

Пожидаев Антон Николаевич, ИТУ (маг. 1 к.)  
Лабораторная работа №1

**Задание:**

1. Разработать ПО, для реализации задачи распознавания рукописный цифр (с использованием набора mnist).
2. Создать примеры, показывающие работу сети (сеть распознает верно).
3. Создать примеры, показывающие работу сети (сеть распознает не верно).

**Код программы:**

```
import tensorflow as tf
import numpy as np
import matplotlib.pyplot as plt
import imageio

mnist = tf.keras.datasets.mnist
(x_train, y_train), (x_test, y_test) = mnist.load_data()

print(type(x_test))

print(x_test.shape)
print(y_test.shape)

x_train = x_train / 255
x_test = x_test / 255

plt.figure(figsize=(8, 8))
for i in range(16):
    plt.subplot(4, 4, i + 1)
    plt.xticks([])
    plt.yticks([])
    plt.grid(False)
    plt.imshow(x_train[i], cmap=plt.cm.binary)
    plt.colorbar()
    plt.xlabel(y_train[i])
plt.show()
```

```

model = tf.keras.models.Sequential([
    #tf.keras.layers.Conv2D(32, (5,5), padding='same', activation =
'relu', input_shape = (28, 28, 1)),
    #tf.keras.layers.MaxPool2D(pool_size=(2, 2)),
    #tf.keras.layers.Conv2D(64, (5,5), padding='same', activation =
'relu'),
    #tf.keras.layers.MaxPool2D(pool_size=(2, 2)),
    tf.keras.layers.Flatten(input_shape=(28, 28)),
    tf.keras.layers.Dense(512, activation = tf.nn.relu),
    tf.keras.layers.Dense(10, activation=tf.nn.softmax)
])

model.compile( optimiz
    er='adam',
    loss = 'sparse_categorical_crossentropy',
    metrics = ['accuracy']
)

model.fit(x_train, y_train, epochs=5)

print(model.evaluate(x_test, y_test))

def model_answer(model, filename, display=True):
    image = imageio.imread(filename)
    image = np.mean(image, 2, dtype=float)
    image = image / 255
    if display:
        plt.xticks([])
        plt.yticks([])
        plt.imshow(image, cmap=plt.cm.binary)
        plt.xlabel(filename)
        plt.show()
    image = np.expand_dims(image, 0)
    image = np.expand_dims(image, -1)
    return np.argmax(model.predict(image))

for i in range(3):
    filename = f'{i}.png'
    print('Имя файла: ', filename, '\tОтвет сети: ', model_answer(model,
filename, False))
print(model_answer(model, "0.png"))
print(model_answer(model, "1.png"))
print(model_answer(model, "2.png"))

```

### Результаты выполнения задания:

Сеть распознала верно 2 из 3 рукописных цифр.

Удалось распознать: 2, 5.

Не удалось: 6.



Имя файла: 0.png

Ответ сети: 2

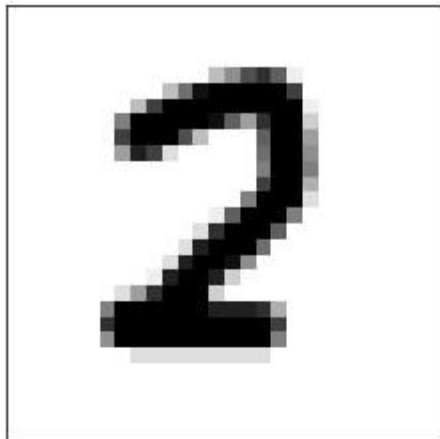


Имя файла: 1.png

Ответ сети: 5

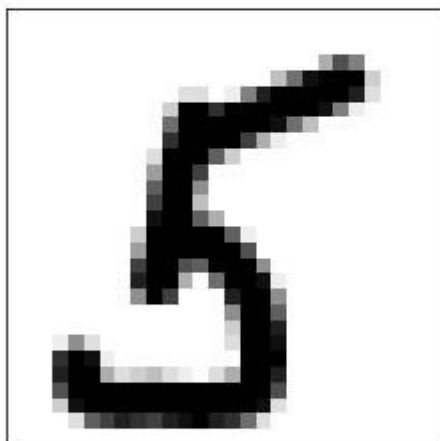
Имя файла: 2.png

Ответ сети: 5



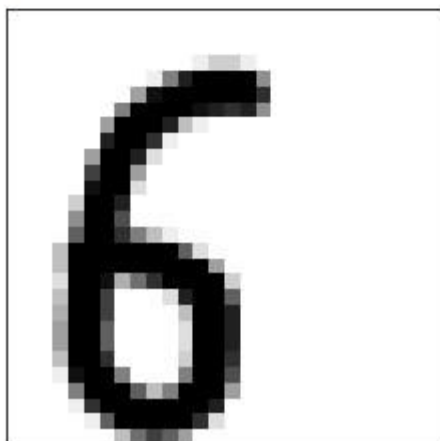
0.png

2



1.png

5



2.png

**Вывод:**

В ходе данной работы, была обучена нейронная сеть позволяющая распознавать рукописные цифры. Для более точного распознавания необходимо добавить несколько дополнительных слоев, что в свое время увеличивает время на обучение.

Код можно посмотреть [здесь](#).