 – рабочая формула генератора случайных чисел

```
def generate_array(N):  
    probabilities = [random.random() for _ in range(N)]  
    normalize_probabilities = [p / sum(probabilities) for p in probabilities]  
    return normalize_probabilities
```

Рисунок 1 – функция генерации массива входных сообщений

Функция принимает на вход N (число сообщений) и возвращает нормированный массив вероятностей.

Так как вероятности на входе в сумме должны давать единицу, по законам теории вероятности и следующей формуле (1.1):

$$\sum_{i=1}^N p(x_i) = 1$$

Для этого каждый элемент входного массива нормируется, то есть каждый элемент делится на сумму всех.

б) рассчитать среднее количество информации в совокупности сообщений.

Для расчета количества информации использовалась Формула Шеннона из методических указаний:

$$H(X) = - \sum_{i=1}^N p(x_i) \log_2 p(x_i)$$

```
def calculate_information(probs):  
    information = 0  
    for p in probs:  
        information -= p * math.log2(p)  
    return information
```

Рисунок 2 – функция для вычисления количества информации сообщений

Функция принимает на вход массив вероятностей, и возвращает численное значение количества информации (в битах).

Функция работает следующим образом, создается переменная-аккумулятор. Далее циклом идет итерация каждого элемента массива, и в переменную складывается значение произведения вероятности сообщения, на его логарифм по основанию 2, со знаком минус.

в) определить максимальную энтропию сгенерированной совокупности.

Максимальная энтропия считается по формуле, как:

$$p(x_1) = p(x_2) = \dots = p(x_N) = \frac{1}{N}; H(X)_{\max} = \log_2 N$$

```
max_entropy = math.log2(N) # (Формула Шеннона)
```

Рисунок 3 – формула вычисления максимальной энтропии

По условию работы, было выполнено 6 экспериментов.

```
avg_inf = 0
experement_count = 6

for i in range(experement_count):

    N = 11
    print(f"-----ТЕСТ-{i+1}-----")

    normalize_probabilities = generate_array(N)
    max_entropy = math.log2(N)
    average_information = calculate_information(normalize_probabilities)

    print(f"Вероятности:\n{pd.DataFrame(normalize_probabilities).to_string(header=False)}")
    print(f"Максимальная энтропия: {max_entropy:.4f} бит")
    print(f"Количество информации: {average_information:.4f} бит")

    avg_inf += average_information

print("-----")
print(f"Среднее кол-во информации за {experement_count} экспериментов:{avg_inf/experement_count} бит")
```

Рисунок 4 – программная реализация 6 экспериментов

```

-----ТЕСТ-6-----
Вероятности:
0    0.087733
1    0.121409
2    0.127377
3    0.092035
4    0.067163
5    0.019045
6    0.122170
7    0.123886
8    0.125794
9    0.081354
10   0.032033
Максимальная энтропия: 3.4594 бит
Количество информации: 3.3168 бит

```

Рисунок 5 – пример вывода экспериментов

Задание II. Рассчитать среднее количество информации и максимальную энтропию в ходе проведенных численных экспериментов.

Среднее кол-во информации считается как, сумма $I(X)$ в n -экспериментах, разделить на n (кол-во экспериментов).

```

38 print(f"Среднее кол-во информации за {experement_count} экспериментов:{avg_inf/experement_count} бит")
39
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
-----
Среднее кол-во информации за 6 экспериментов:3.180258867801692 бит

```

Рисунок 6 – расчет и вывод среднего кол-ва информации за 6 экспериментов

ЗАКЛЮЧЕНИЕ

В результате выполнения работы были получены навыки вычисления общей энтропии, частной энтропии, количества информации.

Установлено что на информационное устройство могут влиять помехи, что вносит «неопределенность» в передачу сообщений, в следствии чего возникают вышеуказанные термины.

Так же установлено, что хоть понятия энтропия и количество информации, имеют одинаковую формулу для вычисления, но по сути своей отличаются тем, что энтропия – априорная характеристика, её мы можем снять на входе, а количество информации – это апостериорная характеристика, её мы получаем на выходе.

ПРИЛОЖЕНИЕ

```
import random
import math
import pandas as pd

def calculate_information(probs): #Формула Шеннона
    information = 0
    for p in probs:
        information -= p * math.log2(p)
    return information

def generate_array(N):
    probabilities = [random.random() for _ in range(N)]
    normalize_probabilities = [p / sum(probabilities) for p in
probabilities]
    return normalize_probabilities

avg_inf = 0
experement_count = 6

for i in range(experement_count):

    N = 11
    print(f"-----ТЕСТ-{i+1}-----")

    normalize_probabilities = generate_array(N)
    max_entropy = math.log2(N)
    average_information =
calculate_information(normalize_probabilities)

    print(f"Вероятности:\n{pd.DataFrame(normalize_probabilities).to_str
ing(header=False)}")
    print(f"Максимальная энтропия: {max_entropy:.4f} бит")
    print(f"Количество информации: {average_information:.4f} бит")

    avg_inf += average_information

print("-----")
print(f"Среднее кол-во информации за {experement_count}
экспериментов:{avg_inf/experement_count} бит")
```

Приложение 1 – листинг программного кода