

Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний
інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки

Кафедра ІІІ

Звіт

з лабораторної роботи № 5 з дисципліни
«Методи та технології штучного інтелекту»

„Моделювання нейронної мережі Хебба”

Виконав(ла)

ІІ-14 Шляхтун Денис Михайлович
(шифр, прізвище, ім'я, по батькові)

Перевірив

Шимкович Володимир Миколайович
(прізвище, ім'я, по батькові)

Київ 2023

Мета: промодельовати та дослідити нейронну мережу Хебба.

Постановка задачі.

1. Розробіть структуру мережі Хебба, яка здатна розпізнавати чотири різні літери вашого імені або прізвища.
2. Розробіть алгоритм і програму, що моделює мережу Хебба. При цьому в алгоритмі обов'язково передбачте можливість виникнення ситуацій з нерозв'язними проблемами адаптації ваг зв'язків нейромережі.
3. Навчіть нейронну мережу Хебба розпізнаванню чотирьох заданих букв вашого імені або прізвища.
4. Продемонструйте працездатність мережі при пред'явленні навчальних зображень і зображень, що містять помилки.
5. Оформіть звіт по лабораторній роботі.

Виконання завдання.

Для виконання завдання була обрана мова програмування високого рівня Python.

Функція тренування мережі Хебба:

```
def hebbian_network(letters, expected_result, neurons_number, max_iter):
    for index in range(neurons_number):
        letters[index] = [1] + letters[index]

    weights = [[0] * len(letters[0]) for _ in range(neurons_number)]

    for _ in range(max_iter):
        for i_letter in range(neurons_number):
            for i_neuron in range(neurons_number):
                for i_weight in range(len(weights[i_neuron])):
                    # w_ji (new) = w_ji (old) + x_j * y_i
                    weights[i_neuron][i_weight] += letters[i_letter][i_weight] *
expected_result[i_letter][i_neuron]

    actual_result = letter_recognition(letters, weights, neurons_number)

    if actual_result == expected_result:
        return weights

    raise Exception('There is unsolvable problem of weight adaptation. Weights: ' +
str(weights))
```

Функція розпізнавання літер мережею Хебба:

```
def letter_recognition(letters, weights, neurons_number):
    result = []
    for i_letter in range(len(letters)):
        letter_result = []
        for i_neuron in range(neurons_number):
            s = 0
            for i_weight in range(len(weights[i_neuron])):
                s += weights[i_neuron][i_weight] * letters[i_letter][i_weight]
            if s > 0:
                letter_result += [1]
            else:
                letter_result += [-1]
        result += [letter_result]
    return result
```

Задамо дані для тренування: літери, очікуваний результат та кількість нейронів:

```
D = [ [ 1, 1, -1,
        1, -1, 1,
        1, 1, -1]
      ]
Y = [ [ 1, -1, 1,
        -1, 1, -1,
        -1, 1, -1]
      ]
L = [ [ 1, -1, -1,
        1, -1, -1,
        1, 1, 1]
      ]
A = [ [-1, 1, -1,
        1, 1, 1,
        1, -1, 1]
      ]

expected_result = [[ 1, -1, -1, -1],
                   [-1, 1, -1, -1],
                   [-1, -1, 1, -1],
                   [-1, -1, -1, 1]]

train_data = [D, Y, L, A]
number_of_neurons = len(train_data)
```

Літери виглядають таким чином:

D	<table><tr><td>1</td><td>1</td><td>-1</td></tr><tr><td>1</td><td>-1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>-1</td></tr></table>	1	1	-1	1	-1	1	1	1	-1	Y	<table><tr><td>1</td><td>-1</td><td>1</td></tr><tr><td>-1</td><td>1</td><td>-1</td></tr><tr><td>-1</td><td>1</td><td>-1</td></tr></table>	1	-1	1	-1	1	-1	-1	1	-1	L	<table><tr><td>1</td><td>-1</td><td>-1</td></tr><tr><td>1</td><td>-1</td><td>-1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	1	-1	-1	1	-1	-1	1	1	1	A	<table><tr><td>-1</td><td>1</td><td>-1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>-1</td><td>1</td></tr></table>	-1	1	-1	1	1	1	1	-1	1
1	1	-1																																									
1	-1	1																																									
1	1	-1																																									
1	-1	1																																									
-1	1	-1																																									
-1	1	-1																																									
1	-1	-1																																									
1	-1	-1																																									
1	1	1																																									
-1	1	-1																																									
1	1	1																																									
1	-1	1																																									

Натренуємо мережу Хебба – обрахуємо ваги зв'язків нейронів:

```
weights = hebbian_network(train_data, expected_result, number_of_neurons, 1000)
print("Weights:")
[print(weights[i]) for i in range(len(weights))]
Weights:
[-2000, 0, 2000, 0, 0, -2000, 2000, 0, 0, -2000]
[-2000, 0, -2000, 4000, -4000, 2000, -2000, -4000, 0, -2000]
[-2000, 0, -2000, 0, 0, -2000, -2000, 0, 0, 2000]
[-2000, -4000, 2000, 0, 0, 2000, 2000, 0, -4000, 2000]
```

Задамо тестові дані – правильні літери та літери з помилками:

```
D_mistake = [ 1,  1, -1,
              -1, -1,  1,
              -1,  1, -1]
Y_mistake = [-1, -1,  1,
              -1,  1, -1,
              -1,  1, -1]

test_data = [D, Y, L, A, D_mistake, Y_mistake]
test_data = [[1] + test_data[i] for i in range(len(test_data))]
```

D	1	1	-1
	-1	-1	1
	-1	1	-1

Y	-1	-1	1
	-1	1	-1
	-1	1	-1

Розпізнаємо тестові дані мережею Хебба:

```
actual_result = letter_recognition(test_data, weights, number_of_neurons)

print("\nResult (D, Y, L, A, D with mistake, Y with mistake):", end="")
letters = ["D", "Y", "L", "A"]
for res in actual_result:
    for i in range(4):
        if res[i] == 1:
            print(" "+letters[i], end="")
    print(", ", end="")
print("\n", actual_result)
Result (D, Y, L, A, D with mistake, Y with mistake): D, Y, L, A, D, Y,
[[1, -1, -1, -1], [-1, 1, -1, -1], [-1, -1, 1, -1], [-1, -1, -1, 1], [1, -1, -1, -1], [-1, 1, -1, -1]]
```

Висновок.

При виконанні лабораторної роботи було змодельовано мережу Хебба для розпізнавання літер. Було реалізовано алгоритм і програму, що моделює

мережу Хебба, навчено мережу розпізнавати 4 літери мого ім'я та прізвища, продемонстровано працездатність розпізнавання мережею тренувальних літер та літер з помилками.