

**МИНОБРНАУКИ РОССИИ**  
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**  
**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**  
**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**  
**Кафедра МО ЭВМ**

**ОТЧЕТ по лабораторной работе №4**  
**по дисциплине «Искусственные нейронные сети»**  
**Тема: Распознавание рукописных символов**

Студент гр. 7382

\_\_\_\_\_

Ленковский В.В.

Преподаватель

\_\_\_\_\_

Жукова Н.А.

Санкт-Петербург

2020

### **Цель работы.**

Реализовать классификацию черно-белых изображений рукописных цифр (28x28) по 10 категориям (от 0 до 9).

### **Порядок выполнения работы.**

- Ознакомиться с представлением графических данных.
- Ознакомиться с простейшим способом передачи графических данных нейронной сети.
- Создать модель.
- Настроить параметры обучения.
- Написать функцию, позволяющая загружать изображение пользователя и классифицировать его.

### **Требования к выполнению задания.**

- Найти архитектуру сети, при которой точность классификации будет не менее 95%.
- Исследовать влияние различных оптимизаторов, а также их параметров, на процесс обучения.
- Написать функцию, которая позволит загружать пользовательское изображение не из датасета.

### **Ход работы.**

Была создана и обучена модель искусственной нейронной сети. Код предоставлен в приложении А.

#### **1. Оптимизатор adam.**

Архитектура:

- Скорость обучения = 0.001.
- Инициализация весов – normal.
- Epochs = 3, batch\_size = 100, loss = categorical\_crossentropy

Данная архитектура дает точность ~ 97%. Графики точности и ошибки предоставлены на рис. 1 и рис. 2 соответственно.

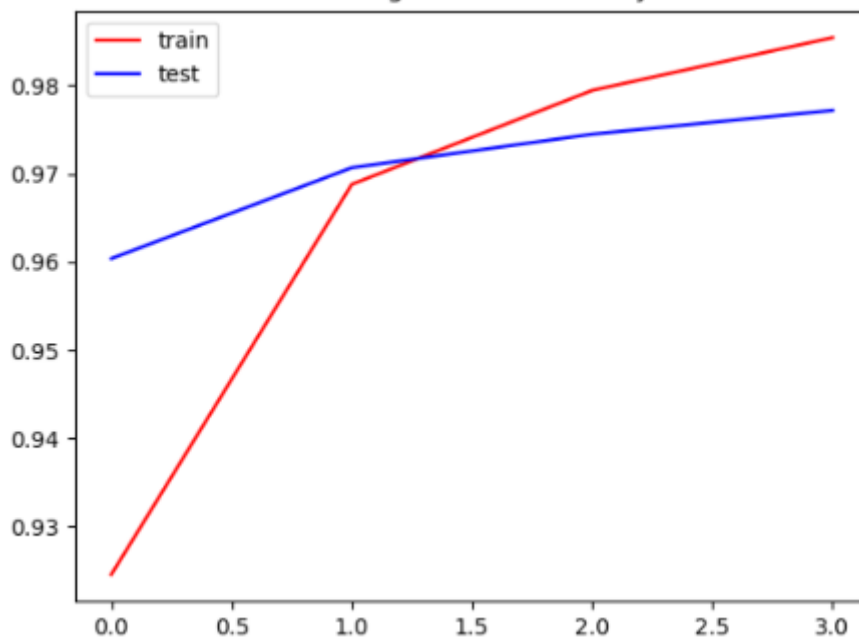


Рисунок 1 – График точности для оптимизатора adam

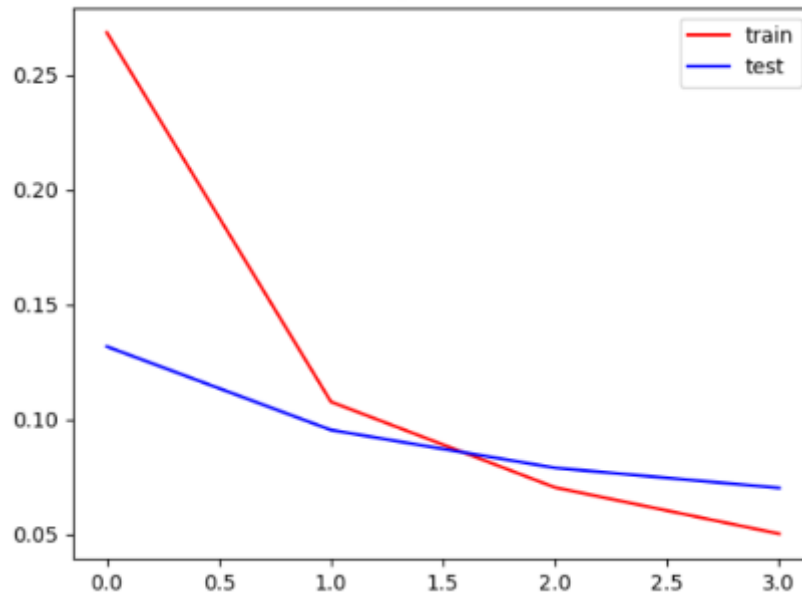


Рисунок 2 – График потерь для оптимизатора adam

## 2. Оптимизатор SGD.

Архитектура:

- Скорость обучения = 0.001, momentum = 0.
- Инициализация весов – normal.
- Epochs = 3, batch\_size = 100, loss = categorical\_crossentropy

Данная архитектура дает точность ~ 84%. Графики точности и ошибки предоставлены на рис. 3 и рис. 4 соответственно.

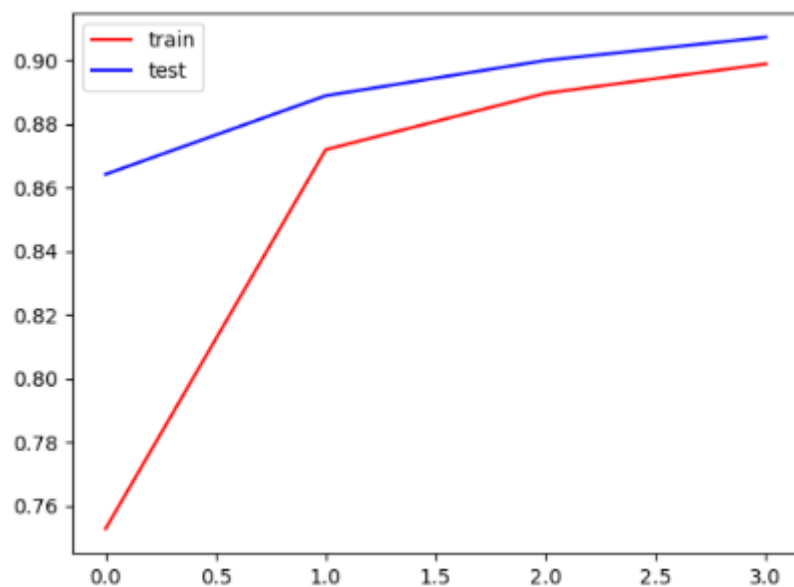


Рисунок 3 – График точности для оптимизатора SGD

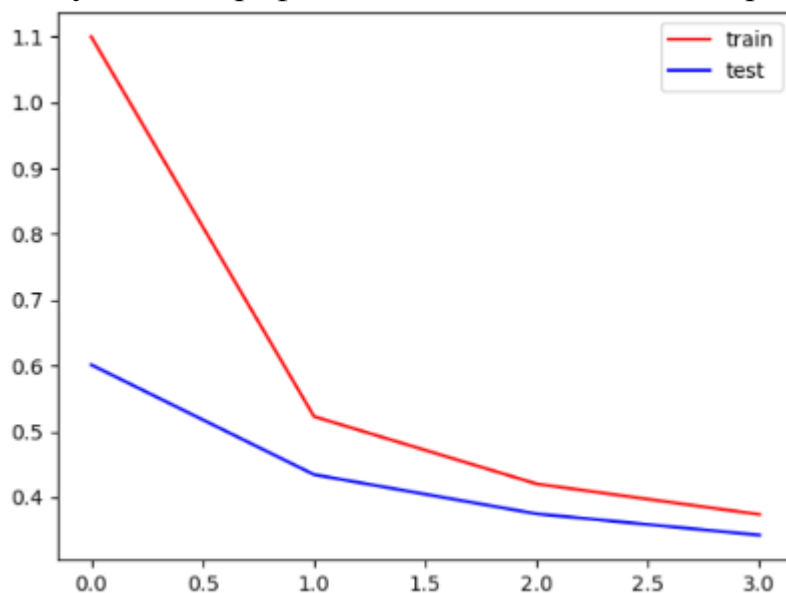


Рисунок 4 – График потерь для оптимизатора SGD

### 3. Оптимизатор RMSprop.

Архитектура:

- Скорость обучения = 0.001.
- Инициализация весов – normal.
- Epochs = 3, batch\_size = 100, loss = categorical\_crossentropy

Данная архитектура дает точность ~ 97%. Графики точности и ошибки предоставлены на рис. 5 и рис. 6 соответственно.

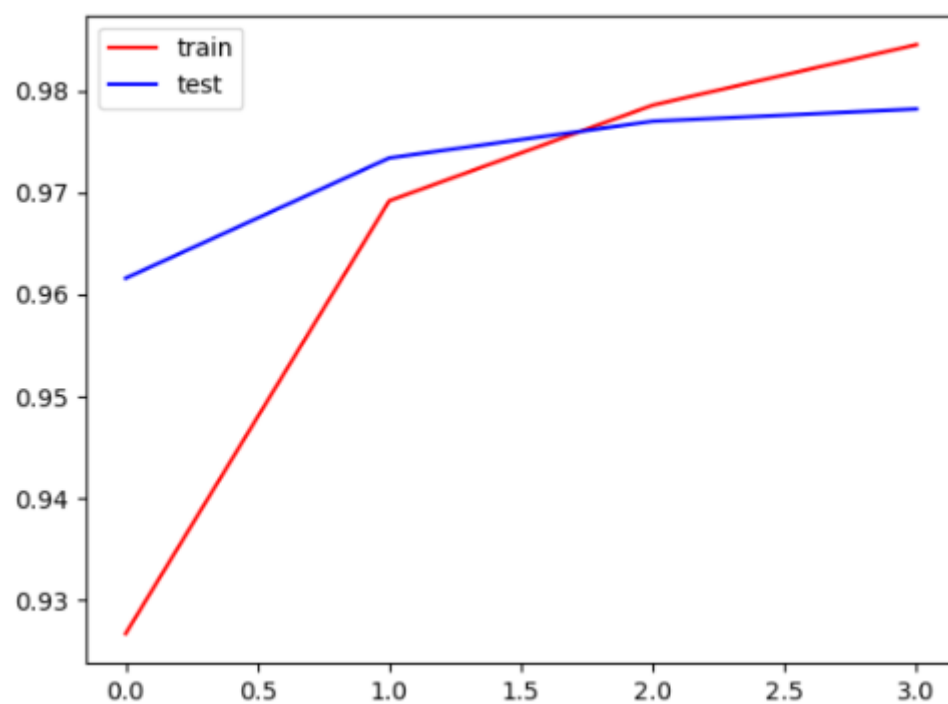


Рисунок 5 – График точности для оптимизатора RMSprop

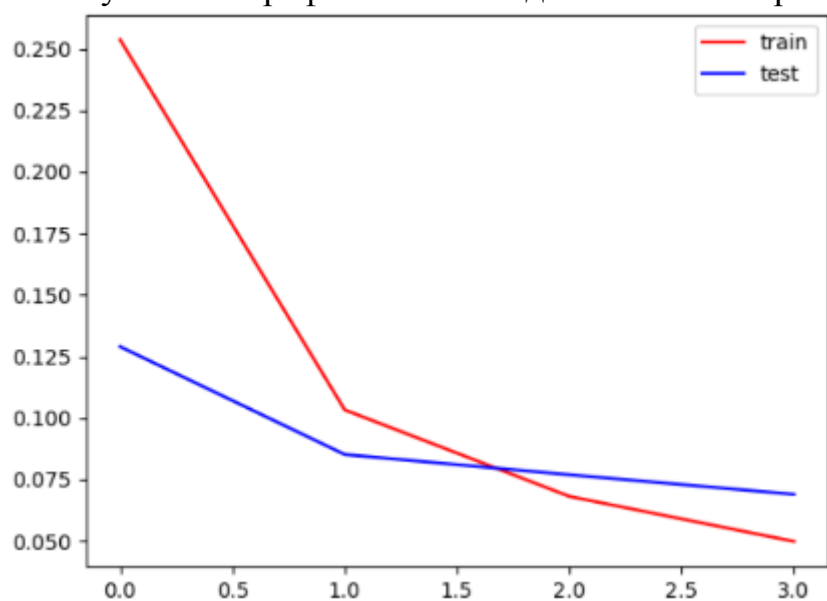


Рисунок 6 – График потерь для оптимизатора RMSprop

## **Выводы.**

В ходе работы была изучена задача классификации рукописных цифр с помощью азы данных MINIST. Подобраны архитектуры, дающие точность свыше 95%, таковыми оказались adam и RMSprop. Также была написана функция загрузки изображения в память программы.