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 глубоких нейронных сетей**

**Выполнили:**

студенты группы 381703-3м

Гладкова Татьяна

Крутоборежская Ирина

Крюкова Полина

Подчищаева Мария

Нижний Новгород

2018

Содержание

[Цели 3](#_Toc531714636)

[Задачи 4](#_Toc531714637)

[Решаемая задача 5](#_Toc531714638)

[Выбор библиотеки 6](#_Toc531714639)

[Метрика качества решения задачи 6](#_Toc531714640)

[Тренировочные и тестовые наборы данных 6](#_Toc531714641)

[Конфигурации нейронных сетей 7](#_Toc531714642)

[Разработанные программы/скрипты 7](#_Toc531714643)

[Результаты экспериментов 7](#_Toc531714644)

[Выводы 7](#_Toc531714645)

# 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 модели для решения исходной задачи.
2. Выполнение трех типов экспериментов по переносу знаний (типы экспериментов описаны в лекции).
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 python данные были преобразованы к размеру 128×128. С помощью скрипта im2rec.py, который входит в библиотеку MXNet, изображения были сконвертированы в формат .rec.

# Выбор библиотеки

Для выполнения лабораторных работ выбрана библиотека MXNet для языка программирования Python.

На этапе проверки корректности установки библиотеки выполнена разработка и запуск тестового примера сети для решения задачи классификации рукописных цифр набора данных MNIST. Достигнута точность 0.9225.

# Метрика качества решения задачи

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

# Тренировочные и тестовые наборы данных

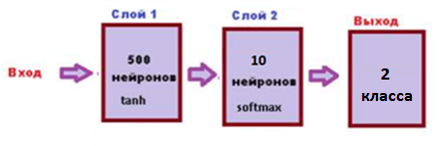
В качестве тренировочной выборки используем тренировочную выборку первого и второго наборов данных, всего 16500 изображений котов и 16505 изображений собак. В качестве тестовой выборки используем тестовую выборку только из первого набора данных, т.к. во втором наборе данных тестовая выборка не разбита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

# Тестовые конфигурации нейронных сетей

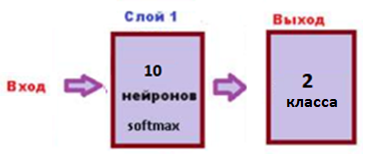
В качестве исходной задачи была выбрана задача классификации изображений на основе базы данных ImageNet, которая содержит 1000 классов различных изображений. В качестве натренированной модели была выбрана нейронная сеть resnext-50, которая содержит 52 сверточных слоя, 49 слоев с функцией активации relu, 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, но с полным ее переобучением. 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’,

learning rate = 0.01

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **№ эксперимента** | **Количество эпох** | **Результат** | | |
| Точность на тренировочном множестве | Точность на тестовом множестве | Время, с |
| 1 | 5 | 0.74 | 0.81 | 293.29 |
| 2 | 5 | 0.98 | 0.99 | 2.58 |
| 3 | 2 | 0.99 | 0.99 | 302.53 |

# 

# Выводы

Наилучшую точность показал третий эксперимент. Значит, обучающих данных хватило для тонкой настройки модели.