ФЕДЕРАЛЬНОЕ АГЕНСТВО СВЯЗИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

Московский технический университет связи и информатики

(МТУСИ)

Кафедра «Математическая кибернетика и информационные технологии» Дисциплина «Интеллектуальные системы»

Лабораторная работа № 5

Распознавание объектов на фотографиях

Выполнил: \_\_\_\_\_\_\_ Д.С. Калининский

Проверил: \_\_\_\_\_\_\_ А.В. Шереметьев

Москва 2022

**Оглавление**

[1 Цель работы 3](#_Toc119580633)

[2 Задачи 3](#_Toc119580634)

[3 Выполнение работы 3](#_Toc119580635)

[4 Результаты 6](#_Toc119580636)

[5 Вывод 7](#_Toc119580637)

# **1 Цель работы**

Распознавание объектов на фотографиях (Object Recognition in Photographs) CIFAR-10 (классификация небольших изображений по десяти классам: самолет, автомобиль, птица, кошка, олень, собака, лягушка, лошадь, корабль и грузовик) (рисунок 1).

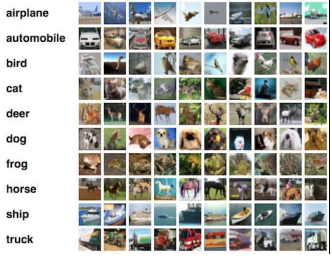


Рисунок 1 – Пример данных

# **2 Задачи**

1) Ознакомиться со сверточными нейронными сетями;

2) Изучить построение модели в Keras в функциональном виде;

3) Изучить работу слоя разреживания (Dropout).

# **3 Выполнение работы**

Построим глубокую сверточную нейронную сеть и применим ее к классификации изображений из набора CIFAR-10 (рисунок 2).

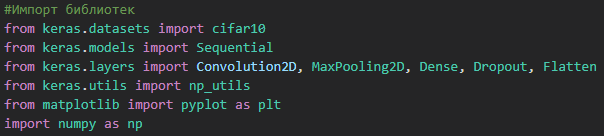


Рисунок 2 – Импорт данных

Как уже говорилось, обычно CNN использует больше гиперпараметров, чем MLP.

Зададим следующие гиперпараметры:

batch\_size — количество обучающих образцов, обрабатываемых одновременно за одну

итерацию алгоритма градиентного спуска;

num\_epochs — количество итераций обучающего алгоритма по всему обучающему

множеству;

kernel\_size — размер ядра в сверточных слоях;

pool\_size — размер подвыборки в слоях подвыборки;

сonv\_depth — количество ядер в сверточных слоях;

drop\_prob (dropout probability) — мы будем применять dropout после каждого слоя подвыборки, а также после полносвязного слоя;

hidden\_size — количество нейронов в полносвязном слое MLP.

Загрузка и первичная обработка CIFAR-10 осуществляется ровно так же, как и загрузка и обработка MNIST, где Keras выполняет все автоматически. Единственное отличие состоит в том, что теперь мы не рассматриваем каждый пиксель как независимое входное значение, и поэтому мы не переносим изображение в одномерное пространство. Мы снова преобразуем интенсивность пикселей так, чтобы она попадала в отрезок [0,1] и используем прямое кодирование для выходных значений.

Тем не менее, в этот раз этот этап будет выполнен для более общего случая, что позволит проще приспосабливаться к новым наборам данных: размер будет не жестко задан, а вычислен из размера набора данных, количество классов будет определено по количеству уникальных меток в обучающем множестве, а нормализация будет выполнена путем деления всех элементов на максимальное значение обучающего множества (рисунок 3).

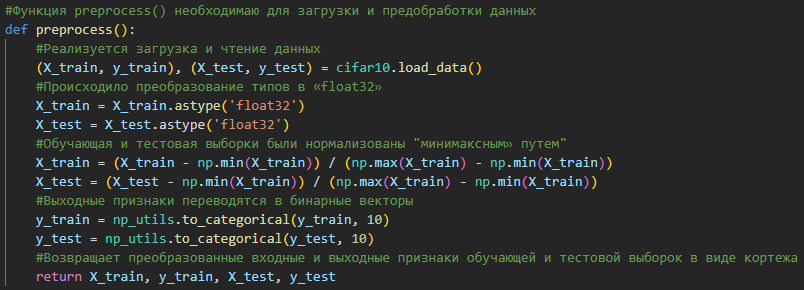


Рисунок 3 – Функция для загрузки и предобработки данных

Настало время моделирования! Наша сеть будет состоять из четырех слоев Convolution\_2D и слоев MaxPooling2D после второй и четвертой сверток. После первого слоя подвыборки мы удваиваем количество ядер (вместе с описанным выше принципом принесения высоты и ширины в жертву глубине). После этого выходное изображение слоя подвыборки трансформируется в одномерный вектор (слоем Flatten) и проходит два полносвязных слоя (Dense). На всех слоях, кроме выходного полносвязного слоя, используется функция активации ReLU, последний же слой использует softmax.

Для регуляризации нашей модели после каждого слоя подвыборки и первого полносвязного слоя применяется слой Dropout. Здесь Keras также выделяется на фоне остальных фреймворков: в нем есть внутренний флаг, который автоматически включает и выключает dropout, в зависимости от того, находится модель в фазе обучения или тестирования (рисунок 4).

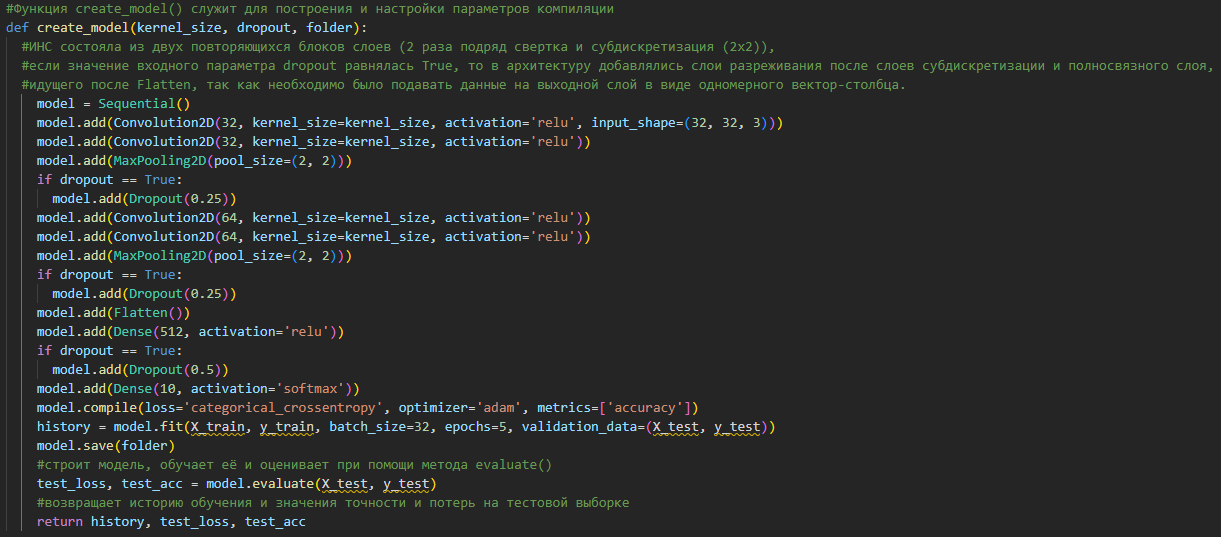


Рисунок 4 – Функция для построения и настройки параметров компиляции

# **4 Результаты**

Для получения визуальных результатов на рисунках 5-6 представлены графики, используем библиотеки matplotlib и numpy.

Таблица 1 – Результаты тестирования

|  |  |  |  |
| --- | --- | --- | --- |
| Размер ядра свертки | Наличие слоя Dropout | Точность | Потери |
| (2, 2) | True | 0.443301 | 197.464157 |
| (2, 2) | False | 0.434899 | 368.358581 |
| (2, 3) | True | 0.431801 | 186.036895 |

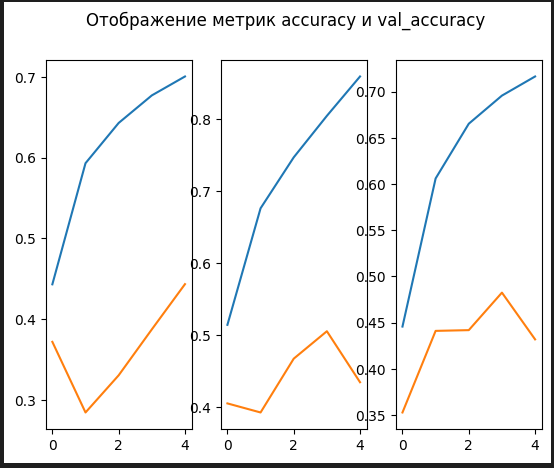


Рисунок 10 – Графики точностей моделей

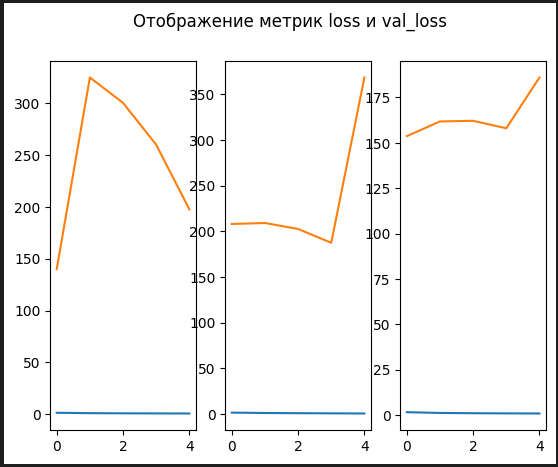


Рисунок 11 – Графики потерей моделей

# **5 Вывод**

В данной лабораторной работе, было реализовано распознавание объектов на фотографиях CIFAR-10, ознакомились со сверточными нейронными сетями, изучили работу слоя разреживания, была построена и обучена сверточная нейронная сеть, исследованы работа сети без слоя Dropout и работа сети при разных размерах ядра свертки.

В хоте тестирования моделей при наличие слоя Dropout, потерей значительно меньше, а точность незначительно увеличилось и при увеличении размера ядра свертки потерей становится меньше, а точность также уменьшилась.