

Нижегородский государственный университет им. Н. И. Лобачевского  
Институт информационных технологий, математики и механики

Направление подготовки Прикладная математика и информатика  
Магистерская программа Вычислительные методы и суперкомпьютерные  
технологии

Образовательный курс «Методы глубокого обучения для решения задач  
компьютерного зрения»

## **Отчёт**

по лабораторной работе № 4

### **«Начальная настройка весов полностью связанных нейронных сетей»**

задача

«Классификация персонажей Симпсонов»

***Выполнили:***

студенты гр. 381603м4

Вершинина О.

Розанов А.

Рой В.

Нижний Новгород  
2017

## Оглавление

Постановка задачи .....	3
Формат данных для предоставления нейронной сети .....	4
Тестовые конфигурации нейронных сетей .....	5
Результаты экспериментов .....	7

## **Постановка задачи**

Необходимо использовать методы обучения без учителя, в частности автокодировщики, для начальной настройки значений весов полностью связанных нейронных сетей, построенных для задачи классификации персонажей из мультфильма «Симпсоны».

В ходе работы необходимо решить следующие задачи:

1. Выбрать архитектуры полностью связанных нейронных сетей, построенных при выполнении лабораторной работы №2.
2. Разработать модели автокодировщиков применительно к выбранным архитектурам нейронных сетей.
3. Обучить построенные глубокие модели.
4. Обучить выбранные архитектуры с начальной инициализацией весов сетей значениями, полученными в ходе обучения без учителя.
5. Протестировать обученные нейронные сети.

## Формат данных для предоставления нейронной сети

Исходные данные представляют собой набор jpg изображений, различного разрешения. Несколько примеров представлены на рис. 1.

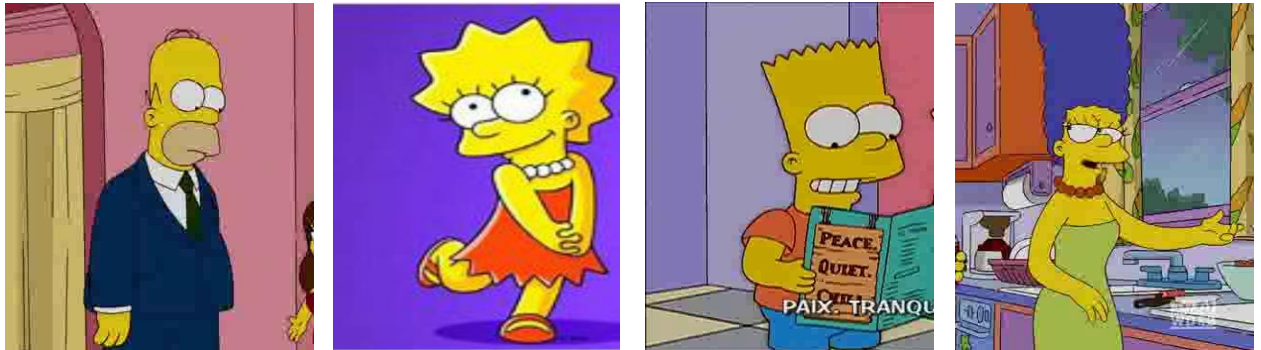


Рис.1. Персонажи из Симпсонов. Слева-направо: Гомер Симпсон, Лиза Симпсон, Барт Симпсон, Мардж Симпсон.

Для предварительной обработки данных использован скриптовый язык Python. Интерпретатор языка входит в дистрибутив Anaconda, который содержит ряд пакетов для анализа данных и машинного обучения.

Библиотека глубокого обучения MXNet может работать с различными типами входных данных, в том числе с однородными многомерными массивами ndarray из пакета NumPy. Для того, чтобы привести входные данные к такому формату, необходимо использовать библиотеку OpenCV для Python (opencv-python).

Был разработан скрипт, в котором с помощью функции cv2.imread считываются трёхканальные .jpg изображения в формате BGR (стандартное цветовое пространство OpenCV) и конвертируются в формат RGB; нормализуются (значение каждого пикселя делится на 255) и масштабируются до размера 28x28. Формируется массив меток от 0 до 17, соответствующих восемнадцати персонажам. Затем данные случайным образом делятся на обучающую и тестовую выборки в отношении 85% к 15%. Полученные ndarray-массивы X\_train, X\_test, y\_train, y\_test сохраняются в файлы для последующего использования нейронной сетью.

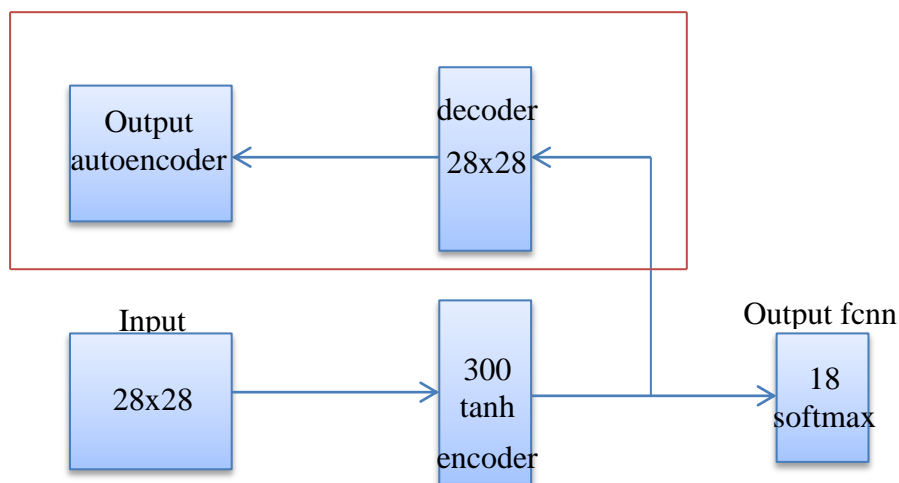
## Тестовые конфигурации нейронных сетей

Начальная настройка весов с помощью автокодировщиков была реализована для конфигураций полностью связанных нейронных сетей, рассмотренных в лабораторной работе №2.

Конфигурация №1 из лабораторной работы №2 представляет собой полностью связанную нейронную сеть с одним скрытым слоем (300 скрытых нейронов, функция активации – tanh). Для данной сети был построен автокодировщик, состоящий из кодирующего слоя – скрытый слой исходной сети, и декодирующего слоя – слой, обратный к кодирующему. На вход автокодировщику подавались изображения персонажей Симпсонов и сеть обучалась максимально приближать значения выходного сигнала к входному. После обучения веса кодирующего слоя запоминались и использовались для инициализации весов в исходной нейронной сети.

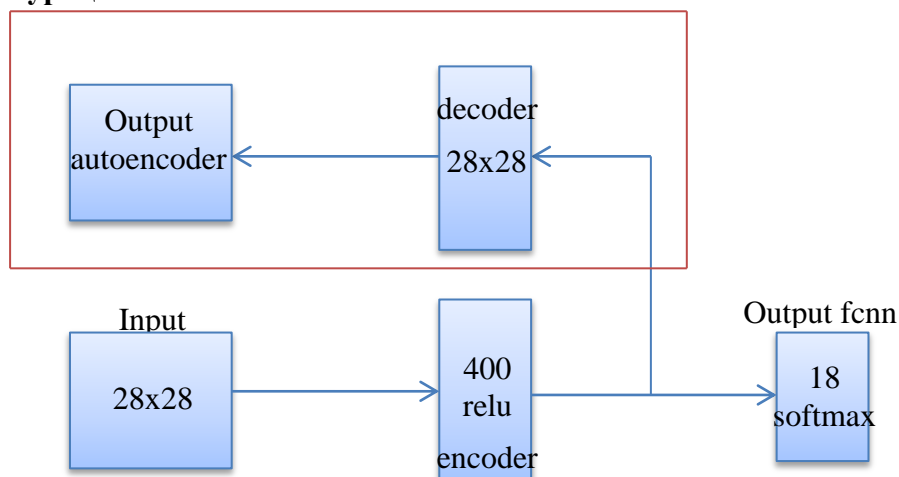
Конфигурация А представляет собой визуальную схему полностью связанной нейронной сети с одним скрытым слоем и построенного к ней автокодировщика.

### Конфигурация А.



Конфигурация В – аналогичная схема полностью связанной нейронной сети (конфигурация №4 из лабораторной работы №2) с одним скрытым слоем и построенного к ней автокодировщика.

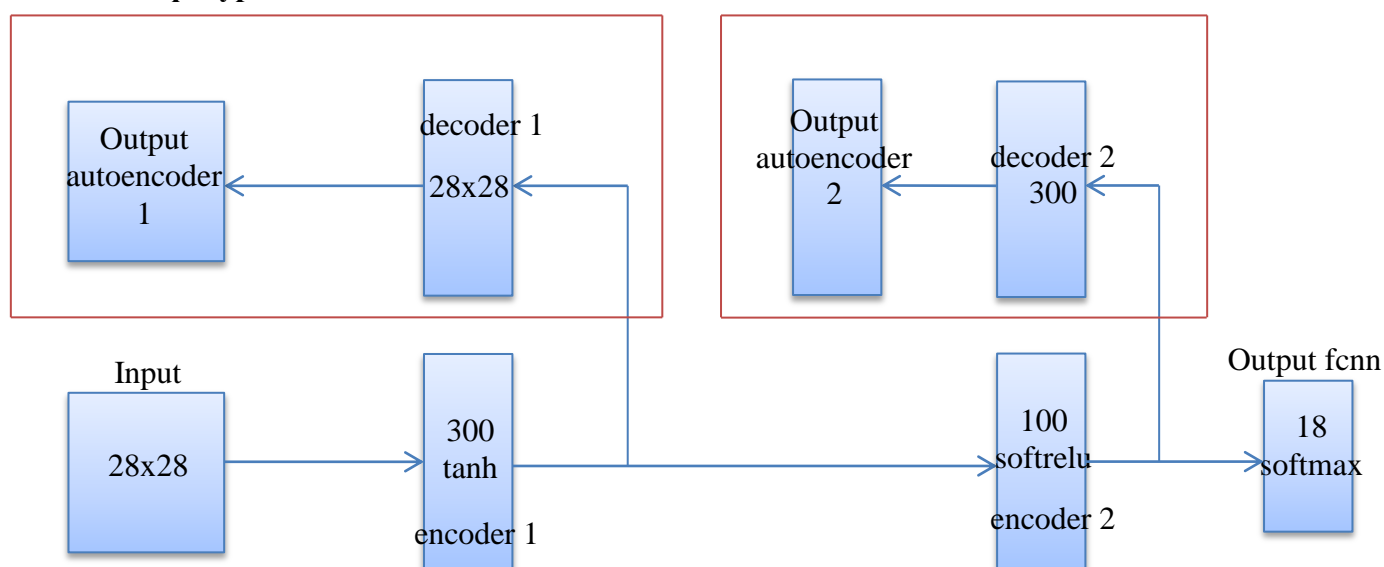
### Конфигурация В.



Конфигурация №2 из лабораторной работы №2 представляет собой полностью связанную нейронную сеть с двумя скрытыми слоями (первый слой: 300 скрытых нейронов, функция активации – tanh; второй слой: 100 скрытых нейронов, функция активации – softrelu). Для настройки параметров такой многослойной сети строился стек автокодировщиков. Каждый автокодировщик обучался последовательно как сеть прямого распространения.

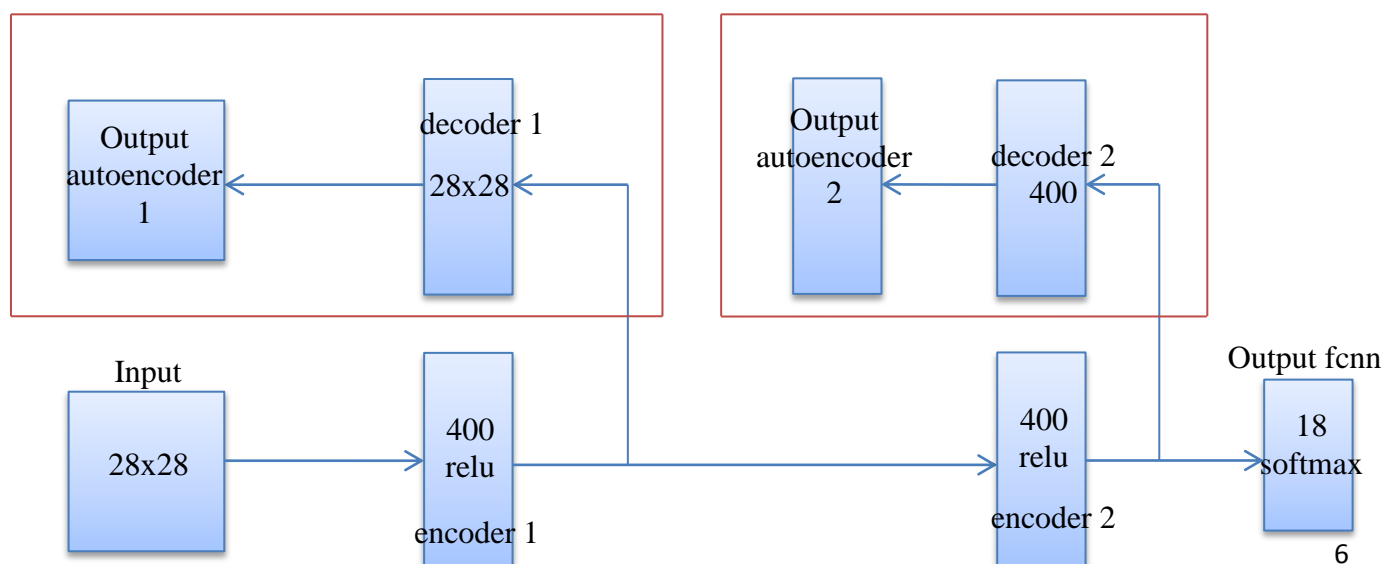
Конфигурация С представляет собой визуальную схему полностью связанной многослойной нейронной сети и построенного к ней стека автокодировщиков.

### Конфигурация С.



Конфигурация D – аналогичная схема полностью связанной многослойной нейронной сети (конфигурация №5 из лабораторной работы №2) и построенного к ней стека автокодировщиков.

### Конфигурация D.



## Результаты экспериментов

Эксперименты проводились при следующих параметрах обучения:  $\text{batch size} = 10$ ,  $\text{optimizer} = \text{'sgd'}$ ,  $\text{learning rate} = 0.01$ . Обучение сети проводилось от начального приближения весов, построенного с помощью автокодировщиков, до тех пор, пока точность на тренировочной выборке не становилась равной 1. В таблице 1 указаны результаты экспериментов. В последней колонке в скобках указана точность классификации сети без начального приближения весов.

Таблица 1. Результаты экспериментов

Конфигурация №	Время обучения модели, с	Точность классификации на тестовой выборке
A	850	0.8236 (0.8068)
B	935	0.8334 (0.8167)
C	980	0.8352 (0.8194)
D	1206	0.8585 (0.8327)

Из таблицы видно, что с начальной инициализацией весов значениями, полученными в ходе обучения без учителя, точность классификации таких сетей увеличилась по сравнению со случаем, когда веса были инициализированы случайными значениями.