Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего профессионального образования

«Национальный исследовательский Нижегородский государственный университет им. Н.И. Лобачевского»

Институт Информационных Технологий, Математики и Механики

Направление: Фундаментальная информатика и информационные технологии

Магистерская программа: Когнитивные системы

**ОТЧЕТ**

по лабораторной работе №4

Тема:

**«Начальная настройка весов полносвязных и сверточных нейронных сетей»**

Выполнили:

Краснов Александр (гр. 381806м4)

Разин Вячеслав (гр. 381806м4)

Ахмедов Саркар (гр. 381806м4)

Проверила:

доцент, к.т.н. Кустикова В.Д.

Нижний Новгород

2019

Оглавление

[Введение 3](#_Toc32832572)

[Постановка задачи 4](#_Toc32832573)

[Описание набора данных 5](#_Toc32832574)

[Метрика качества решения задачи 6](#_Toc32832575)

[Подготовка данных для нейронной сети 7](#_Toc32832576)

[Тестовые конфигурации сетей 8](#_Toc32832577)

[Результаты экспериментов 10](#_Toc32832578)

[Анализ результатов 13](#_Toc32832579)

# Введение

Пусть - множество изображений, – множество номеров (или наименований) классов. Существует неизвестная *целевая зависимость* – отображение , значения которой известны только на объектах конечной обучающей выборки . Требуется построить алгоритм , способный классифицировать произвольный объект .

В рамках данной лабораторной работы ставится [задача классификации немецких дорожных знаков](https://www.kaggle.com/meowmeowmeowmeowmeow/gtsrb-german-traffic-sign) (The German Traffic Sign Benchmark).

# Постановка задачи

**Цели**

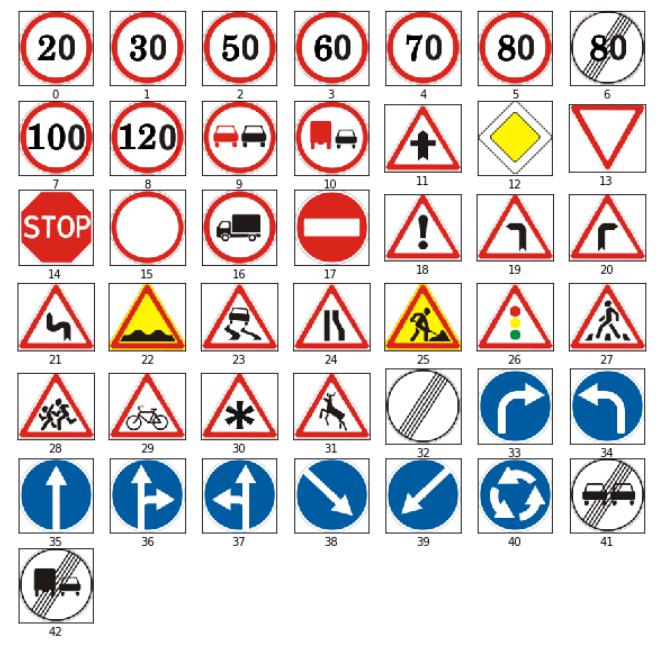
Цель настоящей работы состоит в том, чтобы использовать методы обучения без учителя для настройки начальных значений весов сетей, построенных при выполнении предшествующих практических работ.  
  
 **Задачи**

Выполнение практической 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.

Сбор результатов экспериментов.

# Описание набора данных

Для лабораторных работ был выбран один из представленных на Kaggle набор данных <https://www.kaggle.com/meowmeowmeowmeowmeow/gtsrb-german-traffic-sign> (The German Traffic Sign Benchmark).

*Рис.1. Представители всех классов набора данных The German Traffic Sign Benchmark*

Данный набор содержит в общей сложности более 51840 изображений разных разрешений, 39209 из которых используются для тренировки нейронной сети, а 12631 - для тестирования. На каждом изображении содержится один дорожный знак. Набор данных содержит 43 класса изображений.

Для реализации различных архитектур нейронных сетей была выбрана открытая библиотека глубокого обучения Keras, написанная на языке программирования Python.

# Метрика качества решения задачи

Для оценки качества задачи классификации выбрана метрика "" (точность), показывающая отношение числа правильных предсказаний к числу предсказаний класса для всех изображений:  
где – количество изображений, – значения, выданные сетью при классификации изображений, – истинные значения классов элементов одежды с изображения.

# Подготовка данных для нейронной сети

В силу большого количества входных изображений было решено сжать каждое из них до размера 50 x 50 пикселей. Сжатие происходит с сохранением соотношения сторон, уменьшенное изображение находится по центру, а недостающие пиксели имеют черный цвет. После этого преобразуем изображение (трёхмерный массив числе от 0 до 255 включительно) в один вектор и приводим каждый элемент вектора к значению с отрезка [0, 1].

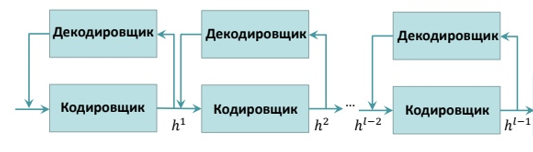
Изначально каждое изображение обучающей выборки находится в пронумерованной папке, номер которой соответствует конкретному классу изображений от 0 до 42. Используя метод one-hot encoding, представляем каждый класс как вектор состоящий из 43 чисел. В итоге мы имеем два двумерных массива: массив изображений, готовых к классификации, и массив меток, отвечающий за то, какому классу данные изображения принадлежат.

# Автокодировщик

В качестве алгоритма без учителя для инициализации начальных весов сети будем использовать автокодировщик.

Автокодировщик – нейронная сеть, которая пытается максимально приблизить значения выходного сигнала к входному, т.е. наилучшим образом аппроксимировать тождественное преобразование. Сеть разделяется на две принципиальные части: кодировщик h = f(x), обеспечивающий кодирование входных данных, и декодировщик y = g(h), восстанавливающий по коду вход. Cкрытый слой h описывает код, используемый для представления переданных на вход данных. Идеальный автокодировщик позволяет добиться равенства входа и выхода x = g(f(x)) и по существу обеспечивает копирование входного сигнала.

Чаще всего автокодировщики применяют каскадно для обучения глубоких (многослойных) сетей. Стекавтокодировщиков (рис. 2), может быть применен при работе с глубокими сетями. Каждый автокодировщик обучается как сеть прямого распространения и работает с одним слоем. Это позволяет постепенно снижать размерность и настраивать параметры.

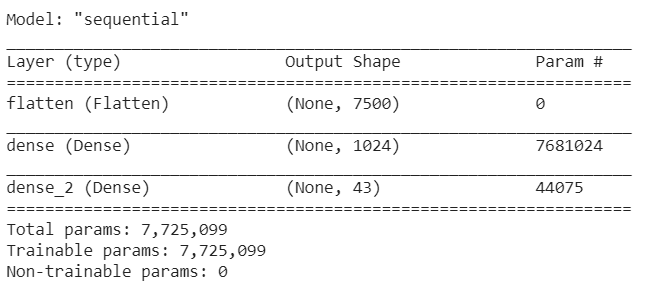


*Рис.2. Стек автокодировщиков*

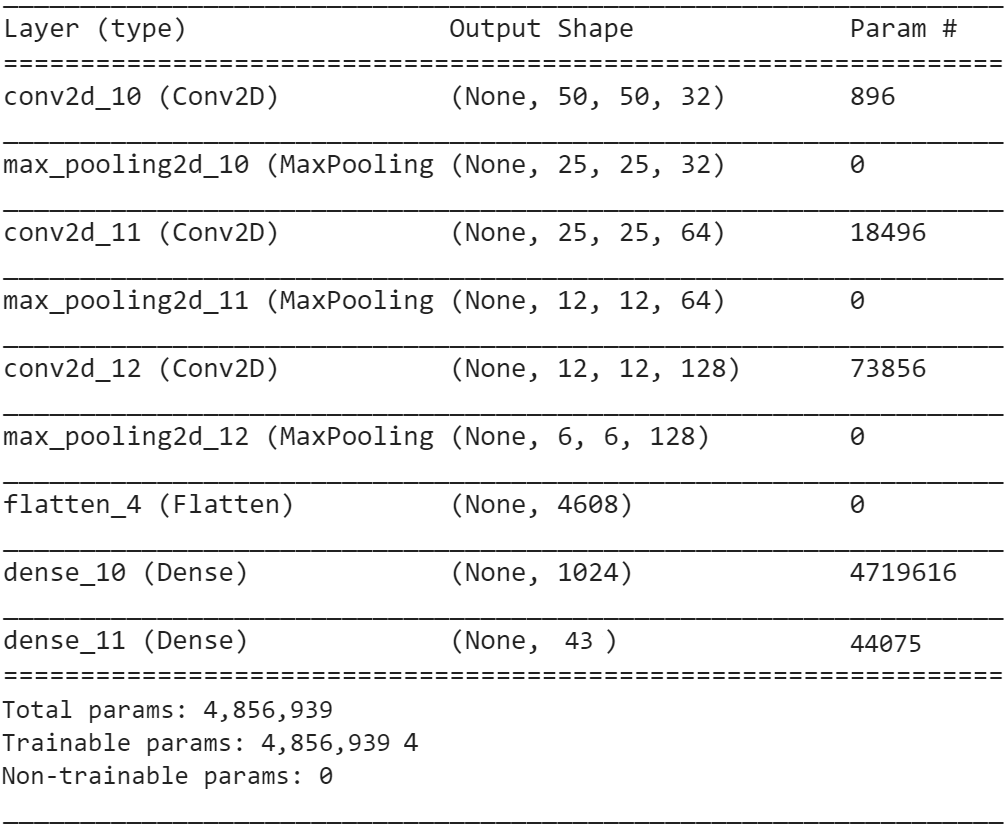
# Тестовые конфигурации сетей

В ходе лабораторной работы были рассмотрены шесть различных конфигураций сетей. Варьировались количество скрытых слоев и функции активации. Схемы конфигураций представлены ниже:

Модель FCNN:



Модель CNN:



# Результаты экспериментов

Все вычислительные эксперименты выполнялись с помощью Google Colaboratory режиме GPU.

Результаты полносвязных автокодировщиков:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Model | Activation | LR | Epochs | Batch size | Time | Accuracy |
| FCNN\_1 | Sigmoid | 0.001 | 10 | 256 | 19 c | **0.8637** |
| FCNN\_2 | Relu | 0.001 | 10 | 256 | 21 c | 0.8373 |
| FCNN\_3 | Tanh | 0.001 | 10 | 256 | 21 c | 0.8320 |

Результаты свёрточных автокодировщиков:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Model | Activation | LR | Epochs | Batch size | Time | Accuracy |
| CNN\_1 | Sigmoid | 0.001 | 10 | 256 | 38 c | 0.8387 |
| CNN\_2 | Relu | 0.001 | 10 | 256 | 38 c | 0.9311 |
| CNN\_3 | Tanh | 0.001 | 10 | 256 | 38 c | **0.9516** |

Суммарные результаты:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Model | Activation | LR | Epochs | Batch size | Time | Accuracy |
| FCNN\_1 | Sigmoid | 0.001 | 10 | 256 | 19 c | 0.8637 |
| CNN\_1 | Tanh | 0.001 | 10 | 256 | 38 c | 0.9516 |

# Анализ результатов

В данной лабораторной работе удалось улучшить точность предсказаний моделей, разработанных в прошлых работах, выполнив начальную настройку весов сетей с помощью автокодировщиков.