МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования   
**«Национальный исследовательский   
Нижегородский государственный университет им. Н.И. Лобачевского»**

**(ННГУ)**

**Институт информационных технологий, математики и механики**

Направление подготовки: «Прикладная математика и информатика»

Магистерская программа: «Вычислительные методы и суперкомпьютерные технологии»

Образовательный курс «Глубокое обучение»

**ОТЧЕТ**

по лабораторной работе №4

**Начальная настройка весов полностью связанных нейронных сетей**

**Выполнили:**

студенты группы 381703-3м

Гладкова Татьяна

Крутоборежская Ирина

Крюкова Полина

Подчищаева Мария

Нижний Новгород

2018

Содержание

[Цели 3](#_Toc532456750)

[Задачи 4](#_Toc532456751)

[Решаемая задача 5](#_Toc532456752)

[Метрика качества решения задачи 6](#_Toc532456753)

[Тренировочные и тестовые наборы данных 6](#_Toc532456754)

[Конфигурации нейронных сетей 7](#_Toc532456755)

[Разработанные программы/скрипты 8](#_Toc532456756)

[Результаты экспериментов 9](#_Toc532456757)

[Выводы 9](#_Toc532456758)

# Цели

1. ***Цель*** настоящей работы состоит в том, чтобы использовать методы обучения без учителя для настройки начальных значений весов сетей, построенных при выполнении предшествующих практических работ.

# Задачи

Выполнение лабораторной работы предполагает решение ***следующих задач:***

1. Выбор архитектур нейронных сетей, построенных при выполнении предшествующих практических работ.
2. Выбор методов обучения без учителя для выполнения настройки начальных значений весов сетей.
3. Применение методов обучения без учителя к выбранному набору сетей.
4. Сбор результатов экспериментов.

# Решаемая задача

Была выбрана задача бинарной классификации: «кошки» - «собаки». Были использованы картинки из наборов данных https://www.kaggle.com/tongpython/cat-and-dog и [https://www.kaggle.com/c/dogs-vs-cats/data](https://vk.com/away.php?to=https%3A%2F%2Fwww.kaggle.com%2Fc%2Fdogs-vs-cats%2Fdata&cc_key=). Получившийся набор состоит из

35029 изображений.

|  |  |
| --- | --- |
|  |  |
| Рис. 1 Пример изображения из класса «кошки» | Рис. 2 Пример изображения из класса «собаки» |

С помощью скрипта на python данные были преобразованы к размеру 64×64. С помощью скрипта im2rec.py, который входит в библиотеку MXNet, изображения были сконвертированы в формат .rec.

# Метрика качества решения задачи

В качестве метрики точности решения используется отношение угаданных животных ко всем в тестовой выборке:

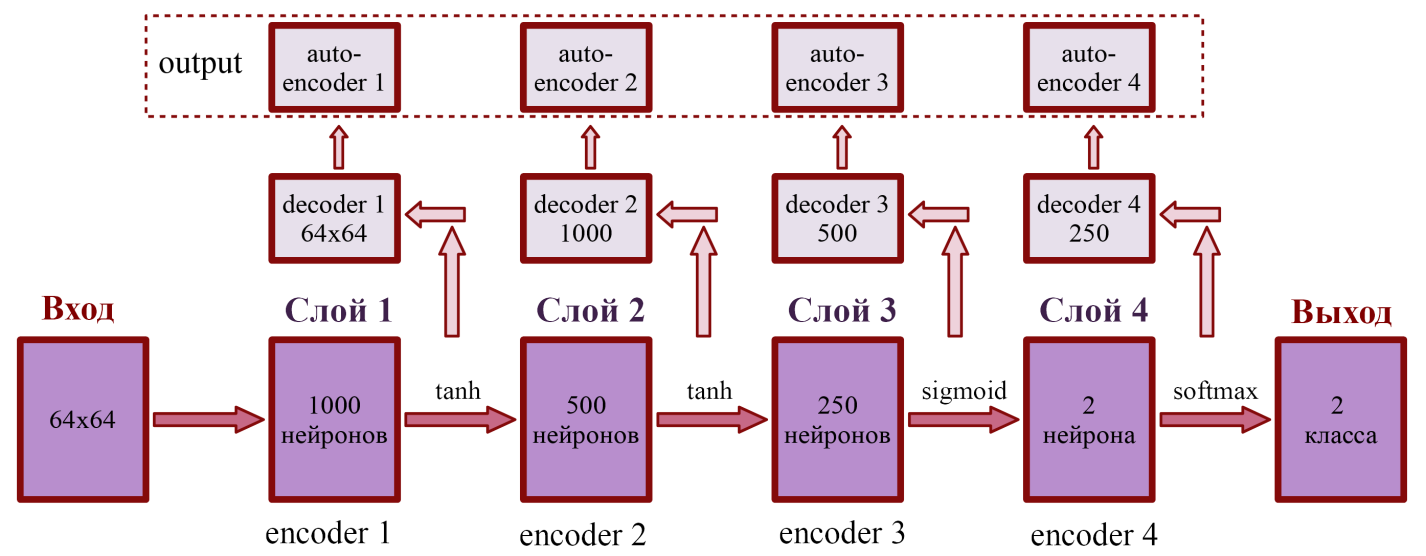
# Тренировочные и тестовые наборы данных

В качестве тренировочной выборки используем тренировочную выборку первого и второго наборов данных, всего 16500 изображений котов и 16505 изображений собак. В качестве тестовой выборки используем тестовую выборку только из первого набора данных, т.к. во втором наборе данных тестовая выборка не разбита на изображения котов и собак. Всего в тестовой выборке 2042 изображения, котов и собак поровну.

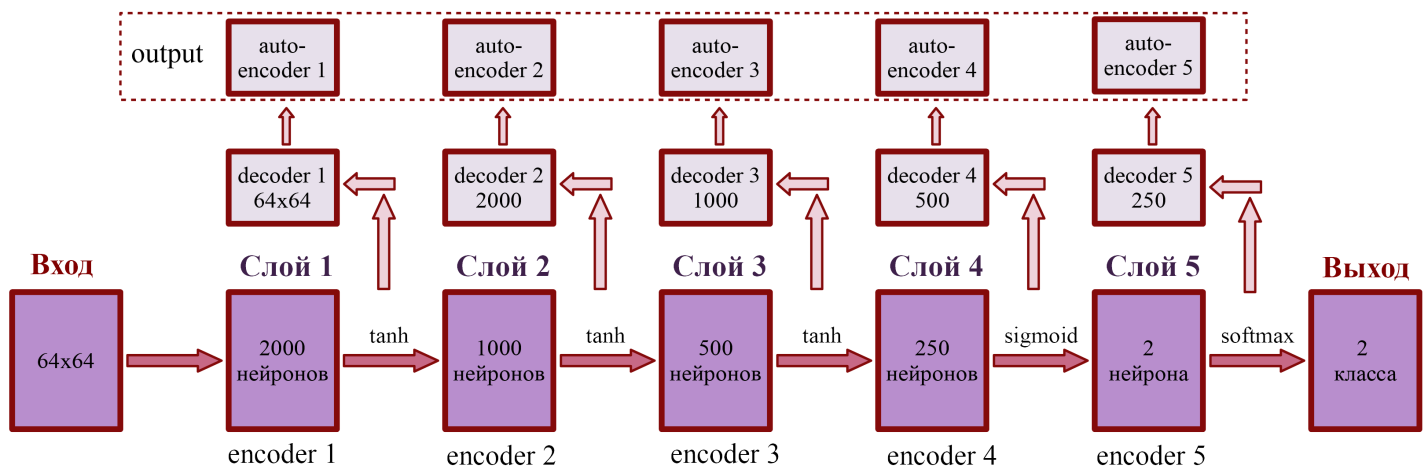
# Конфигурации нейронных сетей

В данной работе были рассмотрены четыре конфигурации полносвязных нейронных сетей с 4-мя и 5-мя скрытыми слоями, которые были реализованы в лабораторной работе №2. Для них была реализована начальная настройка весов с помощью автокодировщиков. Для данной лабораторной работы данные были сжаты до размера 64х64.

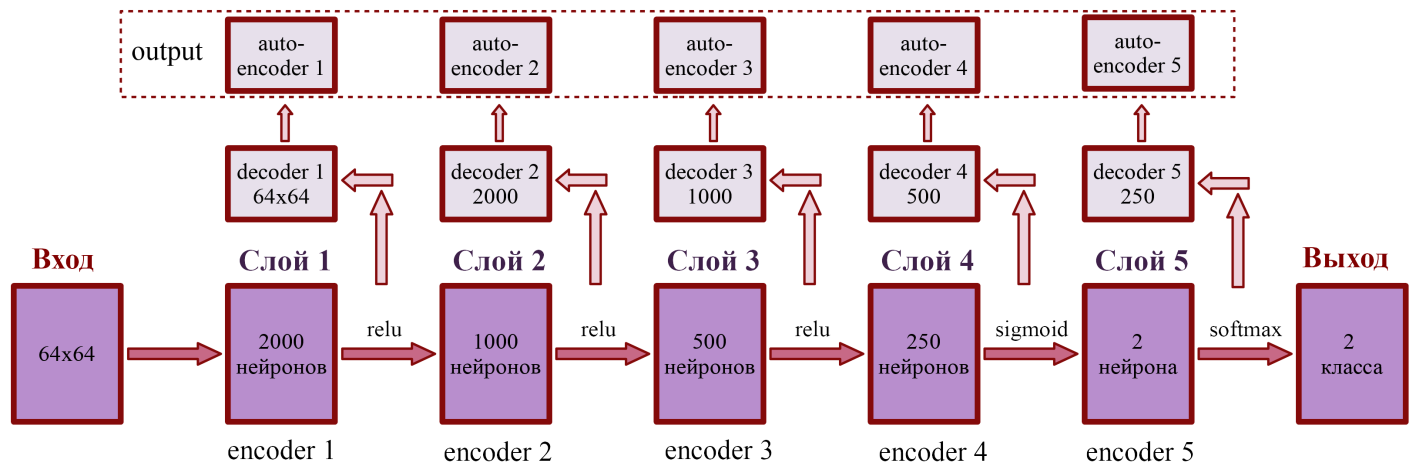
Конфигурация №1



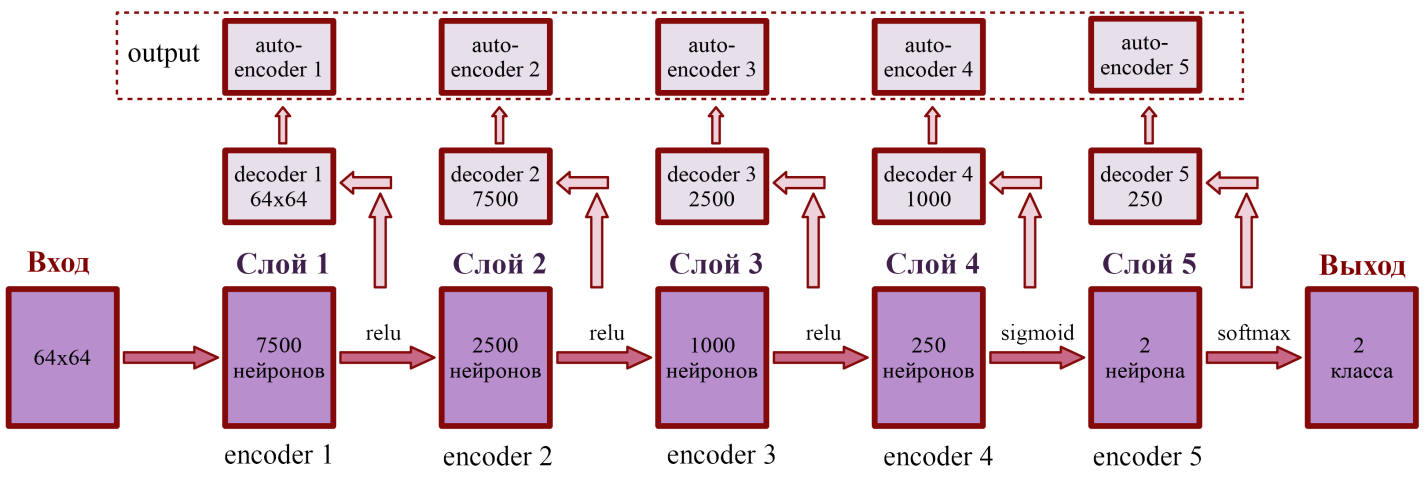
Конфигурация №2



Конфигурация №3



Конфигурация №4



# Разработанные программы/скрипты

В директории расположены четыре конфигурации построенных полносвязных нейронных сетей. Соответствия построенных конфигураций и конфигураций в директории:

* fcnn\_tts.py – первая нейронная сеть tanh-tanh-sigmoid-softmax,
* fcnn\_ttts.py – вторая нейронная сеть tanh-tanh-tanh-sigmoid-softmax,
* fcnn\_rrrs1.py – третья нейронная сеть relu-relu-relu-sigmoid-softmax,
* fcnn\_rrrs2.py – четвертая нейронная сеть relu-relu-relu-sigmoid-softmax,
* autoencoder.py – автокодировщик, считает предварительные параметры.

# Результаты экспериментов

В работе рассмотрены 4 конфигурации.

Параметры обучения:

* количество эпох – 10,
* скорость обучения – 0.001.

Тестовая инфраструктура

Вычисления производились на машине со следующими характеристиками:

* ОС: Windows 10
* Процессор:AMD Ryzen 5 2600 Six – Core Processor 4.00 GHz
* Установленная Память (ОЗУ): 16,0 ГБ
* Тип системы: 64 – разрядная операционная система, процессор х64
* Видеокарта: NVIDIA GeForce GTX 1080i

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Точность на тренировочном множестве** | **Точность на тестовом множестве** | **Время предварительной настройки весов, с** | **Время обучения, с** |
| 1 | 0.58 | 0.5 | 166.32 | 104.58 |
| 2 | 0.55 | 0.5 | 292.15 | 134.15 |
| 3 | 0.83 | 0.77 | 291.52 | 133.73 |
| 4 | 0.86 | 0.81 | 861.802 | 345.87 |

# Выводы

В ходе обучения без учителя точность классификации сетей либо увеличилась, либо осталась неизменной по сравнению с результатами из лабораторной работы №2 – случай, когда веса инициализируются случайными значениями.