Лабораторная работа № 4

Выполнил: Поздняков Иван Олегович

Группа: МПИ-20-4-2

Ссылка на гитхаб:

**Ход работы**

1. Выбрать/разработать архитектуру сверточной нейронной сети(CNN) для классификации цветных изображений.

Описание архитектуры CNN:

Количество слоёв: 5

Функция активации: relu и softmax ( на выходном слое)

, начальные значения весов

Устройство слоёв:

1. Conv2D(64, (3, 3)
2. Conv2D(128, (3, 3)
3. Conv2D(256, (3, 3)
4. Flatten(),Dense(1024)
5. Dense(num\_classes)

Таким образом количество нейронов равно первому числу в скобках, но т.к. фотографии цветные, то нужно учесть все три измерения RGB. Следовательно, количество нейронов в первых трёх слоях равны произведению трёх чисел в скобках.

Четвёртый слой состоит из одного измерения. Количество нейронов в нём = 1024.

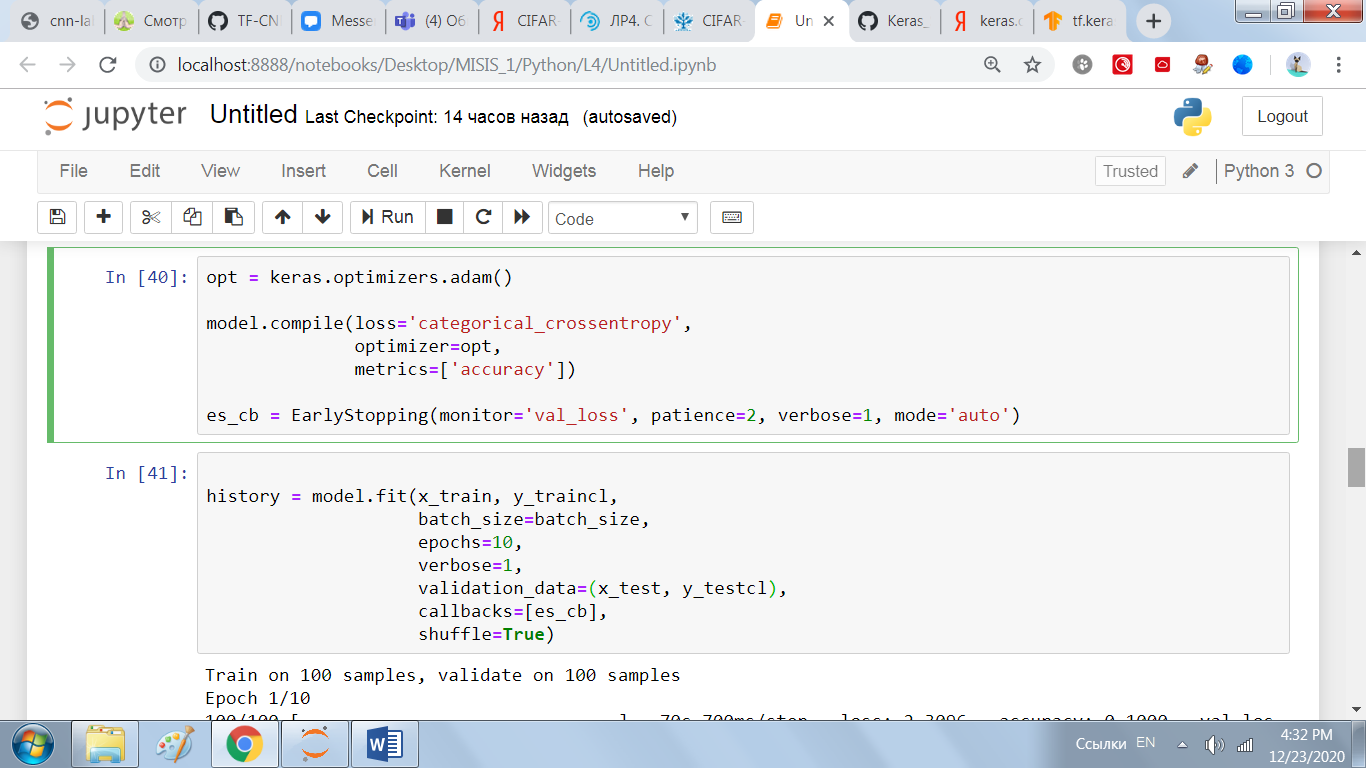
В пятом слое необходимо получить ответ, поэтому туда передаётся константа, котороя равна 10, т.к. количество классов = 10.

Исходя из статистики, взятой с сайта национального института стандартов США, были проанализированы 8 главных методов применяемых для распознавания объектов на фото. В данной лабораторной работе использовалась сеть такого вида т.к. она имела наибольшую точность для датасета CIFAR-10 на 2007 год.

2. Решить задачу классификации изображений на эталонном датасете CIFAR-10.

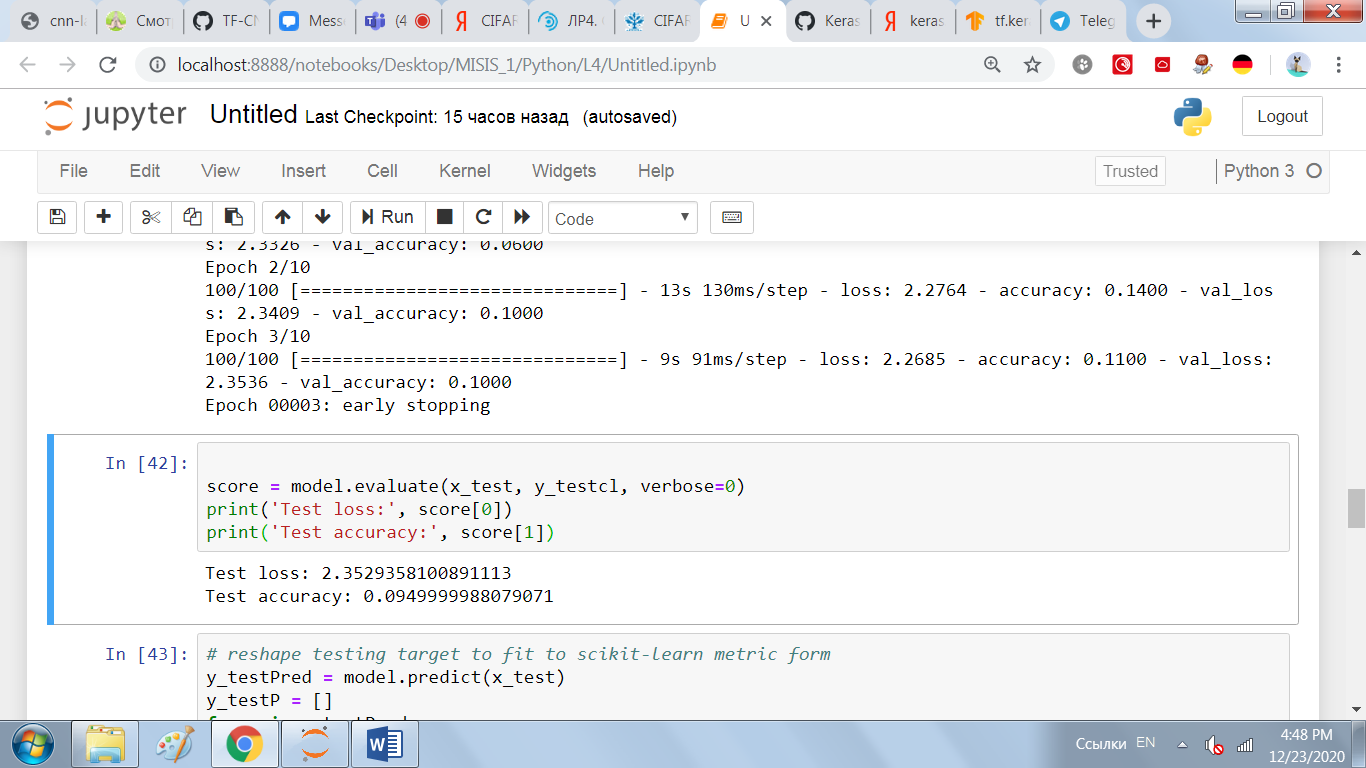
При заданном количестве слоёв, нейронов, функций активаций следует создать и обучить модель для снятия метрик качества.

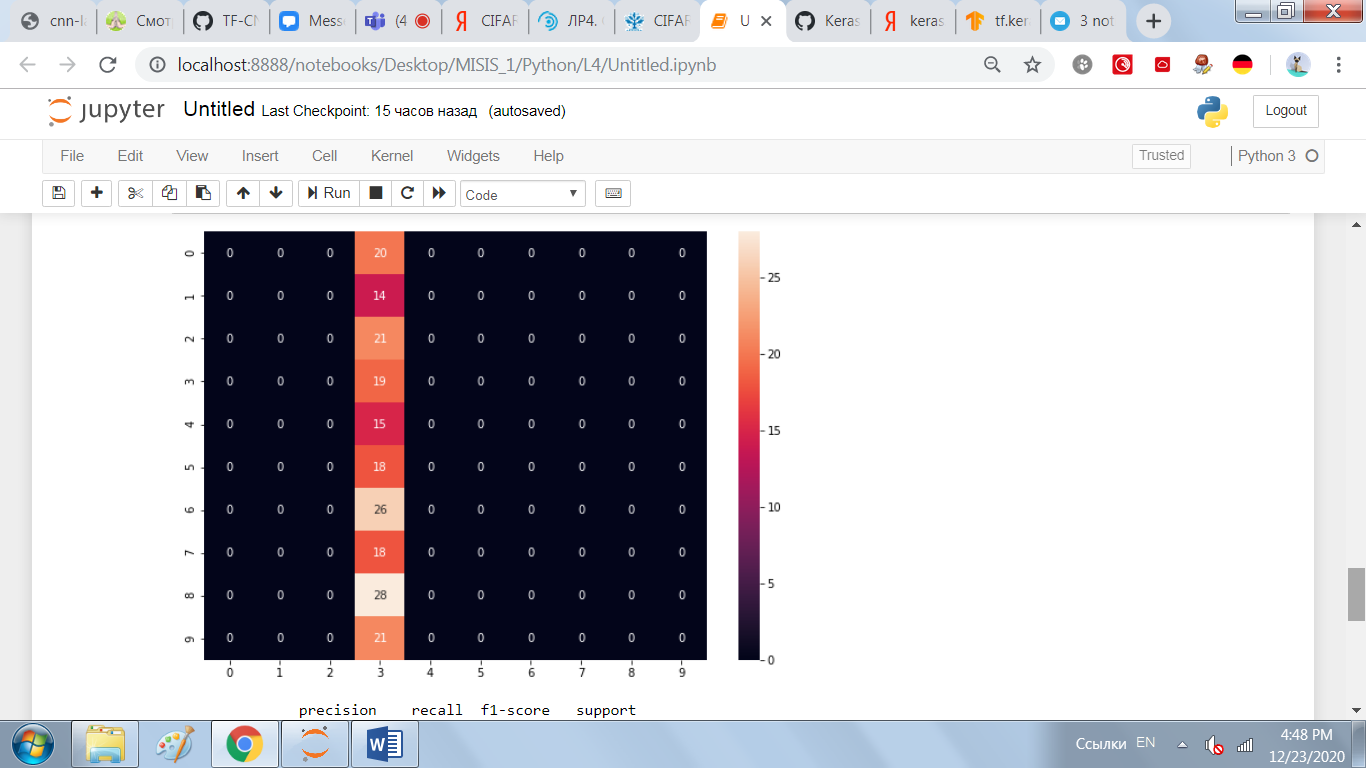
Т.к. размер датасета CIFAR-10 слишком большой, то я буду обучать свою модель на 500 тренировочных снимков и 200 снимков буду использовать для проверки.

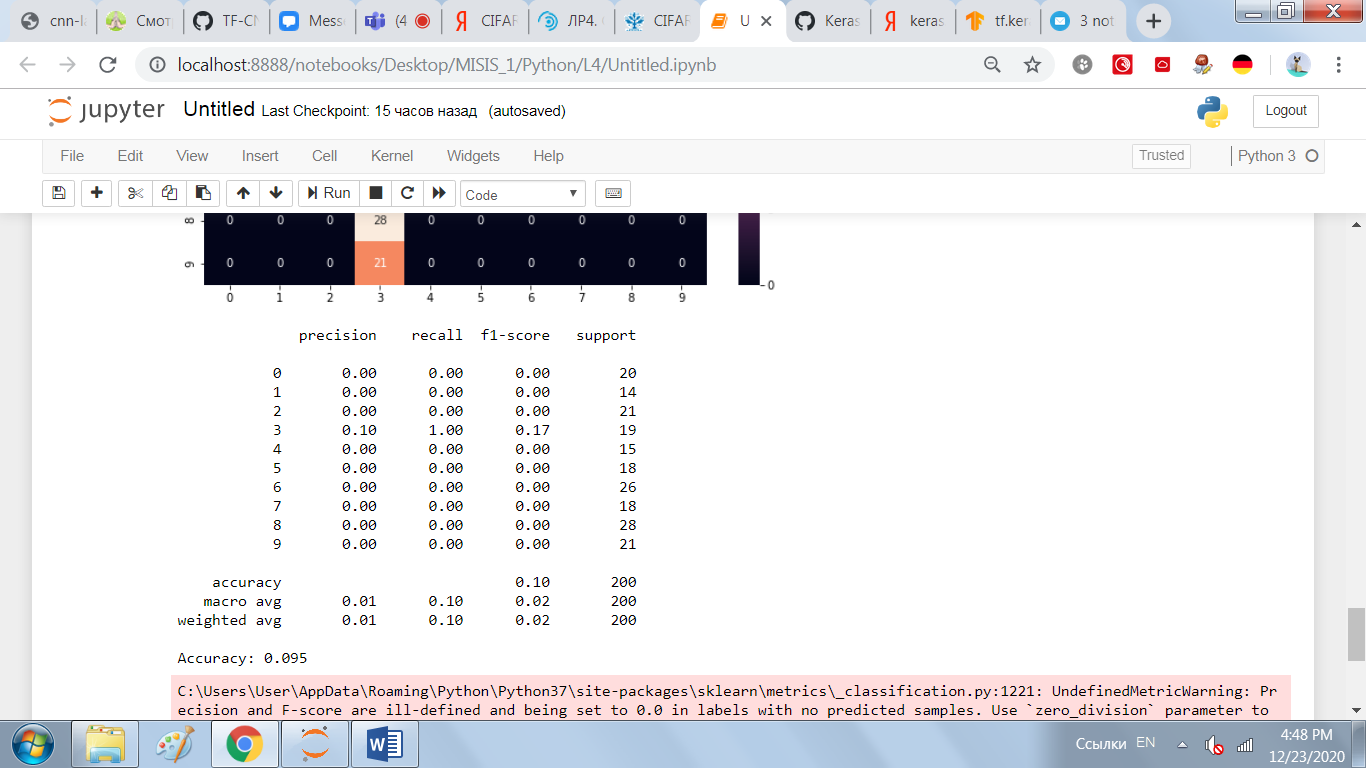


В моей моделия использовал алгоритм Адама для оптимайзера, функцию потерь categorical\_crossentropy, функцию EarlyStopping для завершения обучения если нет улучшения показателей и 10 эпох.

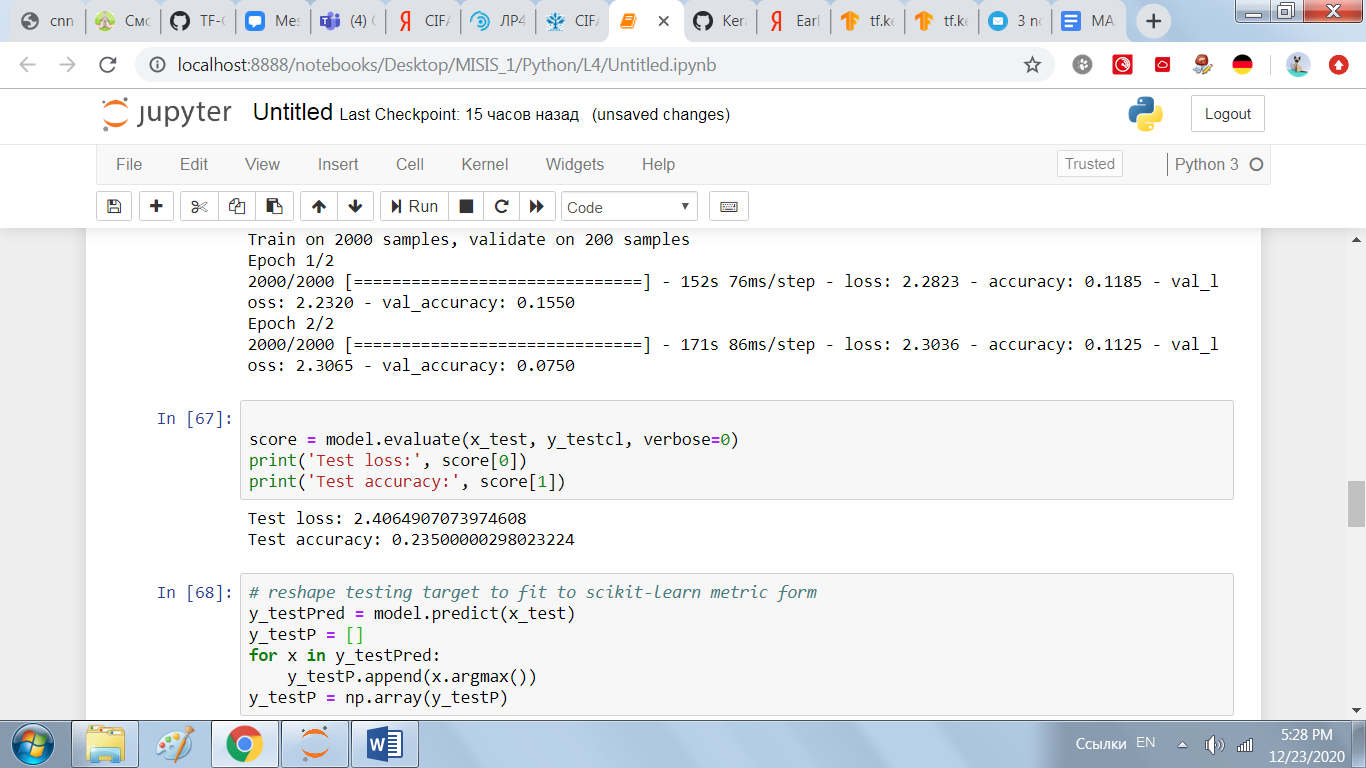
При заданных параметрах получаются следующие результаты на проверочной БД:







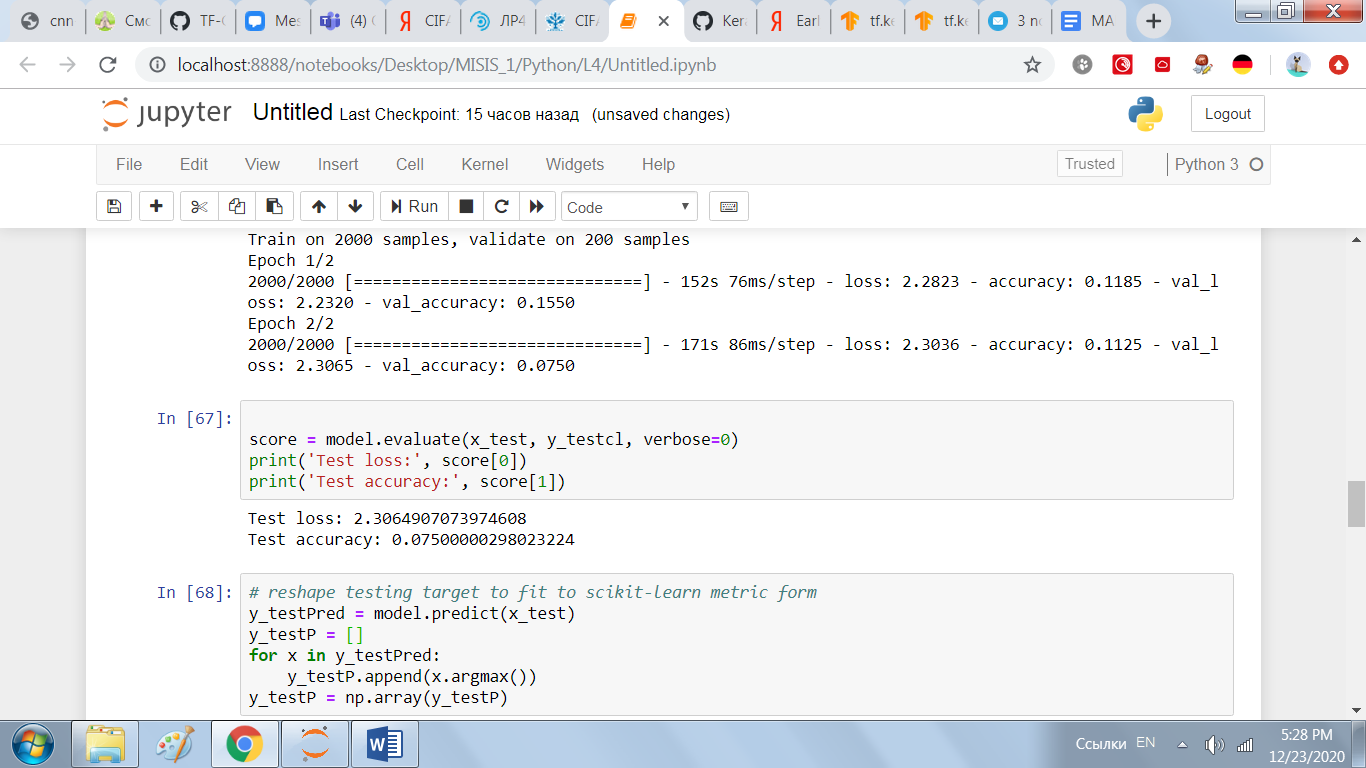
На тренировочной:

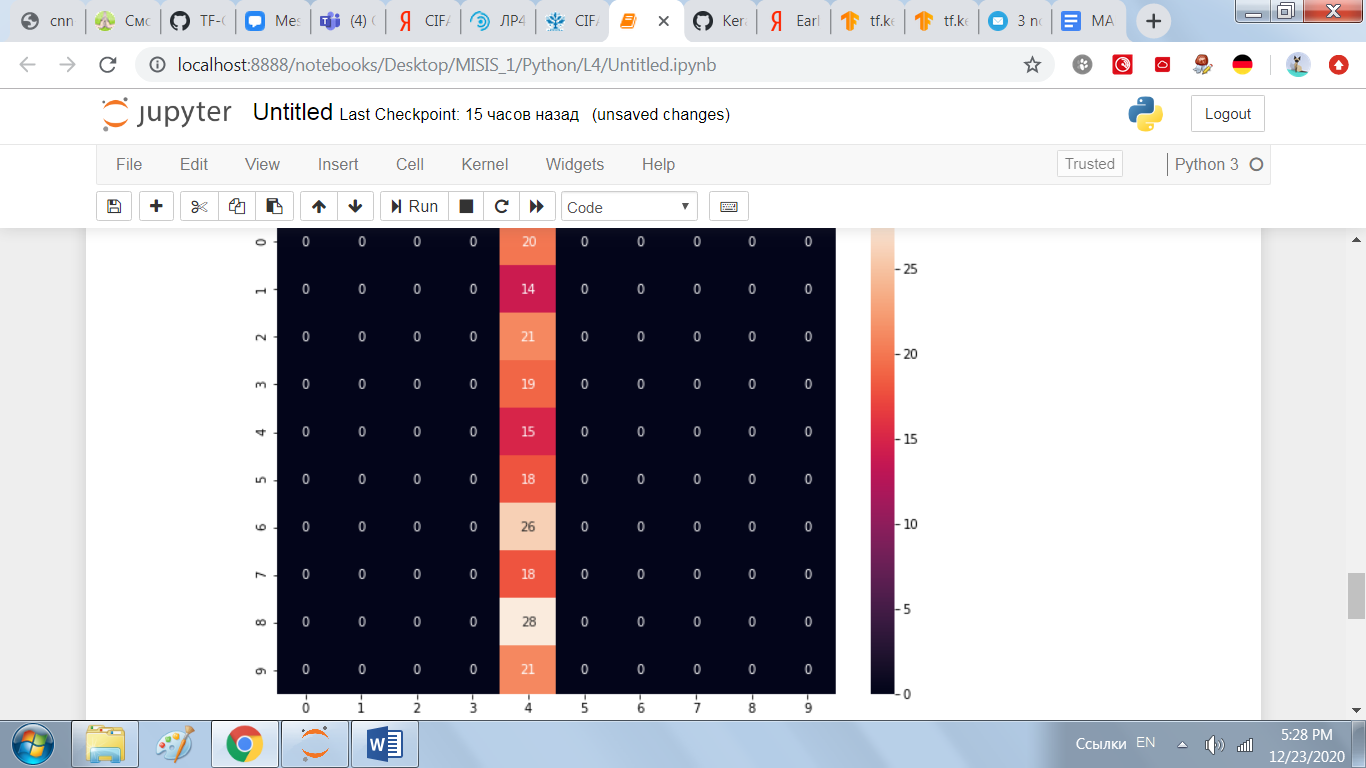


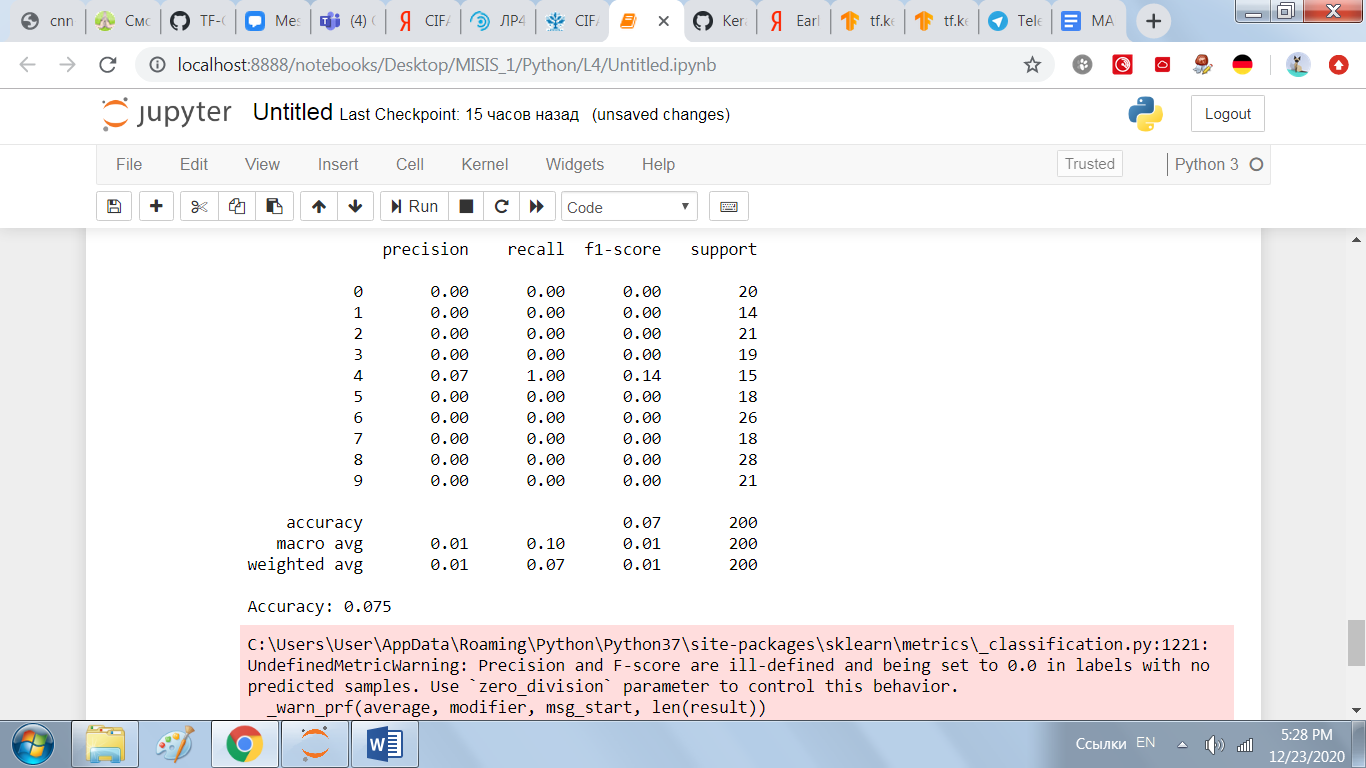
В первую очередь низкие показатели точности связаны с маленькой выборкой. Т.к. мой компьютер будет обрабатывать весть датасет несколько часов, то пришлось снизить выборку с 50 000 до 700 фотографий.

При увеличении тернеровочного датасета в 4 раза и проверочного в два будут следующие результаты:

Как можно увидеть, функция потерь уменьшается, но точность уменьшилась.







Т.к. размер тренировочной базы в 2000 фотографий слишком серьёзен для моего компьютера, то пришлось снизить количество эпох, это повлекло снижение точности.

**Изменение архитектуры CNN для увеличения показателей точности**

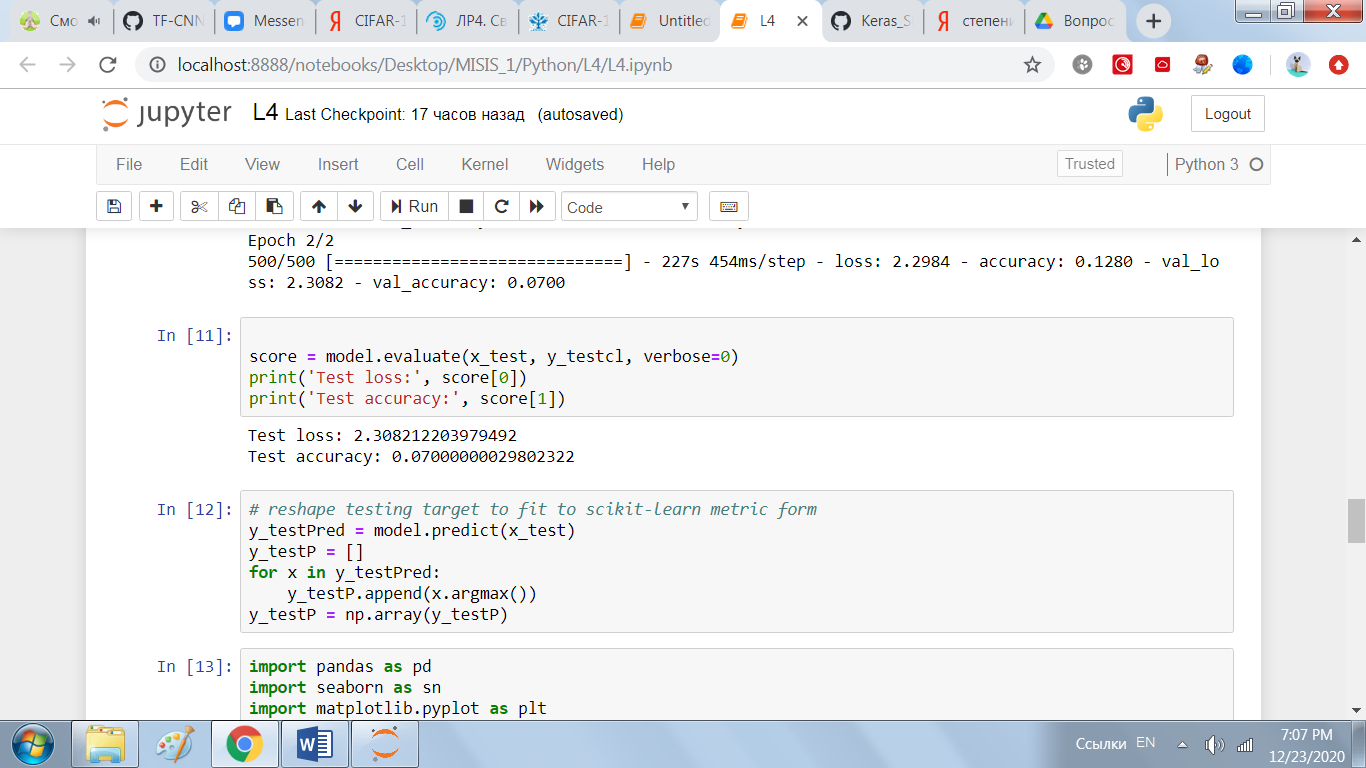
Изначальное устройство слоёв:

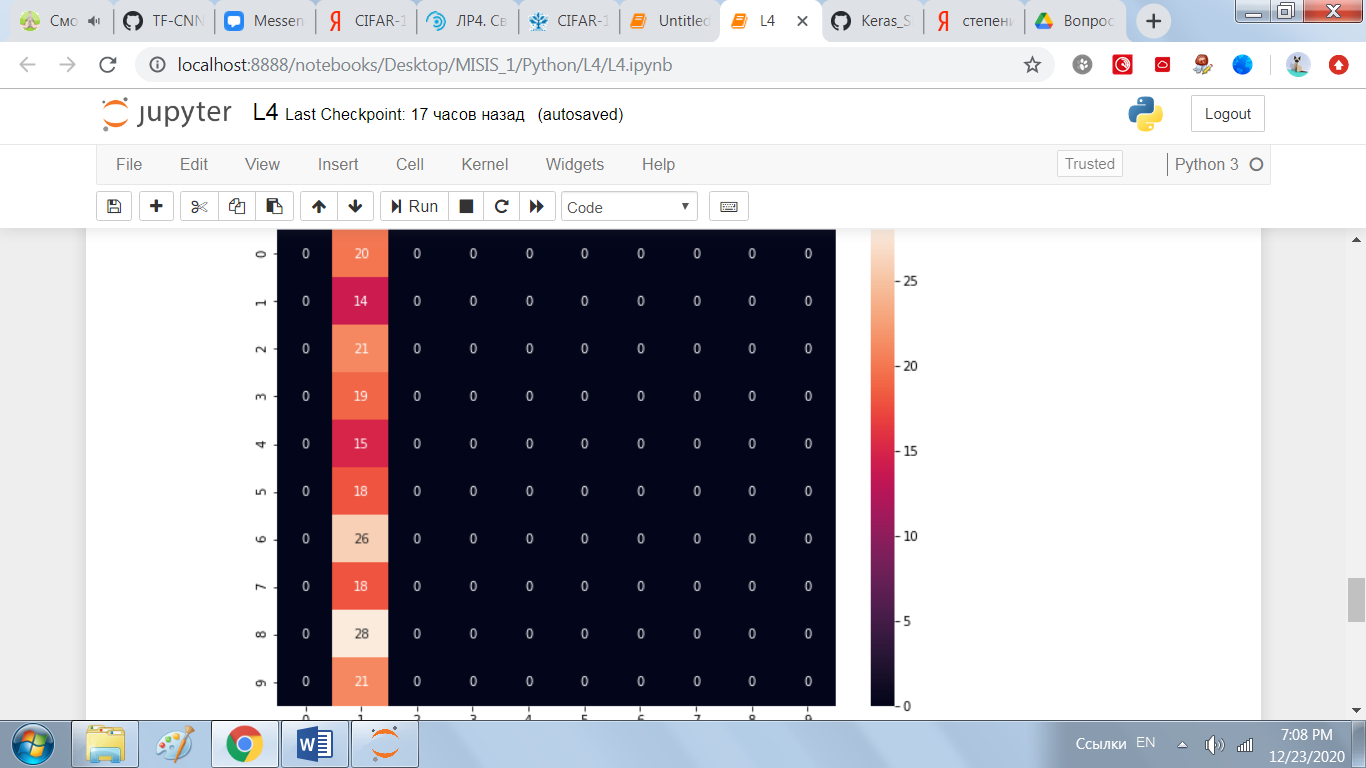
1. Conv2D(64, (3, 3)
2. Conv2D(128, (3, 3)
3. Conv2D(256, (3, 3)
4. Flatten(),Dense(1024)
5. Dense(num\_classes)

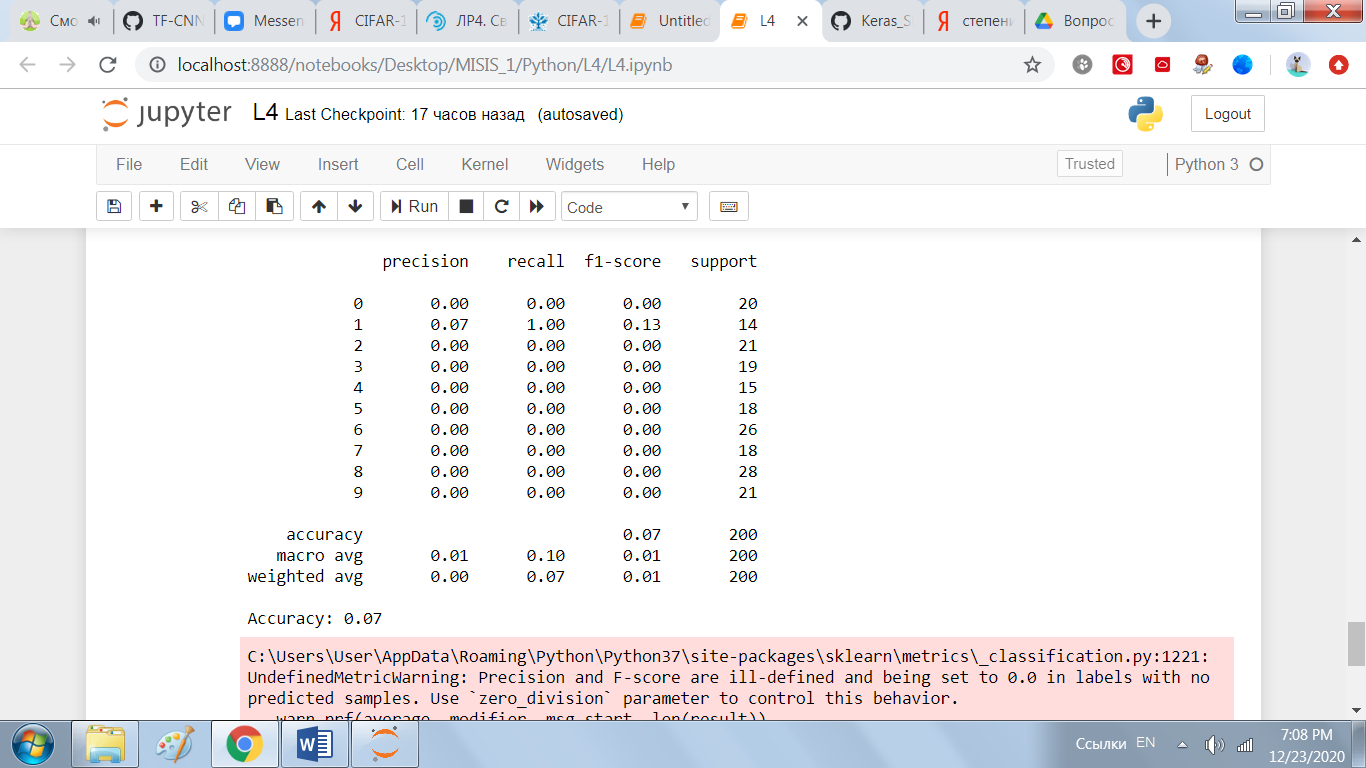
Можно заметить, что в каждом слое количество нейронов зависит от степени двойки, необходимо снять метрики при увеличении и уменьшении количества нейронов для нахождения оптимального решения.

1. Увеличение

1. Conv2D(64, (3, 3)
2. Conv2D(256, (3, 3)
3. Conv2D(1024, (3, 3)
4. Flatten(),Dense(1024)
5. Dense(num\_classes)





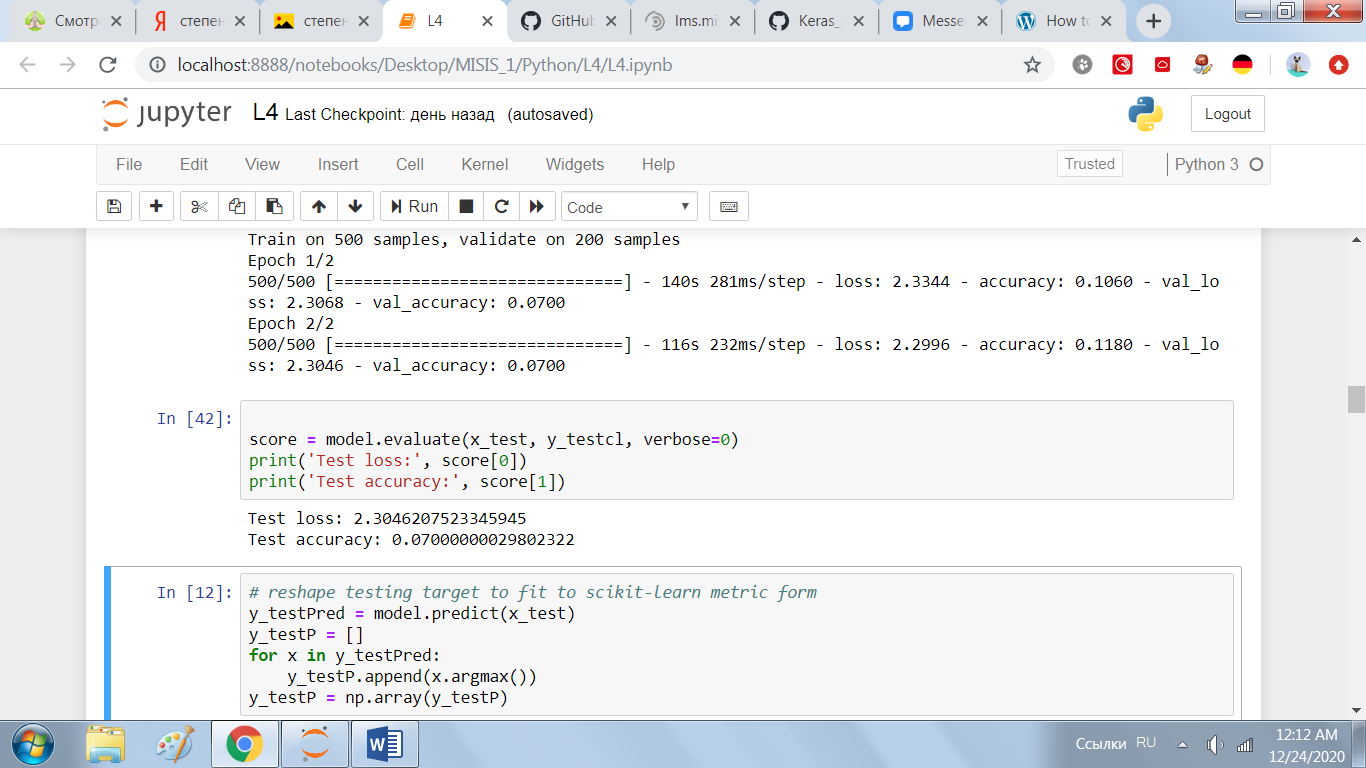


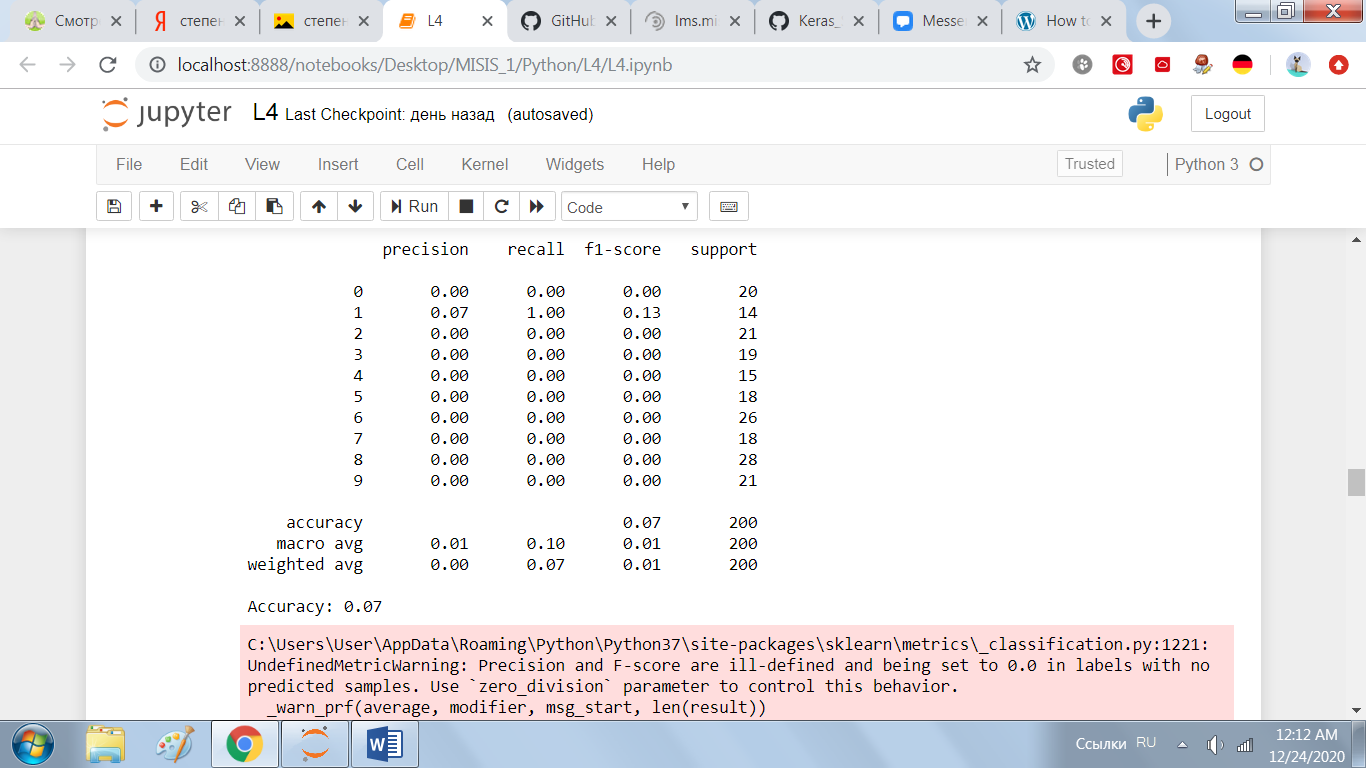
При сравнении с предыдущим результатом, уровень потерь остался почти одинаковым, но точность снизилась.

Соответственно, предыдущая архитектура была лучше.

2. Уменьшение

1. Conv2D(64, (3, 3)
2. Conv2D(128, (3, 3)
3. Conv2D(128, (3, 3)
4. Flatten(),Dense(512)
5. Dense(num\_classes)

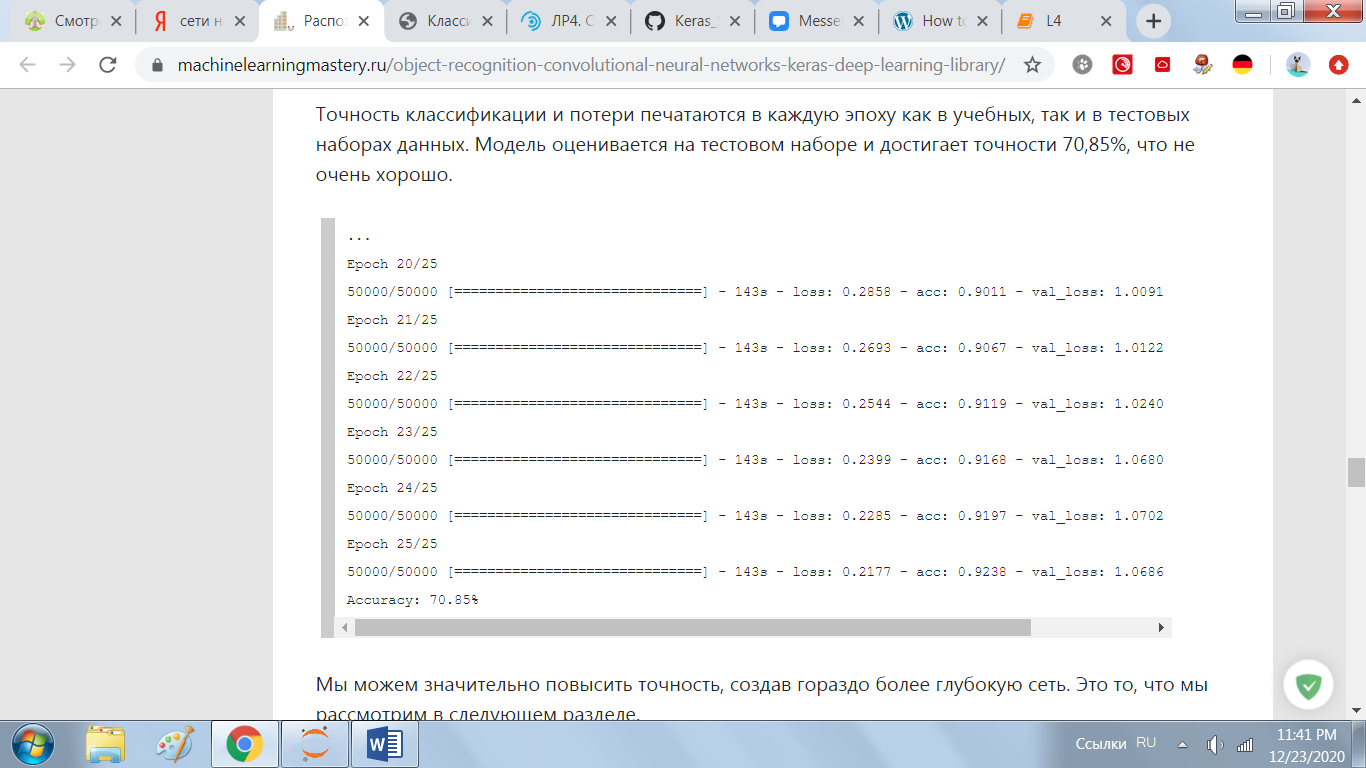




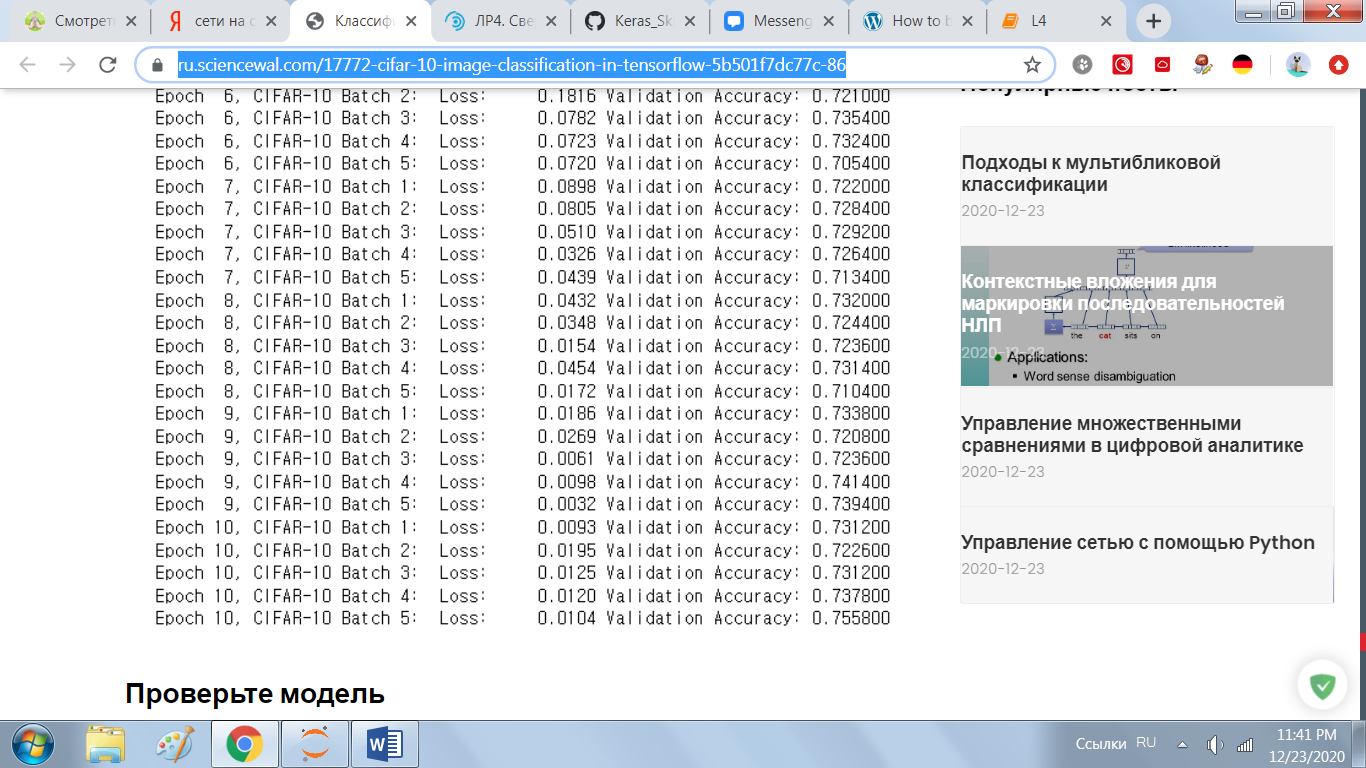
Можно заметить, что при увеличении количество нейронов и при уменьшении результаты очень схожы, только при уменьшении функция потерь меньше. Следовательно, оптимальный вариант находится между двумя этими вариантами, что и демонстрирует моя исходная свёрточная сетью с наивысшим сред них показателем точности.

3. Полученные результаты (см. раздел "Результаты") сравнить с опубликованными результатами для аналогичных/похожих архитектур и дать на них ссылку.

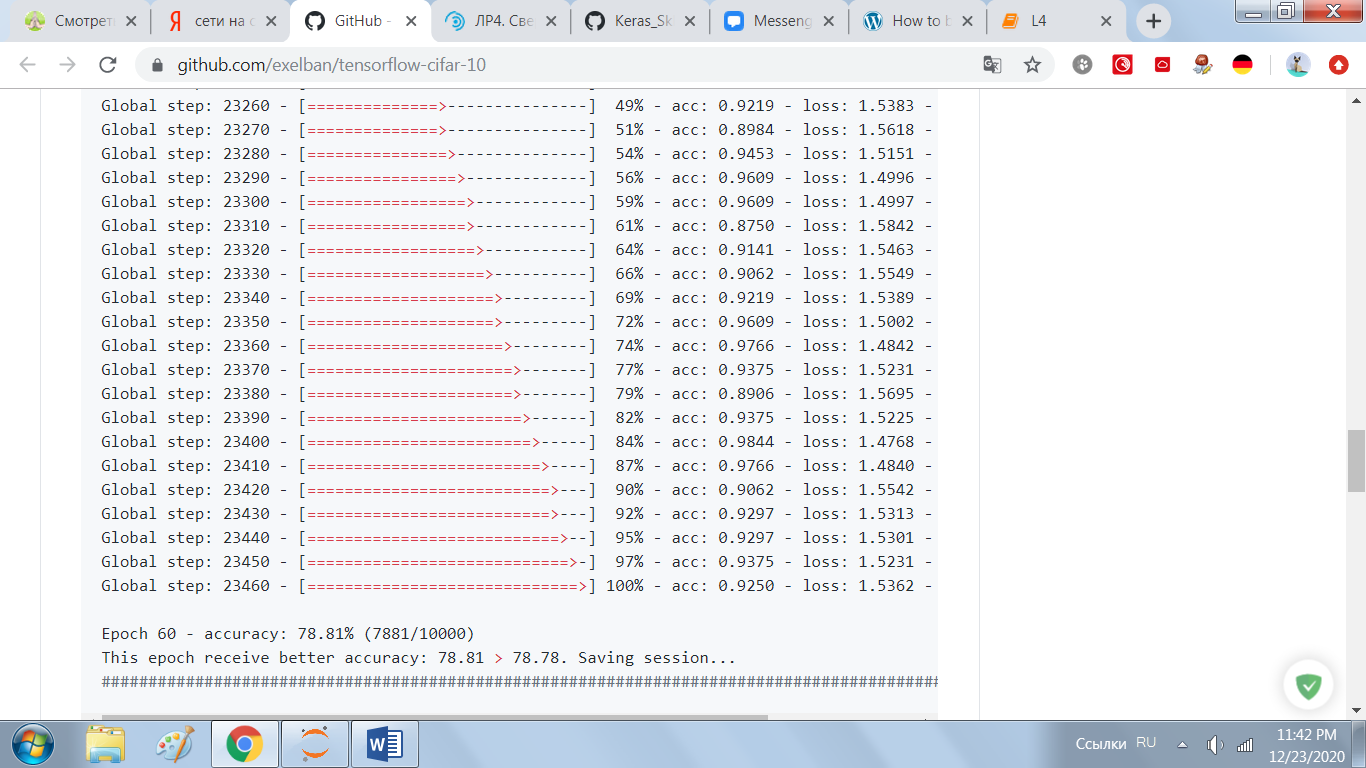
<https://www.machinelearningmastery.ru/object-recognition-convolutional-neural-networks-keras-deep-learning-library/>



<https://ru.sciencewal.com/17772-cifar-10-image-classification-in-tensorflow-5b501f7dc77c-86>



<https://github.com/exelban/tensorflow-cifar-10>

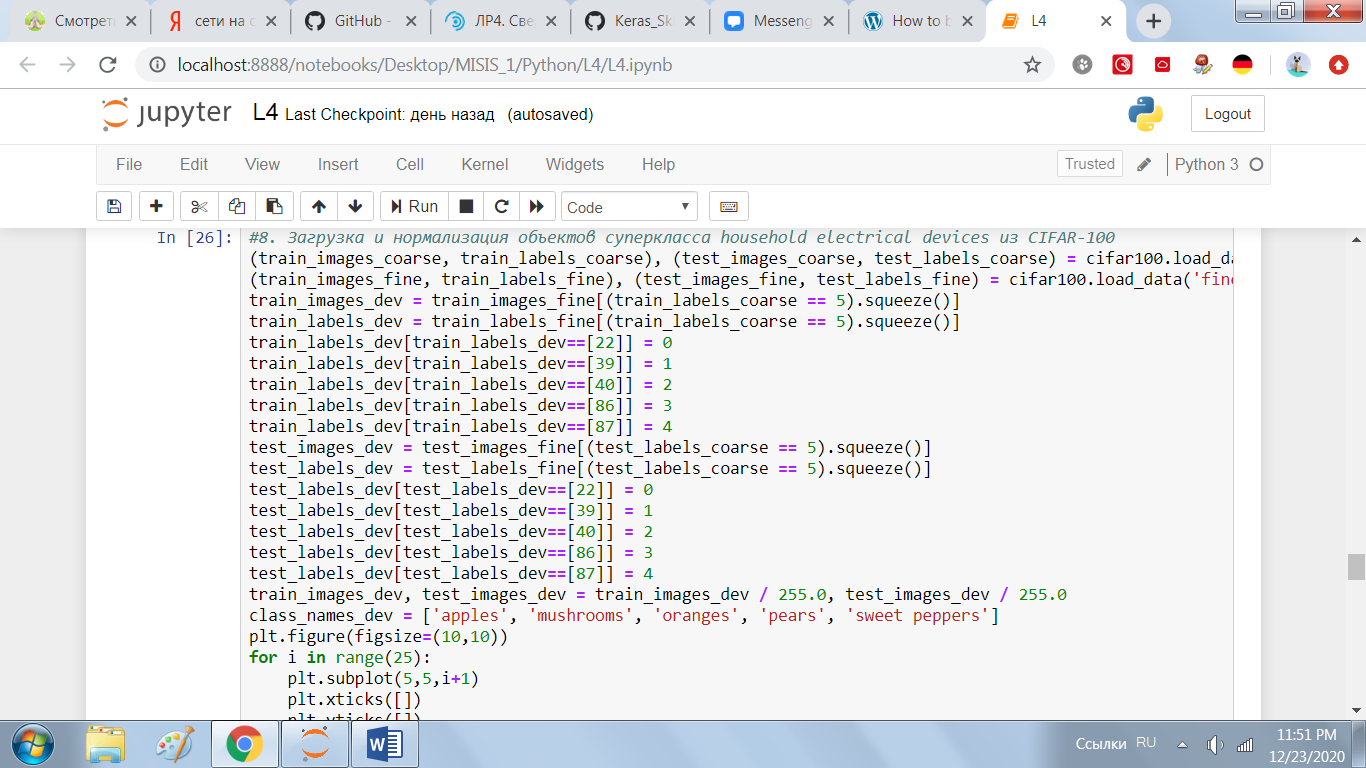


Исходя из найденных результатов можно сделать вывод, что при использовании всего датасета с 10 и большим количество эпох возможно дойти до среднего уровня распознавания, но т.к. мой компьютер позволяет использовать не больше 1000 фото в БД и не больше 2-х эпох, то адекватно сравнить результаты нельзя.

4. Используя ту же CNN и полученный набор весов провести дообучить CNN на классах из одного суперкласса ([по вариантам](https://lms.misis.ru/courses/8822/files/1718147/download?verifier=96mggTk3BjRPuNBSXsMKDrnNYAV1kBsxTzpo66uI&wrap=1)) эталонного датасета CIFAR-100. Задача классификации решается на классах CIFAR-10 + классы из одного суперкласса CIFAR-100.

Для дообучения необходимо извлечь из CIFAR100 необходимый суперкласс. Нет простых способов это сделать.

Я нашёл большой и сложный код для этого:

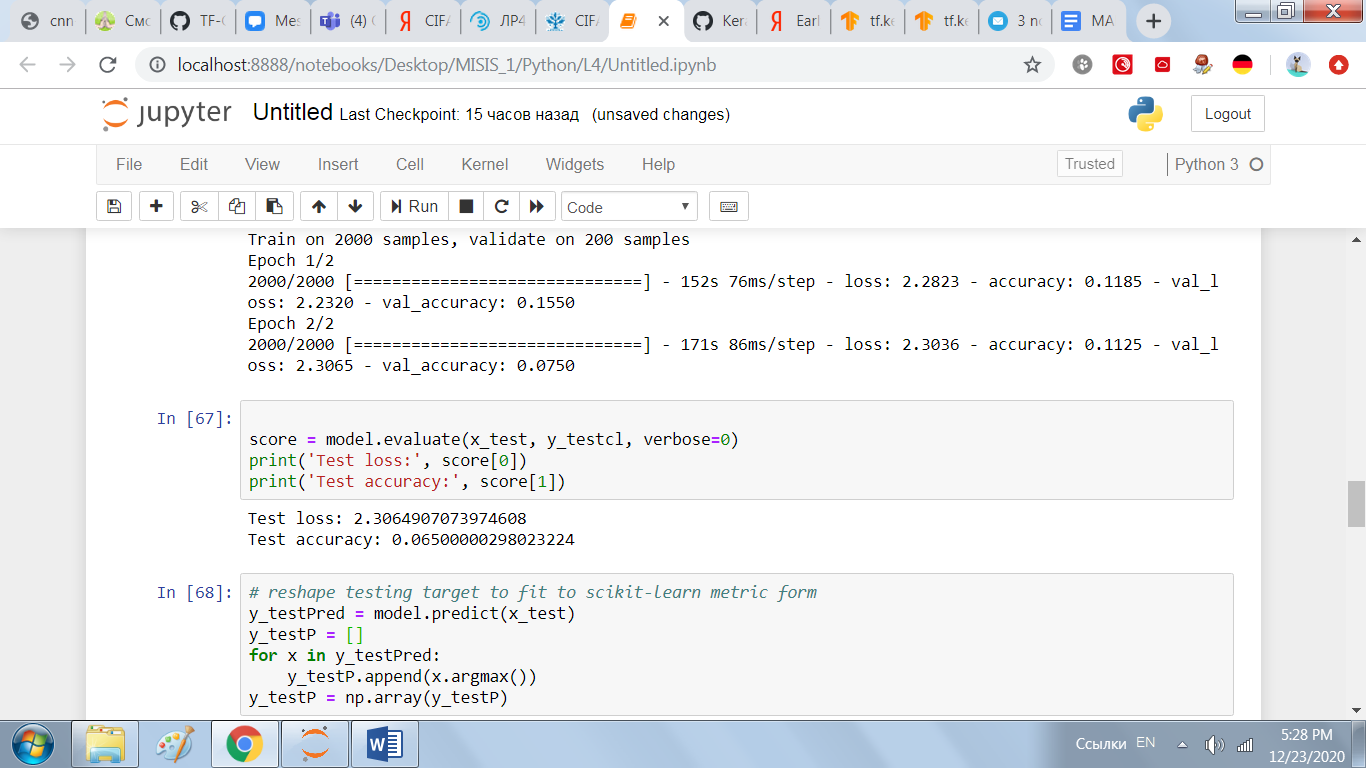


Затем необходимо объединить датасеты, новый и старый.

Но т.к. мой компьютер не позволяет использовать весь датасет, пришлось снизить количество фото.

Т.к. количество рапознаваемых классов увеличилось, то необходимо изменить архитектуру CNN, чтобы на выходе число нейронов было равно числу классов.

Также есть методы для дообучения с заморозкой первых слоёв, но т.к. это слишком сложно, я делаю повторное обучение с новым датасетом.



5. Провести сравнительный анализ результатов (см. раздел "Результаты") с полученными результатами из п.3

Исходя из результатов анализа, можно сделать вывод, что при увеличении количества классов необходимо изменять количество нейронов в слоях для большей точности, что демонстрирует моя дообученная исходная сеть.