

# Финальный проект — Object Detection

## Руководители проекта:

- Илья Захаркин (Samsung AI Center – Moscow) | tg: @izakharkin
- Юрий Яровиков (AIRI, МФТИ) | tg: @yu\_rovikov | github: yu\_rovikov

## Цель работы:

Написание полного пайплайна обучения и измерения метрик

## Основные решаемые задачи в работе:

- 1) Анализ данных: выбор актуальных данных и предобработка
- 2) Обучение модели и расчет метрик для измерения качества

## Основные этапы:

- 1) Выбор актуального датасета.

В сегодняшних реалиях важным элементом для защиты собственного здоровья от новой коронавирусной инфекции стала медицинская маска. Ношение нового аксессуара повседневной жизни уменьшает риск заражения и распространения данной инфекции (Подробные рекомендации по COVID-19 можно найти на сайте Минздрава России <https://minzdrav.gov.ru/news/2021/11/19/17832-minzdrav-vypustil-obnovlennuyu-versiyu-metodrekomendatsiy-po-covid-19>) . Что же касается требований к юридическим лицам, то для них предусмотрены штрафы за несоблюдение санитарных норм их сотрудниками в соответствии с КоАП ([https://base.garant.ru/12125267/5c09fde3c438bfa4e1dbc0d234d7aa47/#block\\_20611](https://base.garant.ru/12125267/5c09fde3c438bfa4e1dbc0d234d7aa47/#block_20611)).

В таких условиях возникает задача о соблюдении норм и минимизации штрафов, вплоть до их отсутствия, для различных предприятий. Ведь лучше предупредить последствия, чем получить дополнительные издержки и, возможно, репутационные риски. Предлагается вариант для предотвращения таких ситуаций в условиях финального проекта первой части курса «Deep Learning» — написание полного пайплайна обучения модели на размеченных данных «Mask Dataset» (MakeML <https://makeml.app/datasets/mask>, kaggle <https://www.kaggle.com/andrewmvd/face-mask-detection>). Датасет размечен для детекции на 3 класса - «правильно надетая маска на лицо человека», «человек без маски» и «неправильно надетая маска на лицо человека», содержит картинки в формате png и разметку в xml-файлах.

- 2) Предобработка данных.

- 2.1) Данные загружаются из Google Colab на Google Drive из kaggle.

- 2.2) Разбиение файлов на тренировочную и валидационную выборку.

- 2.3) Обработка двух форматов файлов для создания разметки и принадлежности к классу каждого объекта.

- 3) Используемая модель

- 3.1) Была выбрана предобученная модель Faster R-CNN с бэкбоун из ResNet-50

- 3.2) Подготовлена функция для дообучения модели с возможностью сохранения весов

- 4) Метрики качества модели

- 4.1) Перед расчетом метрик нужно было сделать обработку получившихся результатов — выбрать только те боксы, которые отвечают объекту с максимальной уверенностью модели в метке класса. Использован метод nms из библиотеки torch с порогом равным максимальному значению пересечения боксов на тренировочной выборке, при котором боксы относятся к разным объектам — 0.71. Или иначе: максимальное значение для метрики IoU для пересекающихся боксов двух различных объектов из тренировочной выборки.

- 4.2) В связи с тем, что модель может не найти некоторые реальные или найти несуществующие объекты на картинке была реализована функция для по классового выравнивания количества этих значений.

4.3) Реализована функция для расчета метрик recall, precision, f1\_score. Считаем, что модель верно предсказала класс при условиях выше и уверенности на предсказании больше 0.5. Среднее значение метрик рассчитывается как средневзвешенное.

4.4) Реализована функция для расчета mean Average Precision по описанию метрики из источников. Считаем, что модель верно предсказала бокс для объекта с разметкой при IoU этих значений больше 0.71. При таком условии можно точно утверждать, что это боксы относятся к одному и тому же объекту. Возможно при реализации могут присутствовать ошибки

5) Единое место запуска и настройки последовательности выполнения кода  
Нужно для того, чтобы было удобно запускать и редактировать готовый код.

### **Полученные результаты:**

Далее будут результаты расчетов для используемых метрик, построены графики mean Average Precision для каждого класса объектов для используемой модели с 5, 10 и 15 эпохами обучения.

Содержимое тестовой выборки:

Значение для 1 класса - маска надета верно	Значение для 2 класса - маска надета неправильно	Значение для 3 класса - маска отсутствует
654	28	158

### Модель с 5 эпохами

Расчет метрики precision\_score:

Значение для 1 класса	Значение для 2 класса	Значение для 3 класса
0.5	0	0.5
Среднее значение: 0.4833		

Расчет метрики recall\_score:

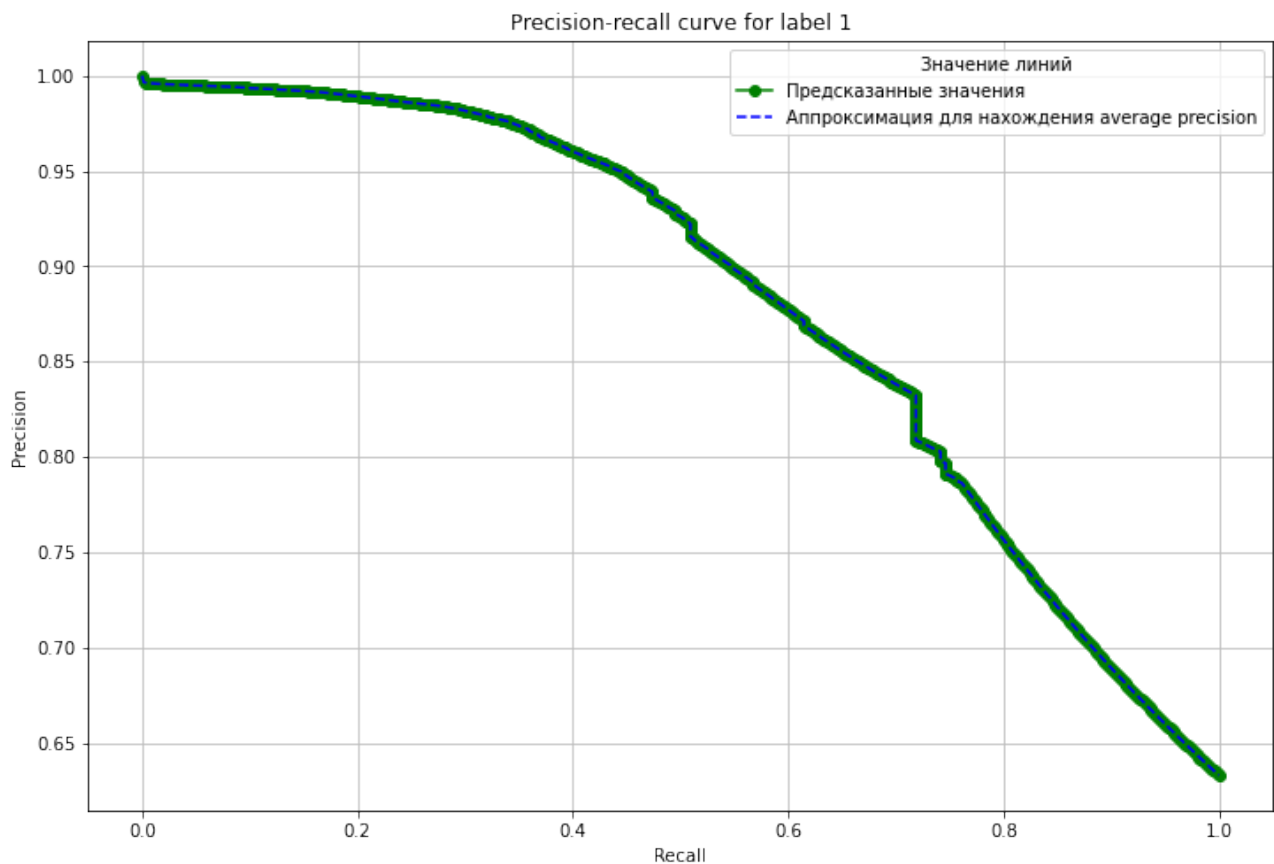
Значение для 1 класса	Значение для 2 класса	Значение для 3 класса
0.4633	0	0.4113
Среднее значение: 0.4381		

Расчет метрики f1\_score:

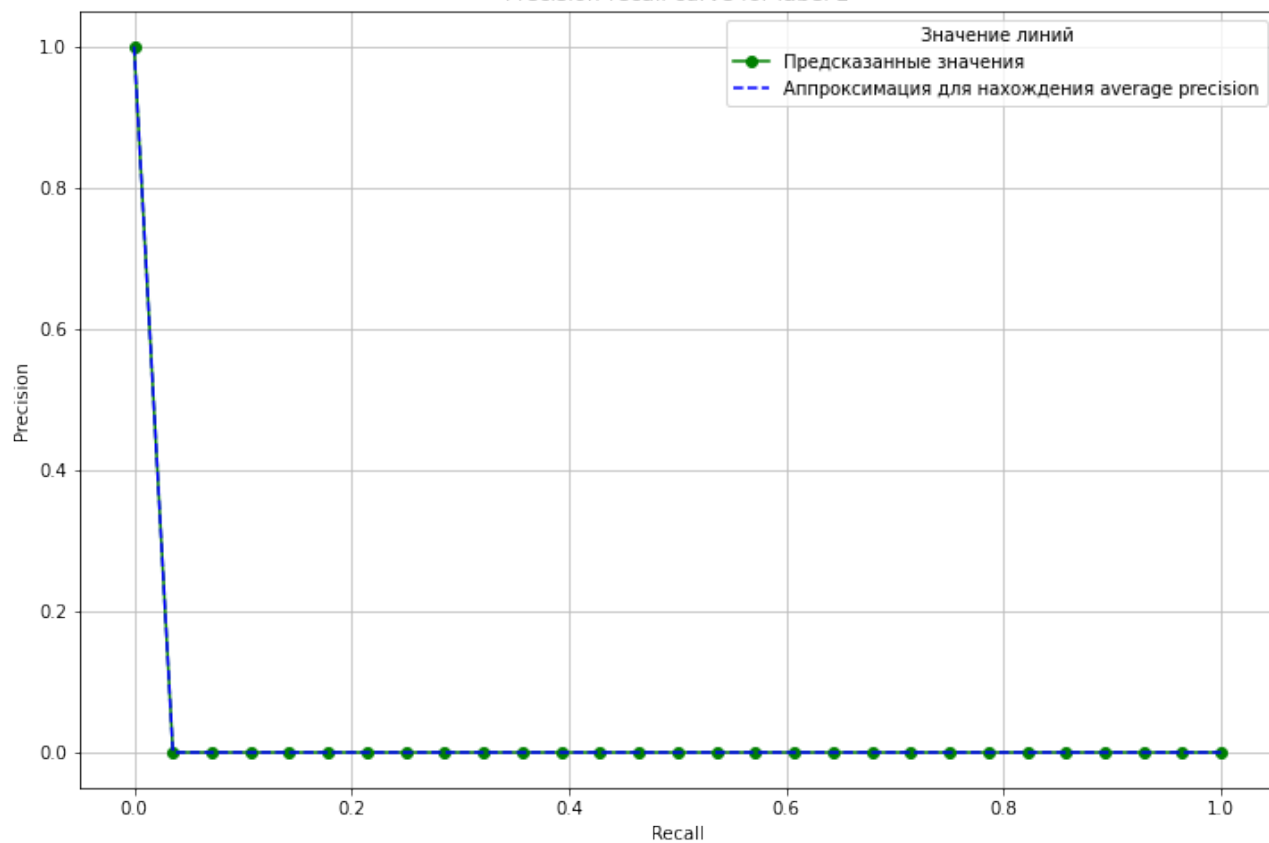
Значение для 1 класса	Значение для 2 класса	Значение для 3 класса
0.4809	0	0.4513
Среднее значение: 0.4594		

Расчет метрики mean Average Precision:

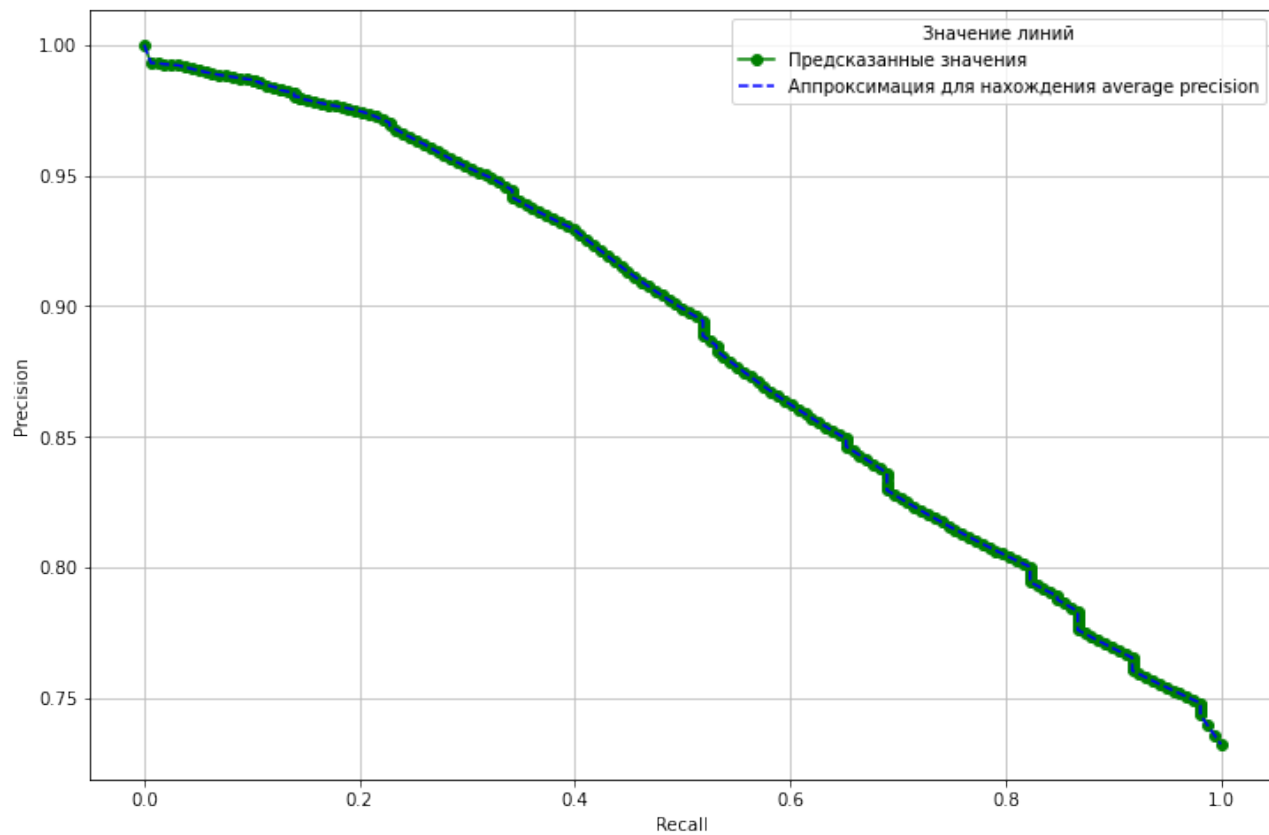
Значение для 1 класса	Значение для 2 класса	Значение для 3 класса
0.8819	0.0	0.8863
Среднее значение: 0.8534		



Precision-recall curve for label 2



Precision-recall curve for label 3



### Модель с 10 эпохами

Расчет метрики precision\_score:

Значение для 1 класса	Значение для 2 класса	Значение для 3 класса
0.5	0.5	0.5
Среднее значение: 0.5		

Расчет метрики recall\_score:

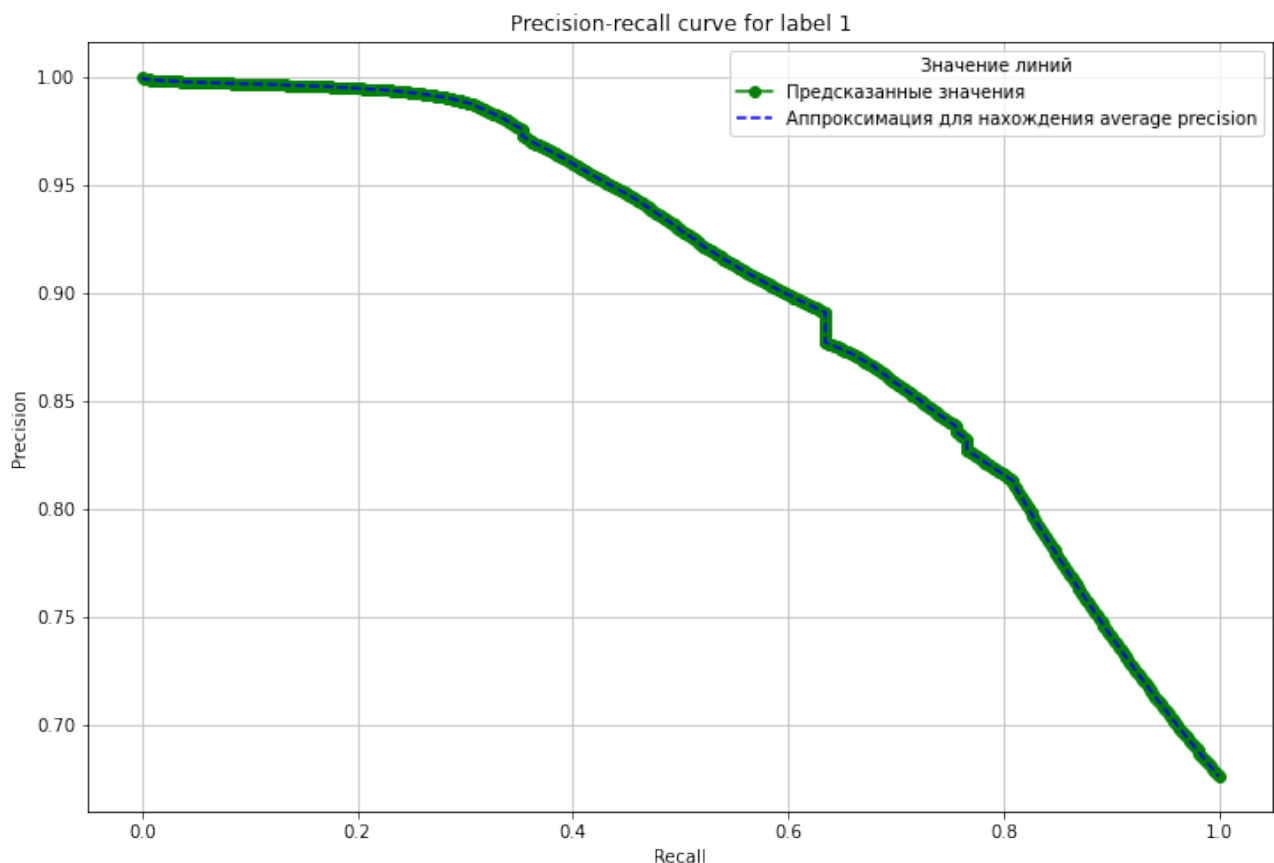
Значение для 1 класса	Значение для 2 класса	Значение для 3 класса
0.4564	0.0178	0.4113
Среднее значение: 0.4333		

Расчет метрики f1\_score:

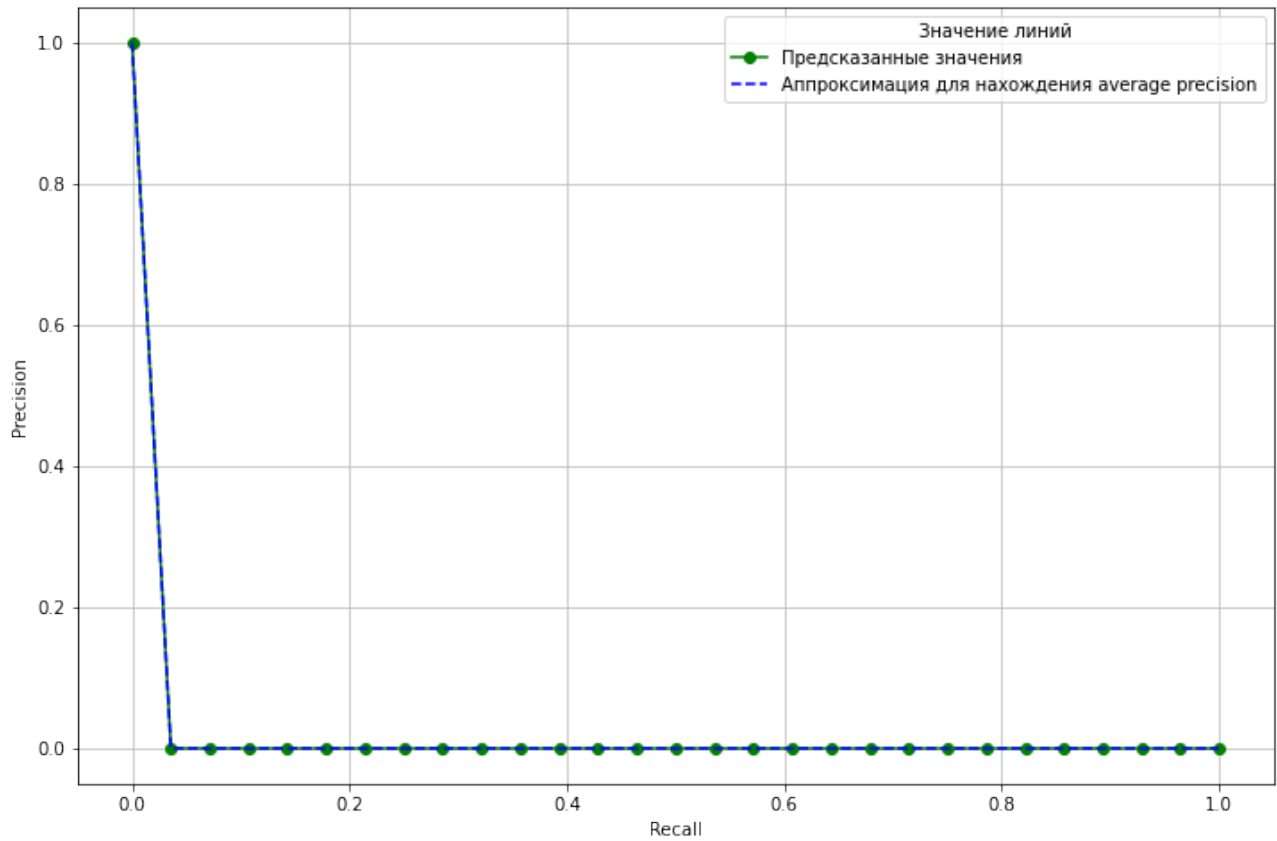
Значение для 1 класса	Значение для 2 класса	Значение для 3 класса
0.4772	0.0344	0.4512
Среднее значение: 0.4576		

Расчет метрики mean Average Precision:

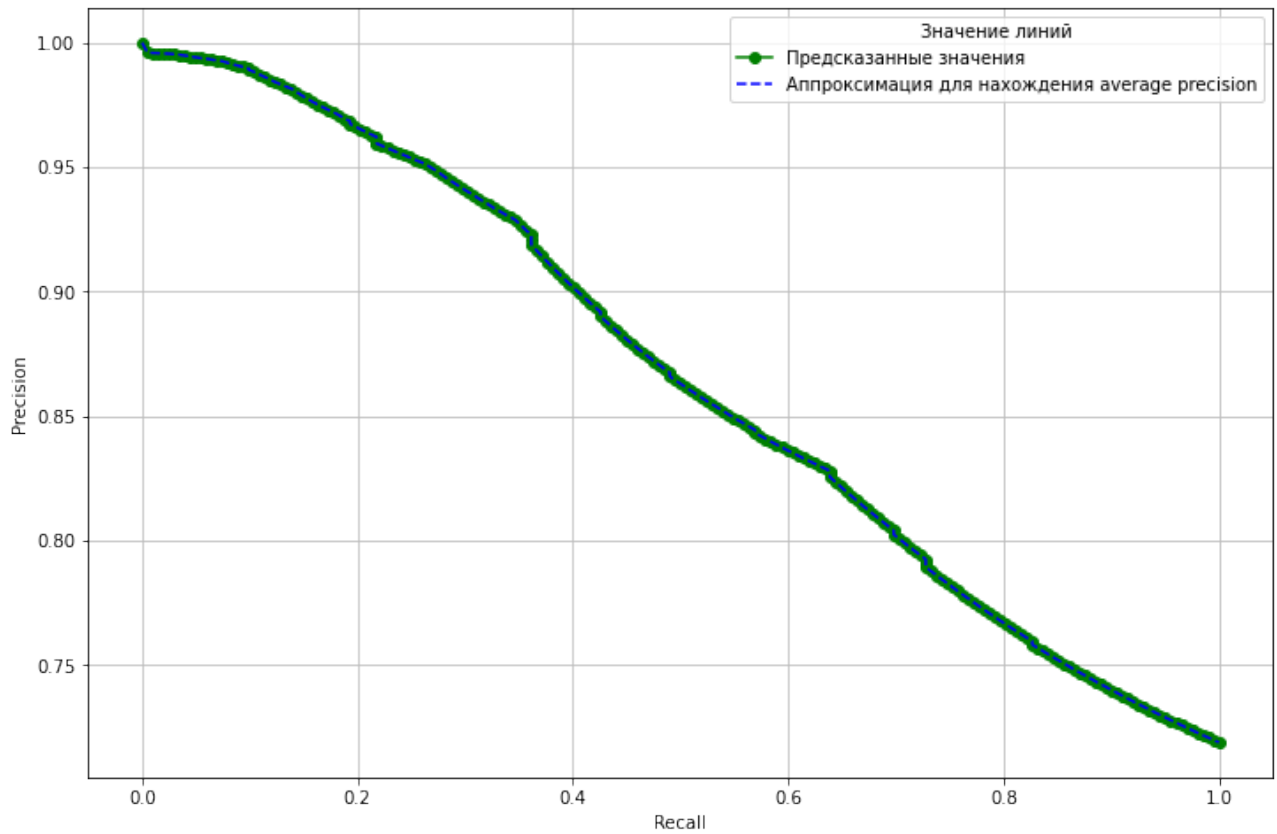
Значение для 1 класса	Значение для 2 класса	Значение для 3 класса
0.9018	0.0	0.8659
Среднее значение: 0.8650		



Precision-recall curve for label 2



Precision-recall curve for label 3



### Модель с 15 эпохами

Расчет метрики precision\_score:

Значение для 1 класса	Значение для 2 класса	Значение для 3 класса
0.5	0.5	0.5
Среднее значение: 0.5		

Расчет метрики recall\_score:

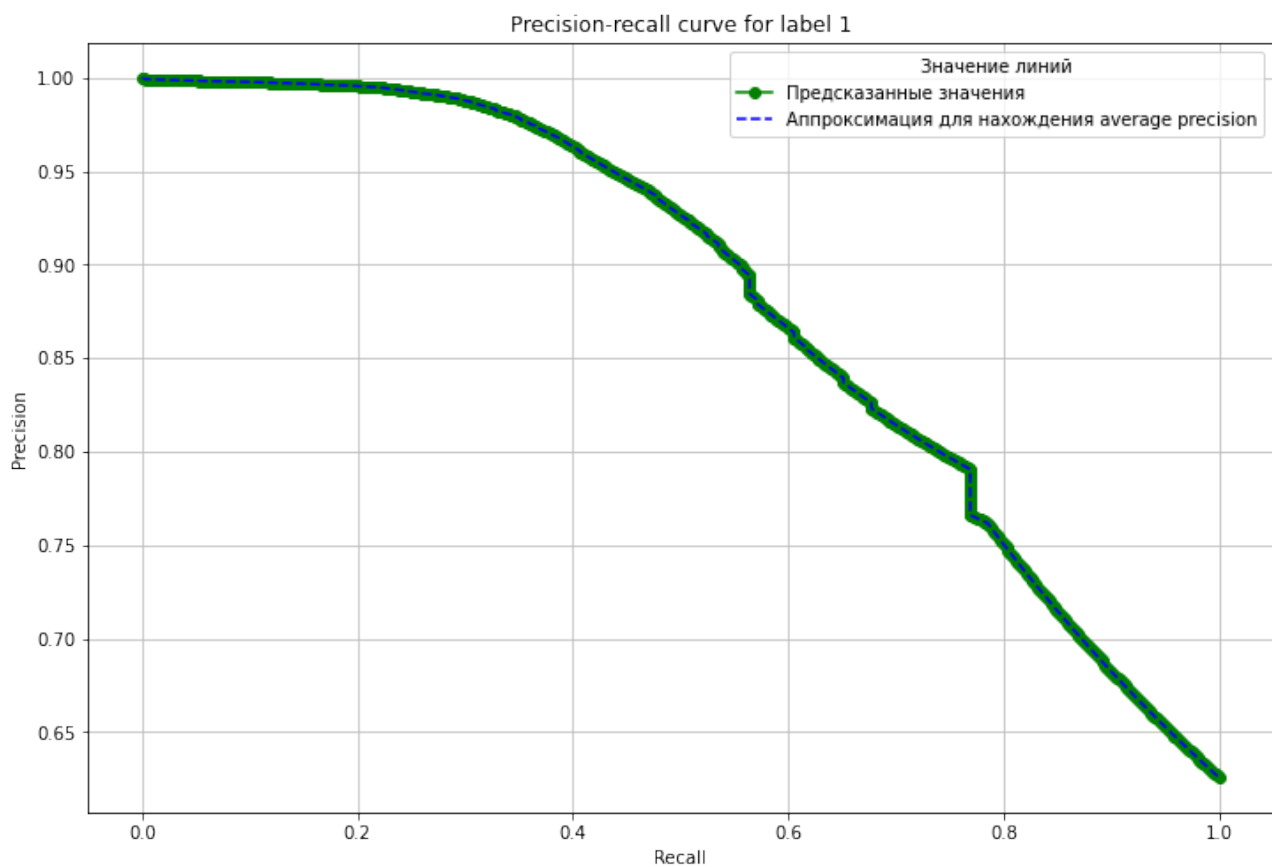
Значение для 1 класса	Значение для 2 класса	Значение для 3 класса
0.4594	0.0357	0.4082
Среднее значение: 0.4357		

Расчет метрики f1\_score:

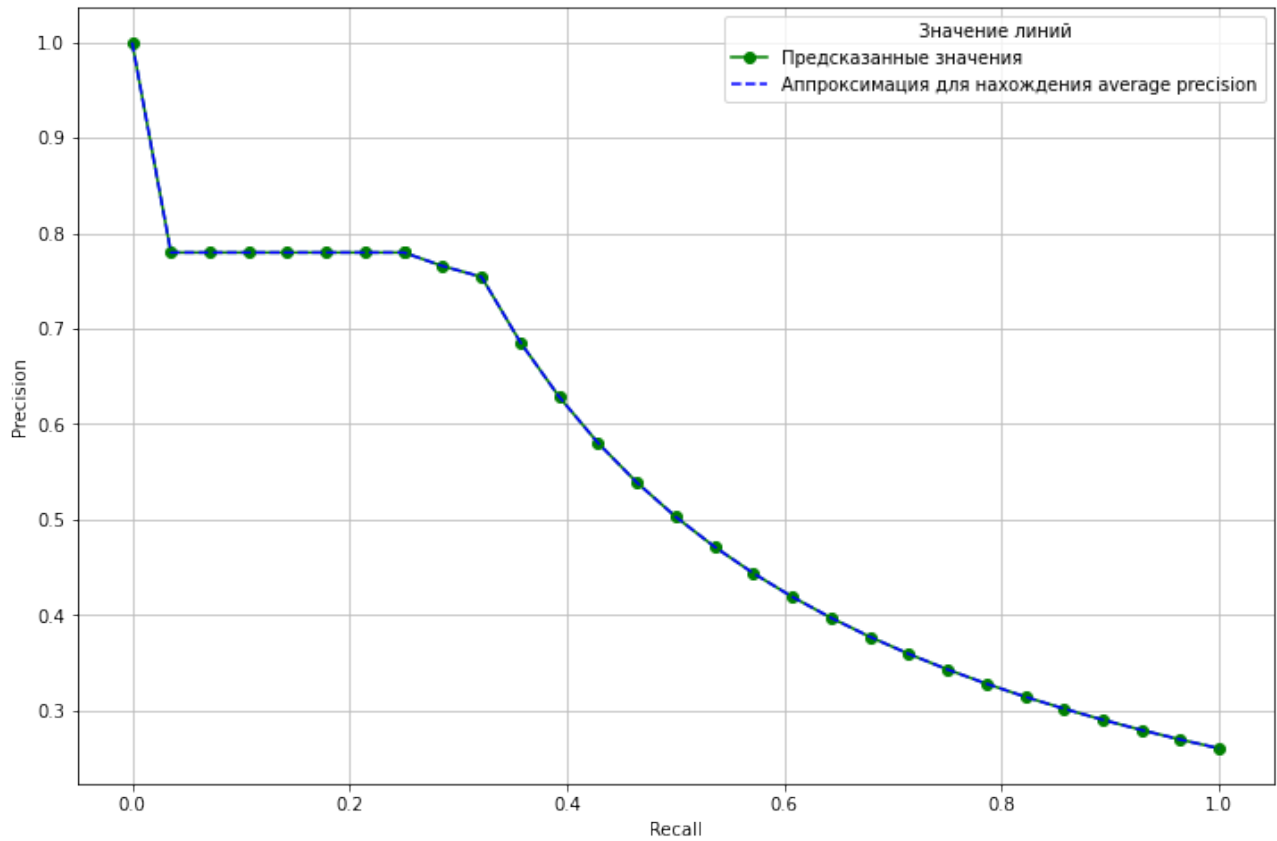
Значение для 1 класса	Значение для 2 класса	Значение для 3 класса
0.4788	0.0666	0.4494
Среднее значение: 0.4596		

Расчет метрики mean Average Precision:

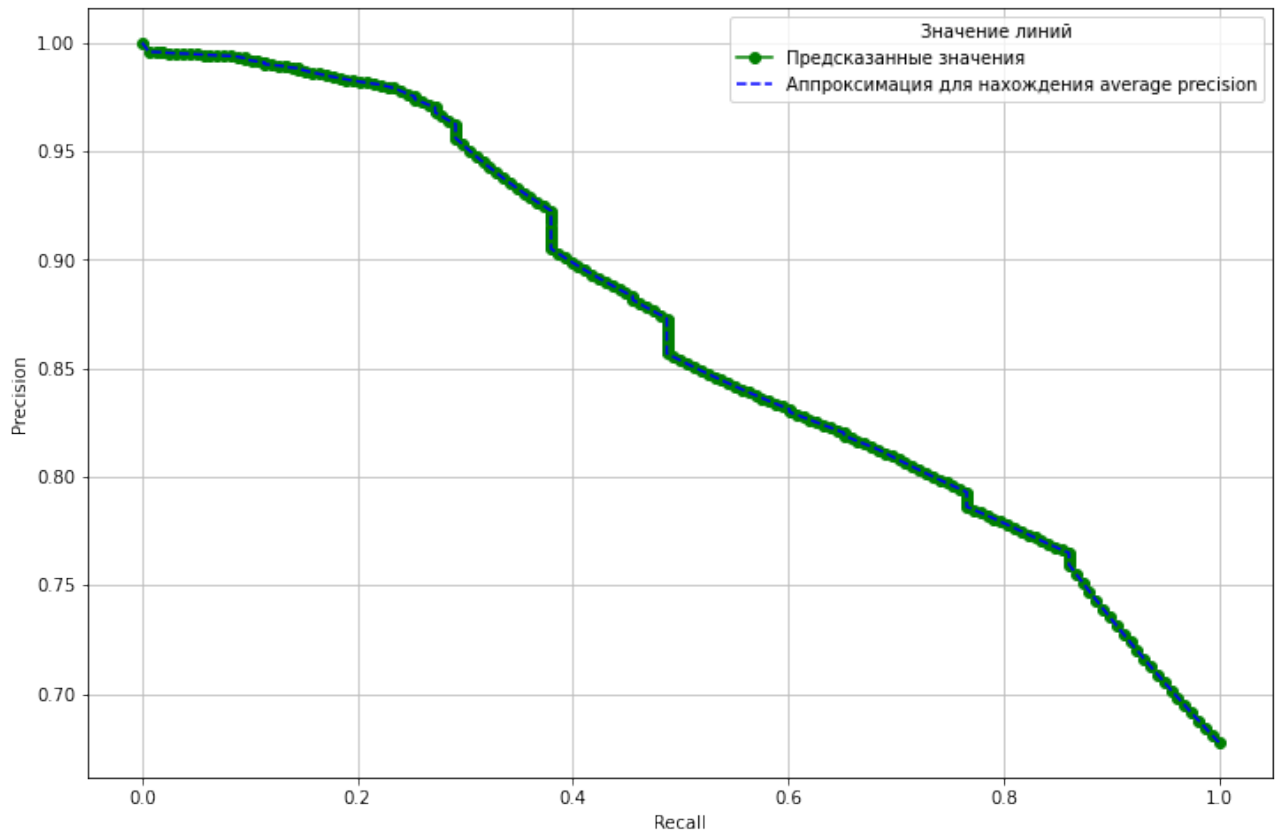
Значение для 1 класса	Значение для 2 класса	Значение для 3 класса
0.8802	0.5273	0.8680
Среднее значение: 0.8662		



Precision-recall curve for label 2



Precision-recall curve for label 3





### **Краткий итог:**

- Данные в классах не сбалансированны. Это обусловлено реальностью
- С ростом количества эпох для обучения модели качество растет незначительно, но начинают корректно проявляться данные из самого сложного класса - «маска надета некорректно», причем на остальных классах качество немного снижается, но все еще довольно хорошее
- Логгирование наше все!

### **Примечания:**

- техническая реализация находится в файле hw.ipynb

### **Что еще стоит попробовать:**

- другую модель для обучения
- встроить модель с помощью OpenCV для детекции объектов в видео потоке
- собрать и использовать докер-образ модели с обработкой входных сигналов