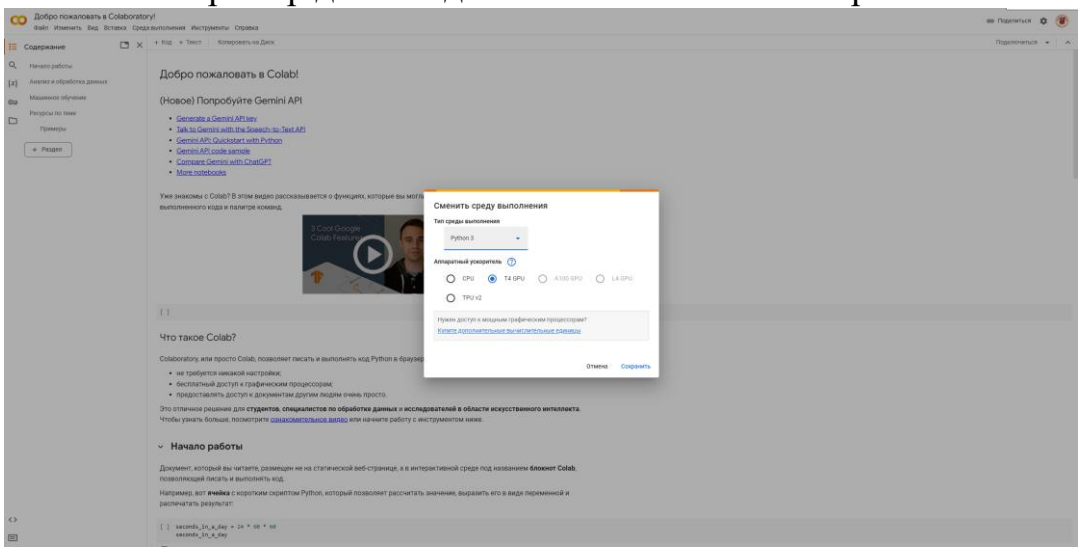
Тестовое задание

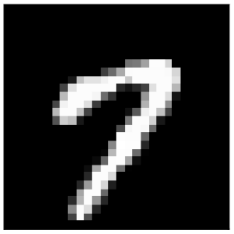
1. Прочитать статью BadNets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain;
2. Python фреймворком в данном задании будет выступать PyTorch, на котором в python-ноутбуке реализована простая сверточная нейронная сеть, описанная в статье выше в таблице 1 на странице 5. В качестве среды разработки можно использовать Google Colab, т.к. в нем можно бесплатно производить обучение с использованием графических ускорителей (GPU). Не забудьте, что зачастую в Google Colab аппаратный ускоритель в среде выполнения по умолчанию выбран как центральный процессор (CPU). Для выбора GPU перейдите на панель слева сверху в “Среда выполнения”→ “Сменить среду выполнения”→ “Аппаратный ускоритель” и выбрать T4 GPU или TPUv2. Пример данных действий показан на изображении ниже;



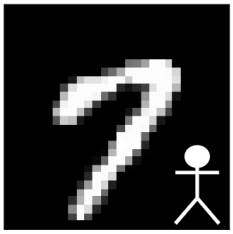
3. В качестве набора данных для обучения реализованной сверточной нейронной сети использовать датасет черно-белых изображений рукописных цифр (от 0 до 9) MNIST (Данный набор данных можно загрузить с помощью описанных выше фрейморков для глубокого обучения) напрямую в среду выполнения.
4. Попробуйте реализовать backdoor-атаку описанную в статье в главе 4. Case Study: MNIST Digit Recognition Attack. В качестве триггера, внедряемого в изображение, предлагаем использовать представленные в статье. Возможно, Вам захочется придумать свой собственный триггер (пример изображен на рисунке ниже). Подумайте, каким образом можно сделать используемый триггер более незаметным? Для выбора целевой метки можно воспользоваться механизмом, представленным в статье или, выбрать свою собственную, из 10 представленных классов (помните, что

целевая метка не должна совпадать с истинной меткой для зараженных изображений в наборе данных). В качестве модели сверточной нейронной сети используйте модель, реализованную в python-ноутбуке, а также зафиксированы следующие гиперпараметры:

a. Оптимизатор – SGD, с шагом градиентного спуска 0.01
b. Размер батча – 32 изображения
c. Количество эпох обучения – 10



Оригинальное изображение



Изображение с внедренным триггером (белый человечек)

5. Сравните полученные результаты с результатами, представленными в работе выше все в той же главе. Получилось ли у Вас воспроизвести результаты? Сильно ли деградировало качество обученной модели с внедренной backdoor-атакой и сопоставимо ли оно с результатами авторов статьи? Как думаете, успешность такой backdoor-атаки сильно зависит от внешнего вида внедряемого триггера в изображение? Как много данных Вам пришлось заразить в исходном наборе данных для обучения MNIST (1%, 2%, 10% …)?