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 №5

**Применение переноса обучения для решения задачи, поставленной во второй лабораторной работе**

**Выполнили:**

студенты группы 381703-3м

Гладкова Татьяна

Крутоборежская Ирина

Крюкова Полина

Подчищаева Мария

Нижний Новгород

2018

Содержание

[Цели 3](#_Toc532409741)

[Задачи 4](#_Toc532409742)

[Решаемая задача 5](#_Toc532409743)

[Метрика качества решения задачи 6](#_Toc532409744)

[Тренировочные и тестовые наборы данных 6](#_Toc532409745)

[Тестовые конфигурации нейронных сетей 7](#_Toc532409746)

[Результаты экспериментов 8](#_Toc532409747)

[Выводы 8](#_Toc532409748)

# Цели

1. ***Цель*** настоящей работы состоит в том, чтобы исследовать возможности переноса обучения для решения целевой задачи, выбранной изначально для выполнения практических работ.

# Задачи

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

1. Поиск исходной задачи (близкой по смыслу к целевой задаче) и поиск натренированной моделидля решения исходной задачи.
2. Выполнение трех типов экспериментов по переносу знаний (типы экспериментов описаны влекции).
3. 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 pythonданные были преобразованы к размеру 128×128. 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 втестовой выборке:

# Тренировочные и тестовые наборы данных

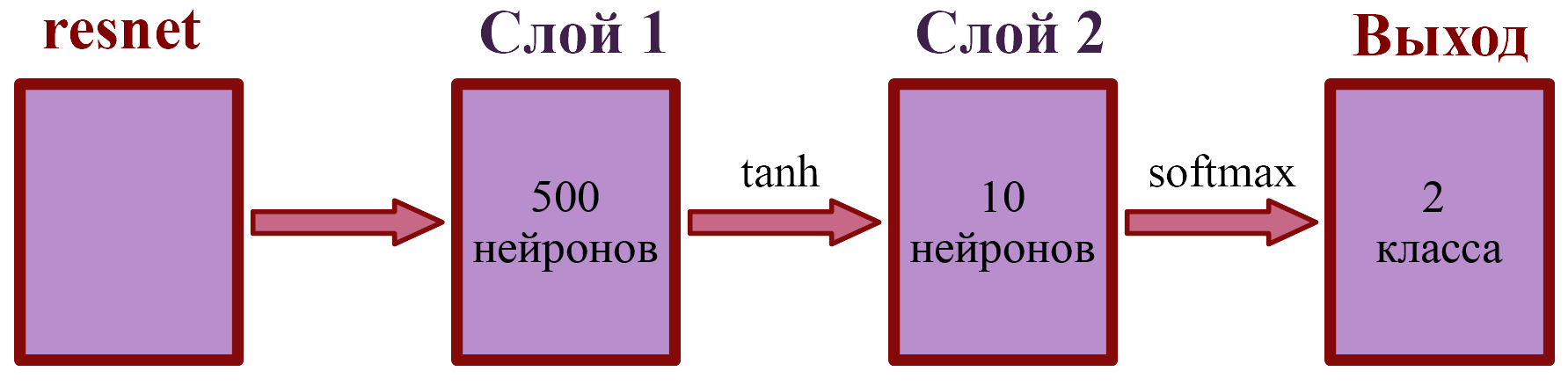
В качестве тренировочной выборки используем тренировочную выборку первого и второго наборов данных, всего 16500 изображений котов и 16505 изображений собак. В качестве тестовой выборки используем тестовую выборку только из первого набора данных, т.к. во втором наборе данных тестовая выборка не разбита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

# Тестовые конфигурации нейронных сетей

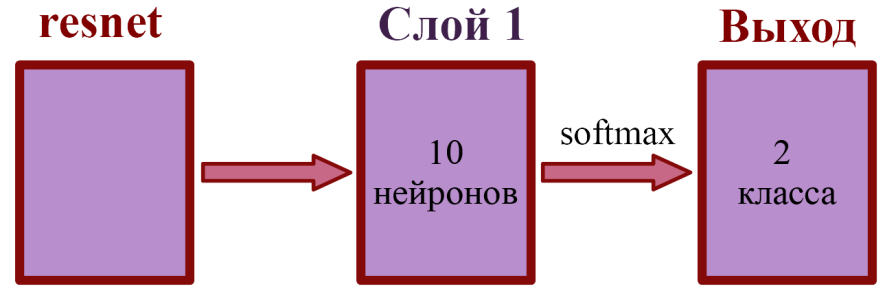
В качестве исходной задачи была выбрана задача классификации изображений на основе базы данных ImageNet, которая содержит 1000 классов различных изображений. В качестве натренированной модели была выбрана нейронная сеть resnet-50, которая содержит 52 сверточных слоя, 49 слоев с функцией активации relu, 2 слоя с пространственным объединением и 1 полносвязный слой на 1000 нейронов с функцией активации softmax.

В данной лабораторной работе были проведены следующие эксперименты:

1. Использовались модели без изменений (за исключением того, что на последнем слое на выходе 2 нейрона вместо 1000), но с полным ее обучением. Веса инициализировались случайным образом.
2. Замена классификатора в исходной модели. Веса в нем инициализируются случайным образом. Оставшаяся часть модели используется как метод выделения признаков и данная часть модели не переобучается. В качестве нового классификатора был выбран классификатор с одним полносвязным скрытым слоем на 500 нейронов и функцией активации tanh и еще одним полносвязным слоем с функцией активации softmax.



1. Тонкая настройка параметров модели. В данном эксперименте обучается вся нейронная сеть. При этом классификатор заменяется на новый со случайными весами. А оставшаяся часть модели инициализируется весами из натренированной модели. В качестве нового классификатора был выбрал классификатор с одним полносвязным слоем на 10 нейронов с функцией активации softmax.



# Результаты экспериментов

Эксперименты проводились при следующих параметрах обучения:

batch\_size = 8,

optimizer = ‘sgd’,

learningrate = 0.01.

Тестовая инфраструктура

Вычисления производились на машине со следующими характеристиками:

* ОС: Windows 10
* Процессор: Intel(R) Core(TM) i7-6700k CPU @ 4.00GHz 4.01 GHz
* Установленная Память (ОЗУ): 16,0 ГБ
* Тип системы: 64 – разрядная операционная система, процессор х64
* Видеокарта: NVIDIA GeForce GTX 1070

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№** | **Количество эпох** | **Результат** | | |
| Точность на тренировочном множестве | Точность на тестовом множестве | Время, с |
| 1 | 5 | 0.74 | 0.81 | 1470,24 |
| 2 | 5 | 0.98 | 0.99 | 13,06 |
| 3 | 2 | 0.99 | 0.99 | 610,15 |

# Выводы

# Достигнута очень высокая точность, за исключением первого эксперимента, где обучалась сеть без изменений. В первом эксперименте случайная инициализация весов, а во втором и третьем использовалась сеть с предобученнными весами. Результаты во втором и третьем эксперименте получились примерно одинаковыми, но вторая сеть обучилась быстрее, так как обучались только последние 2 слоя сети.