МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего

образования

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Отчет по заданию №3

по дисциплине

«ГЛУБИННОЕ ОБУЧЕНИЕ»

**«Задание №3 – Обучение нейросетевых моделей классификации изображений»**

Выполнил

Студент гр. 932226

\_\_\_\_\_\_\_\_\_\_\_Савенкова М. М. «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г

Проверил

\_\_\_\_\_\_\_\_ к.т.н. Аксёнов С.В.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г.

Томск – 2023

Оглавление

[1 Индивидуальное задание 3](#_Toc154406459)

[2 Цель работы 4](#_Toc154406460)

[3 Листинг программ 5](#_Toc154406461)

[4 Входные данные 6](#_Toc154406462)

[5 Построение моделей 8](#_Toc154406463)

[5.1 Бинарная классификация 8](#_Toc154406464)

[5.1.1 Построенные модели 8](#_Toc154406465)

[5.2 Бинарная классификация с аугментацией данных 16](#_Toc154406466)

[5.2.1 Модель №1 16](#_Toc154406467)

[5.2.2 Модель №2 17](#_Toc154406468)

[5.2.3 Модель №3 19](#_Toc154406469)

[5.3 Многоклассовая классификация 21](#_Toc154406470)

[5.3.1 Заключительные полносвязные слои 21](#_Toc154406471)

[5.3.2 Fine-tuning 25](#_Toc154406472)

[6 Выводы 27](#_Toc154406473)

# 1 Индивидуальное задание

Задачи – построить нейросетевые модели – бинарный и многоклассовый классификатор для набора данных plantdisease, расположенного по ссылке <https://www.kaggle.com/datasets/emmarex/plantdisease>.

Классы:

* Класс 0: Tomato\_YellowLeaf\_Curl\_virus,
* Класс 1: Tomato\_healthy,
* Класс 2: Tomato\_Septoria\_Leaf\_spot

1. Бинарный классификатор

Сначала построить бинарный классификатор для классов 0 и 1.

Сохранить лучшую архитектуру нейронной сети – бинарного классификатора. Выполнить загрузку и предварительную обработку данных из наборов. Разделить каждую выборку на обучающую, тестовую и валидационную. Произвести обучение набора нейросетевых архитектур, отличающихся разным набором параметров:

* количество карт признаков,
* распределение слоёв,
* процедур оптимизации.

Выполнить эксперименты с прореживанием (Dropout). Привести кривые изменения функции потерь (Loss) и верности (Accuracy) во время обучения на обучающем и валидационном наборах. Выполнить аугментацию данных и провести повторное обучение на новом расширенном наборе.

1. Многоклассовый классификатор

Создать нейронную сеть – многоклассовый классификатор с использованием сверточной основы, обученной ранее на ImageNet (ResNet, VGG, Inception, Xception и т.д.).

Сохранить лучшие модели. Выполнить дообучение модели в двух вариантах: заключительные полносвязные слои и fine-tuning. Привести кривые изменения функции потерь (Loss) и верности (Accuracy) во время обучения на обучающем и валидационном наборах из трех классов.

Привести результаты обучения моделей и сделать выводы.

# 2 Цель работы

Цель работы – получить навыки создания и обучения нейронных сетей для классификации изображений, а также переноса обучения.

# 3 Листинг программ

Листинги программ представлены в отдельном файле: CNNs\_(Savenkova\_932226).ipynb.

# 4 Входные данные

Набор данных изображений листьев больных растений и соответствующих меток. Входные данные представлены в виде изображений, рассортированных по директориям, соответствующим классам (рисунок 1). Каждое изображение имеет размеры 256x256 и является трёхканальным.

|  |
| --- |
| C:\MARINA\STUDY\Магистратура\семестр 3\глубинное обучение\задание 3\0.png |
| C:\MARINA\STUDY\Магистратура\семестр 3\глубинное обучение\задание 3\1.png |
| C:\MARINA\STUDY\Магистратура\семестр 3\глубинное обучение\задание 3\3.png |
| Рисунок 1 — Класс 0, Класс 1, Класс 2 |

Количество изображений в каждом классе представлено на рисунке 2.

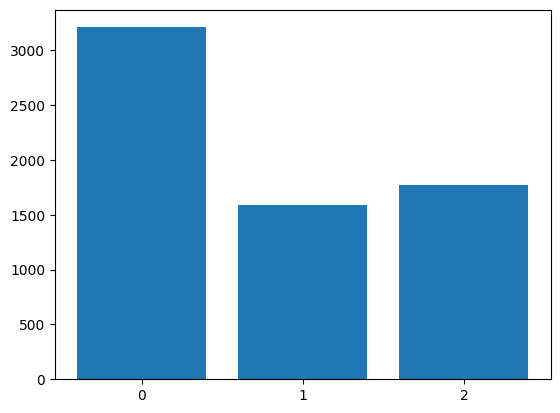


Рисунок 2 — Количество изображений в каждом классе

Поскольку имеется некоторый дисбаланс классов, при оценке моделей добавим метрику weighted accuracy. Взвешенные метрики, такие как взвешенная точность (weighted accuracy), используются в случаях, когда классы в задаче имеют различное количество экземпляров и/или когда ошибки классификации имеют разную важность.

Например, в задачах многоклассовой классификации, если один класс содержит гораздо больше экземпляров, чем другие классы, простая точность может быть сильно смещена в пользу этого класса. В таких случаях применяется взвешенная точность, которая учитывает соотношение количества экземпляров в каждом классе.

# 5 Построение моделей

В рамках данной лабораторной с использованием библиотеки Keras было построено три модели для задачи бинарной классификации и две для задачи мультиклассовой классификации.

5.1 Бинарная классификация

Бинарные классификаторы в данной лабораторной были построены для классификации изображений с листьями, зараженными вирусом скручивания желтых листьев помидоров, и здоровых томатных листьев.

5.1.1 Построенные модели

5.1.1.1 Модель №1

**Параметры архитектуры и обучения**

Основные параметры в слоях Conv2D и MaxPool2D:

* filters – задает количество сверточных фильтров (каналов) в этом слое. Во время обучения сети фильтры обновляются таким образом, чтобы свести потери к минимуму. Таким образом, в ходе обучения фильтры научатся распознавать определенные элементы, такие как края и текстуры. Представляет собой целое число, размерность выходного пространства.
* kernel\_size – размер ядра каждого фильтра.
* pool\_size – размер окна, в котором выбирается максимальное значение.

Создадим модель с одним сверточным слоем и одним полносвязным.

Параметры свёрточного слоя:

* размеры входного тензора – (256,256,3);
* количество фильтров – 3;
* размер ядра каждого фильтра – (3,3);
* функция активации – relu;
* размер окна, в котором выбирается максимальное значение – (2,2).

Параметры полносвязного слоя:

* количество нейронов – 1
* функция активации – sigmoid

Параметры обучения:

* функция потерь – бинарная кросс-энтропия;
* оптимизатор – adam;
* метрики – BinaryAccuracy, AUC, Precision, Recall;
* взвешенные метрики – Accuracy;
* callback-функции – EarlyStopping (завершение обучения, если улучшение точности не наступит после заданного числа эпох, в нашем случае эпох 7), ModelCheckpoint (сохранение лучшей модели).

**Полученные метрики**

Метрики для модели показаны на рисунке 3.

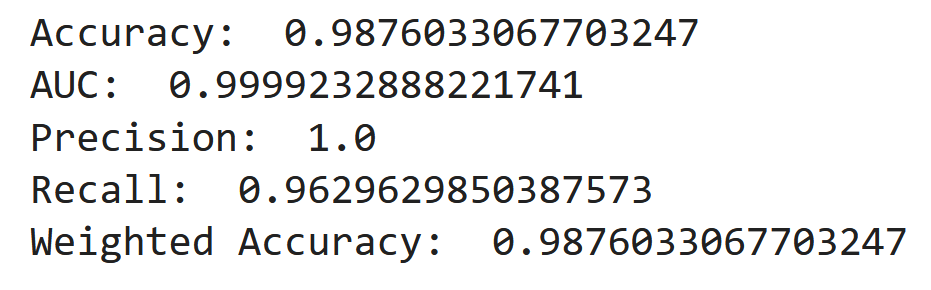


Рисунок 3 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 4).

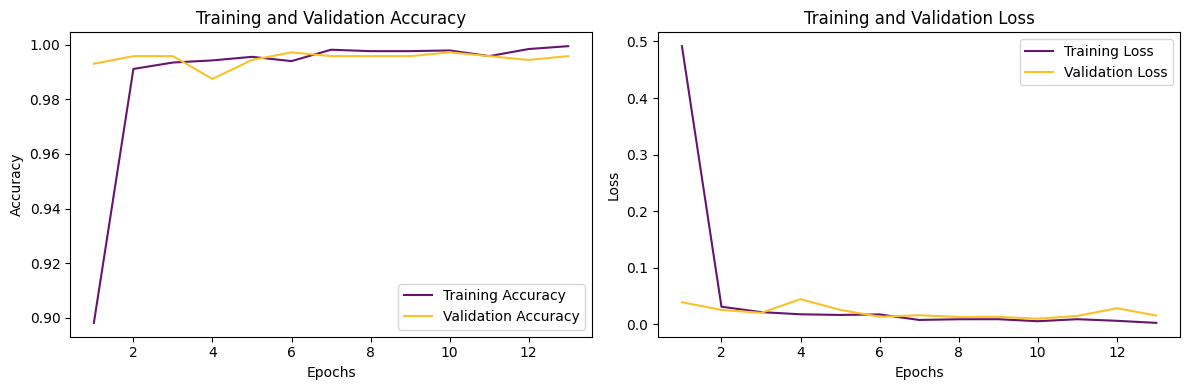


Рисунок 4 — График обучения

График оценки предсказаний (рисунок 5).

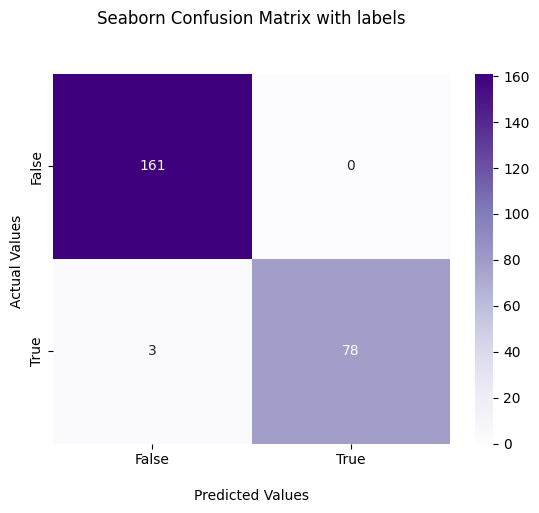


Рисунок 5 — Оценка предсказаний

5.1.1.2 Модель №2

**Параметры архитектуры и обучения**

Создадим модель с двумя сверточными слоями и двумя полносвязными.

Параметры свёрточных слоев:

* размеры входного тензора – (256,256,3);
* количество фильтров – 16 и 32;
* размер ядра каждого фильтра – (3,3);
* функции активации – relu;
* размер окна, в котором выбирается максимальное значение – (2,2).

Параметры полносвязных слоев:

* количество нейронов – 128, 1;
* функции активации – relu, sigmoid.

Параметры обучения:

* функция потерь – бинарная кросс-энтропия;
* оптимизатор – RMSprop;
* метрики – BinaryAccuracy, AUC, Precision, Recall;
* взвешенные метрики – Accuracy;
* callback-функции – EarlyStopping (7 эпох), ModelCheckpoint.

**Полученные метрики**

Метрики для модели показаны на рисунке 6.

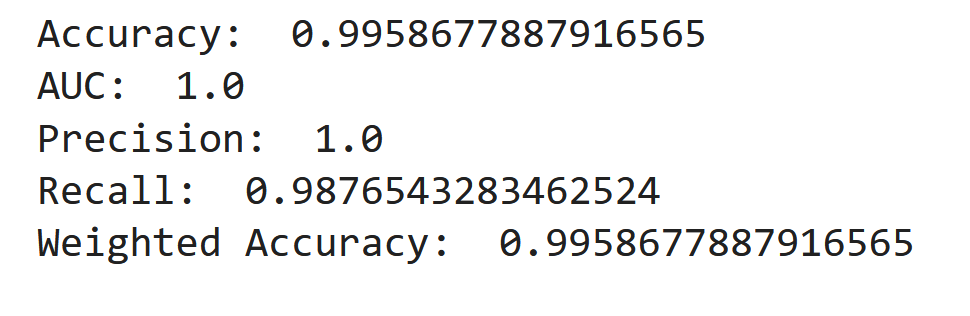


Рисунок 6 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 7).

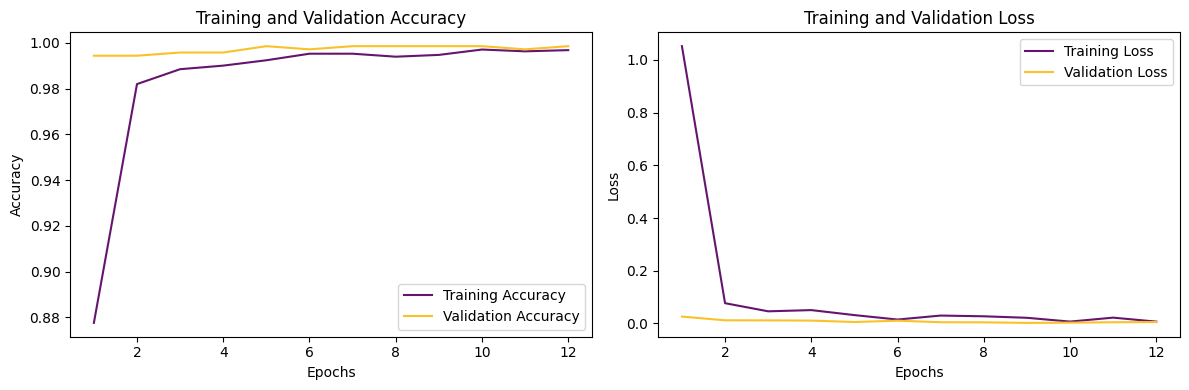


Рисунок 7 — График обучения

График оценки предсказаний (рисунок 8).

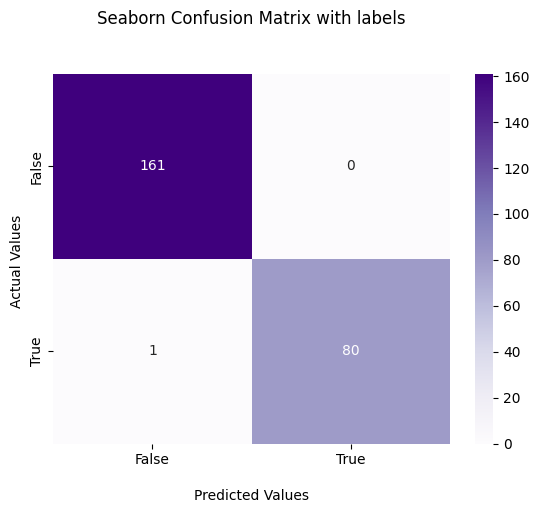


Рисунок 8 — Оценка предсказаний

5.1.1.3 Модель №3

**Параметры архитектуры и обучения**

Добавим в предыдущую модель два слоя Dropout - между сверточными и между полносвязными слоями.

Параметры свёрточных слоев:

* размеры входного тензора – (256,256,3);
* количество фильтров – 16 и 32;
* размер ядра каждого фильтра – (3,3);
* функции активации – relu;
* размер окна, в котором выбирается максимальное значение – (2,2).

Параметры полносвязных слоев:

* количество нейронов – 128, 1;
* функции активации – relu, sigmoid.

Параметры обучения:

* функция потерь – бинарная кросс-энтропия;
* оптимизатор – RMSprop;
* метрики – BinaryAccuracy, AUC, Precision, Recall;
* взвешенные метрики – Accuracy;
* callback-функции – EarlyStopping (7 эпох), ModelCheckpoint.

**Полученные метрики**

Метрики для модели показаны на рисунке

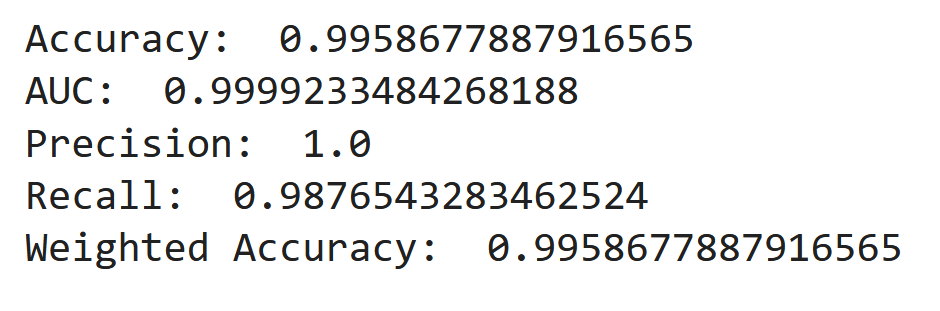


Рисунок 9 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 10).

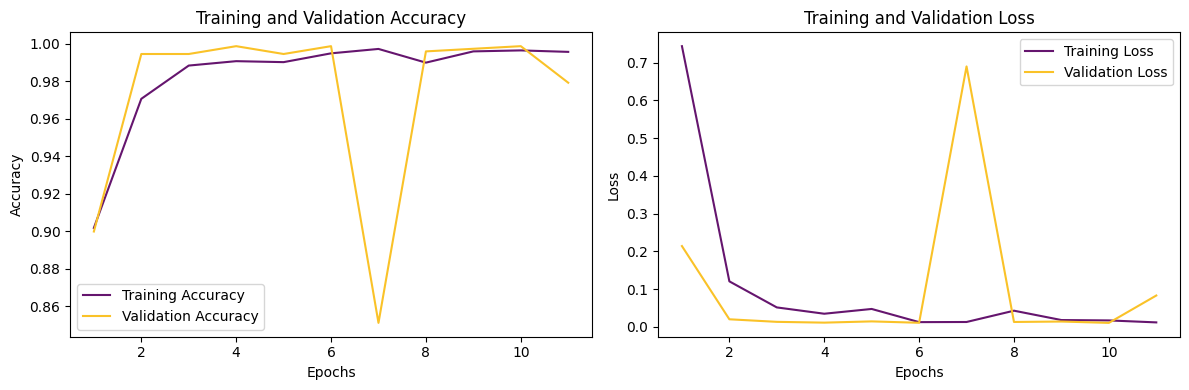


Рисунок 10 — График обучения

График оценки предсказаний (рисунок 11).

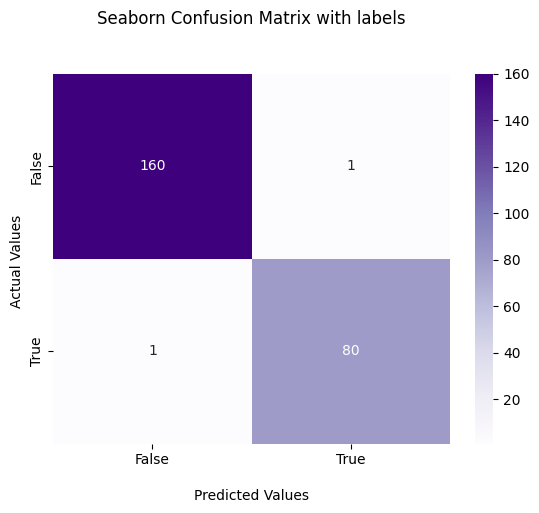


Рисунок 11 — Оценка предсказаний

5.2 Бинарная классификация с аугментацией данных

Для реализации аугментации данных в нейронные сети был добавлен ещё один слой data\_augmentation, который реализует следующие трансформации:

* отражение по горизонтали
* вращение на случайное значение (до 0.05 – доля от 2\*pi)
* меняем контрастность изображений (с параметром 0.23)
* изменяем размер (с параметром изменения 0.2)

5.2.1 Модель №1

**Полученные метрики**

Метрики для модели показаны на рисунке 12.

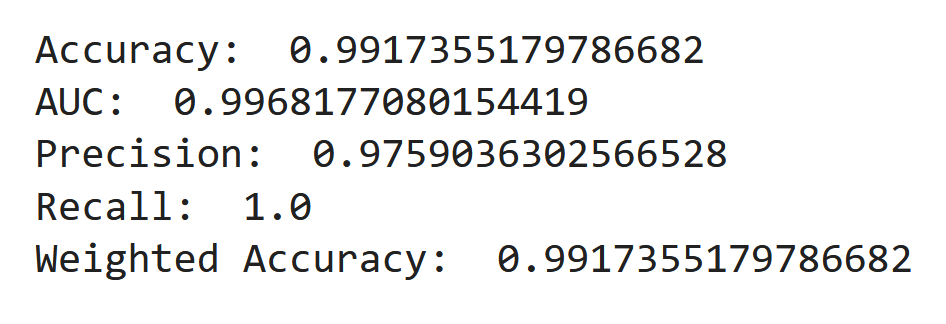


Рисунок 12 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 13).

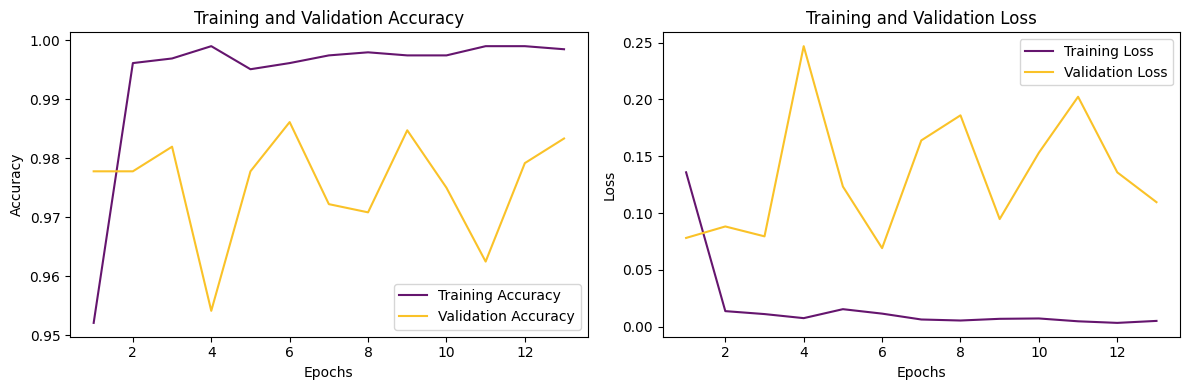


Рисунок 13 — График обучения

График оценки предсказаний (рисунок 14).

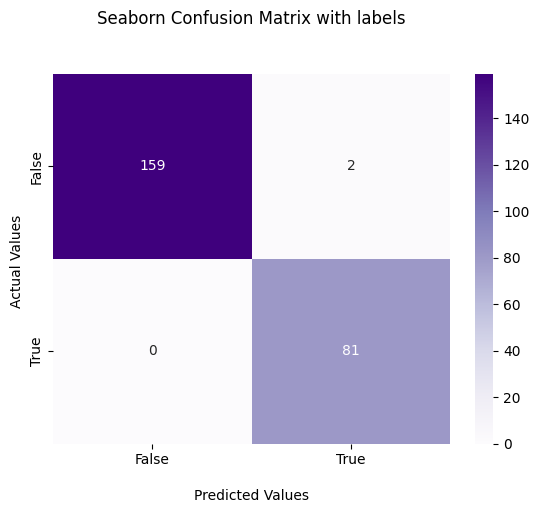


Рисунок 14 — Оценка предсказаний

5.2.2 Модель №2

**Полученные метрики**

Метрики для модели показаны на рисунке 15.

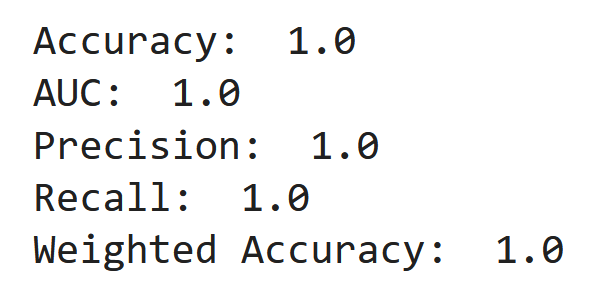


Рисунок 15 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 16).

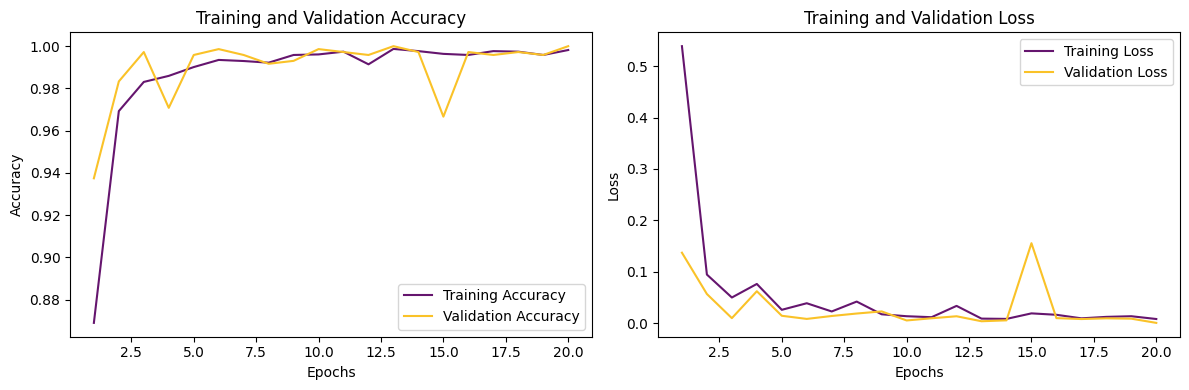


Рисунок 16 — График обучения

График оценки предсказаний (рисунок 17).

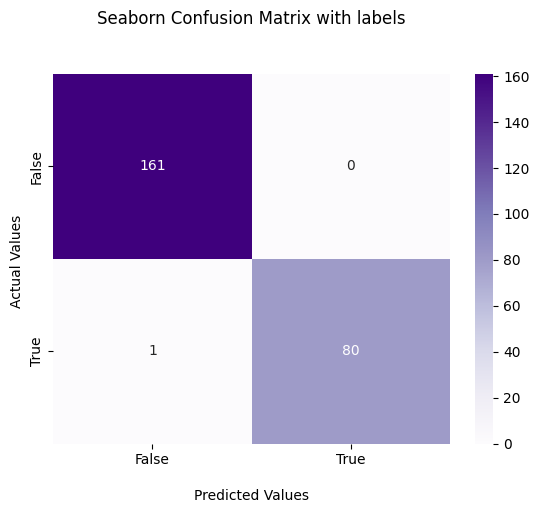


Рисунок 17 — Оценка предсказаний

5.2.3 Модель №3

**Полученные метрики**

Метрики для модели показаны на рисунке 18.

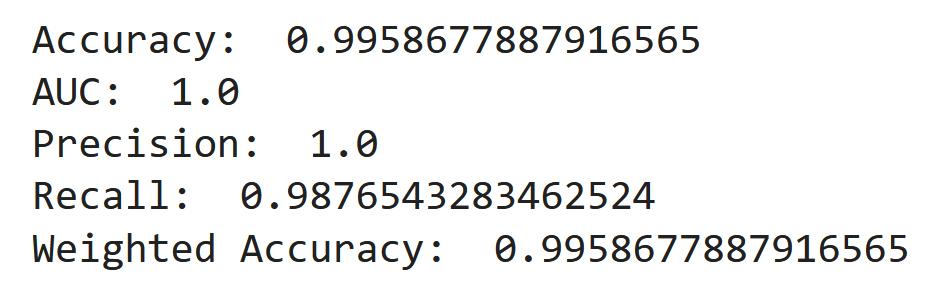


Рисунок 18 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 19).

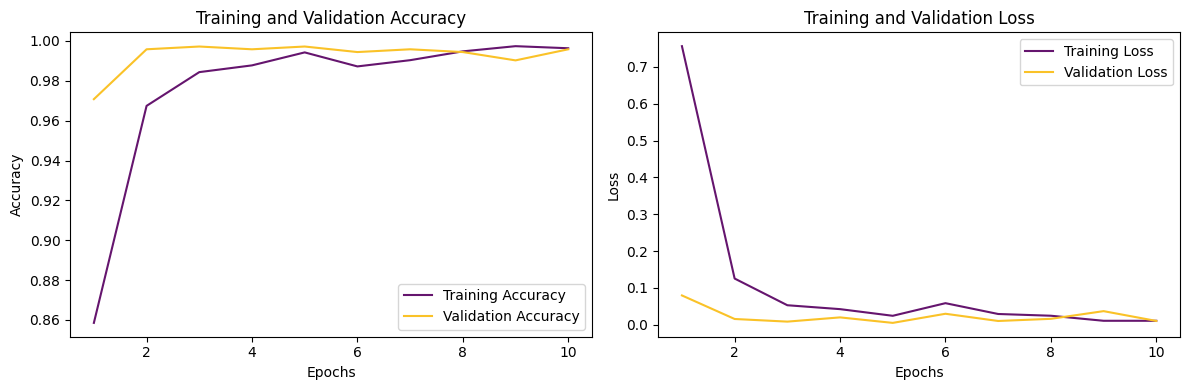


Рисунок 19 — График обучения

График оценки предсказаний (рисунок 20).

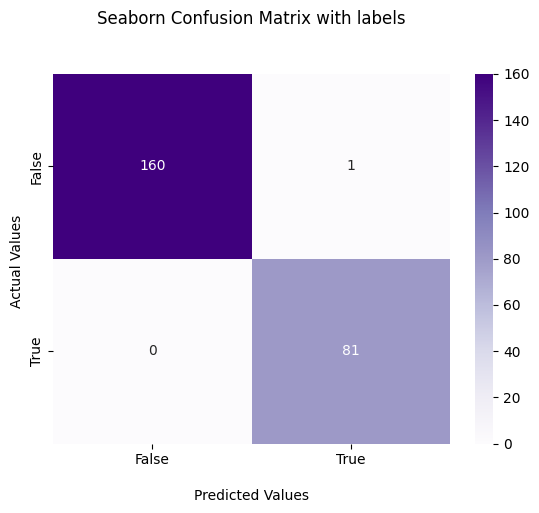


Рисунок 20 — Оценка предсказаний

5.3 Многоклассовая классификация

В качестве третьего рассматриваемого класса используются изоюражения томатных листьев с пятном Septoria.

Пятно Septoria начинается как маленькие коричневатые круги на нижней стороне нижних листьев. Пятно Septoria, также называемое пятнистостью Septoria, является очень распространенным заболеванием томатов, которое вызвано грибом (Septoria lycopersici).

Для реализации мультиклассового классификатора в качестве предобученной нейросети была выбрана ResNet50, обученная на наборе данных ImageNet. ImageNet — это крупный набор данных с аннотированными объектами. Цель создания ImageNet — развития алгоритмов компьютерного зрения. Это коллекция объектов-изображений, включающая около 1000 категорий изображений с примечаниями.

ResNet расшифровывается как «остаточная сеть» и представляет собой особый тип сверточной нейронной сети (CNN), представленный в статье 2015 года “Глубокое остаточное обучение для распознавания изображений” Хэ Каймина, Чжан Сянъюя, Жэнь Шаоцина и Сунь Цзяня.

5.3.1 Заключительные полносвязные слои

5.3.1.1 Модель №1

**Параметры архитектуры и обучения**

После слоя ResNet50 следуют слой Flatten и два полносвязных слоя, между которыми расположен слой прореживания Dropout с параметром 0.5.

Количество нейронов в полносвязных слоях – 256 и 3, функции активации – relu, softmax.

В качестве функции потерь была выбрана категориальная кросс-энтропия, в качестве оптимизатора – Adam.

**Полученные метрики**

Метрики для модели показаны на рисунке 21.

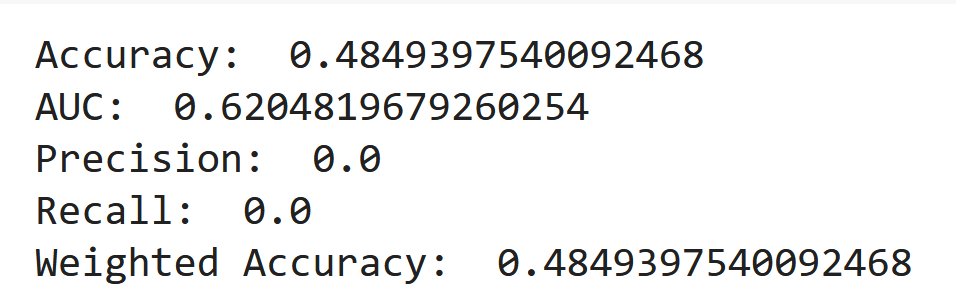


Рисунок 21 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 22).

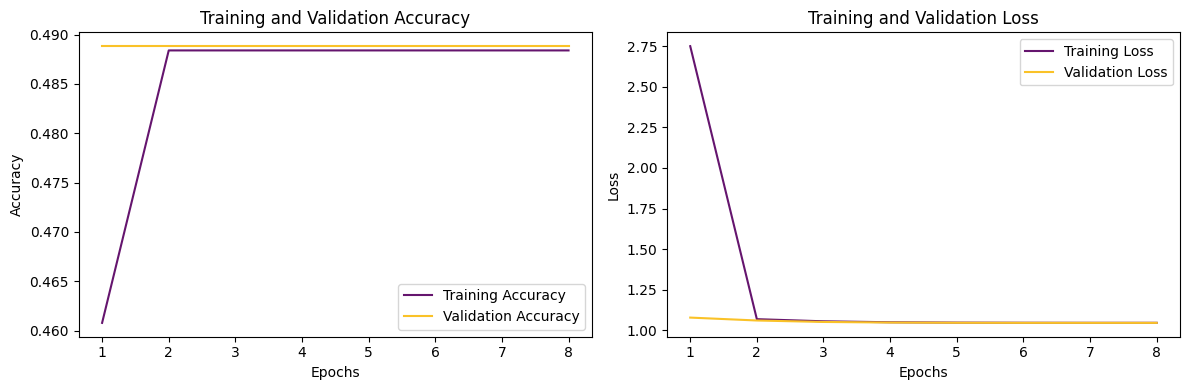


Рисунок 22 — График обучения

График оценки предсказаний (рисунок 23).

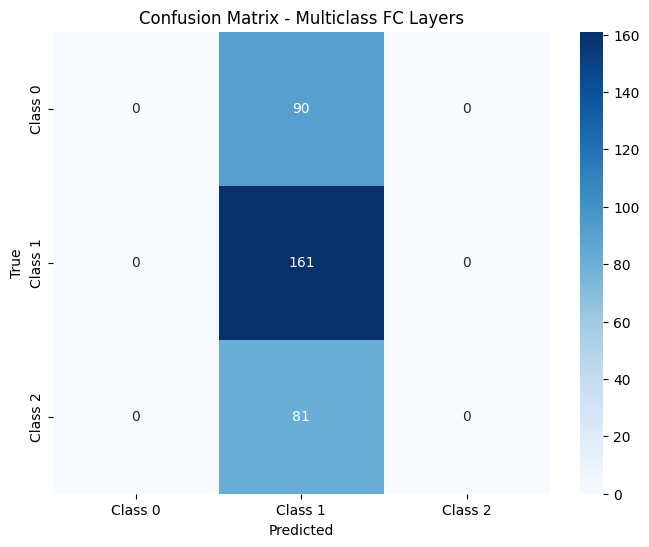


Рисунок 23 — Оценка предсказаний

5.3.1.2 Модель №2

**Параметры архитектуры и обучения**

Модель №2 имеет такую же архитектуру, как и предыдущая, но в оптимизатор Adam был добавлен параметр learning\_rate, равный 0.001 (коэффициенты обучения со значениями 0.1 и 0.01 не улучшили процесс обуечния).

**Полученные метрики**

Метрики для модели показаны на рисунке 24.

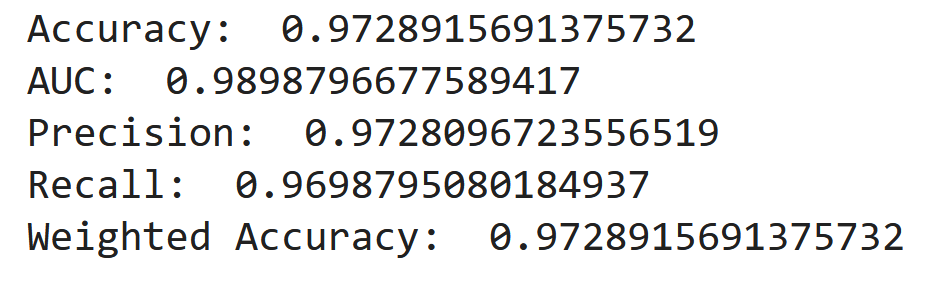


Рисунок 24 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 25).

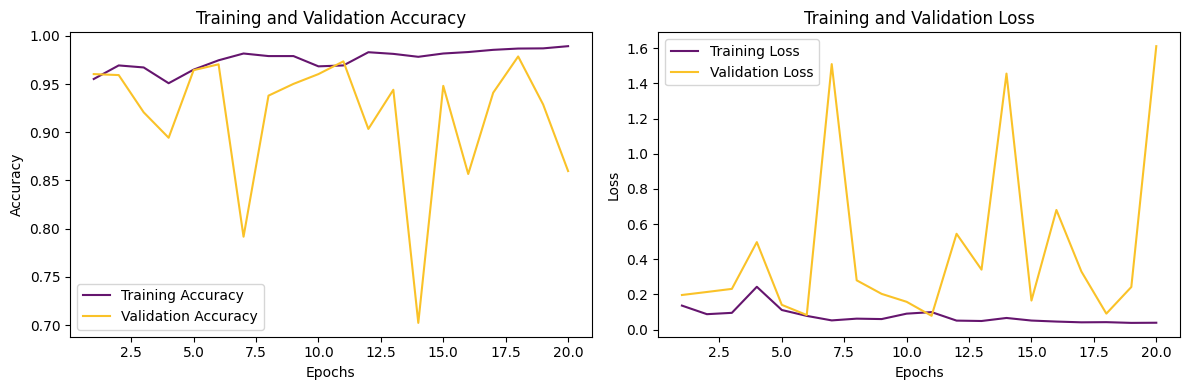


Рисунок 25 — График обучения

График оценки предсказаний (рисунок 26).

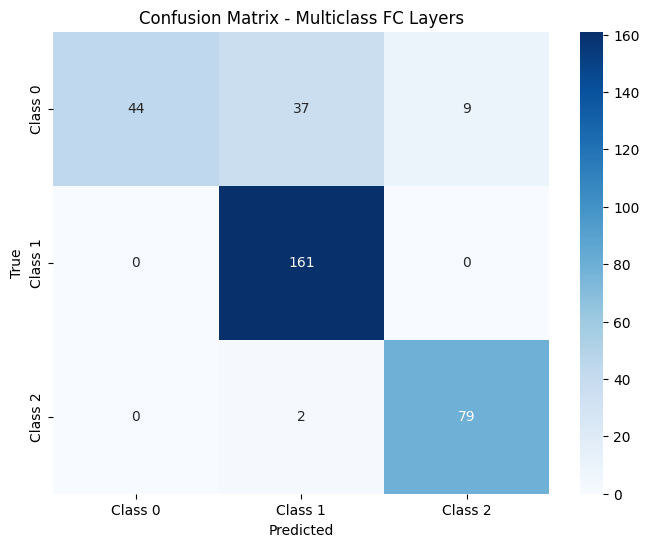


Рисунок 26 — Оценка предсказаний

5.3.2 Fine-tuning

Fine-Tuning — это способ улучшить предварительно обученную модель, которая уже имеет некоторые знания, путем небольших корректировок. Тонкая настройка помогает модели лучше работать над конкретной задачей, не обучая ее с самого начала.

Точная настройка на предварительно обученной модели включает в себя обучение некоторых слоев вместо замораживания всех весов.

**Параметры архитектуры и обучения**

Используем предварительно обученную ResNet для точной настройки, выполнив следующее:

* Заморозим первых 150 слоев и обучим только последние 15 слоев в ResNet.
* Добавим полносвязные слои с 256 и 3 нейронами, функциями активации relu и softmax соответственно.

В качестве функции потерь была выбрана категориальная кросс-энтропия, в качестве оптимизатора – Adam.

Также к уже имеющимся callback-функциям EarlyStopping и ModelCheckpoint была добавлена EarlyStoppingByAccuracy, которая останавливает обучение при достижении точности 0.95.

**Полученные метрики**

Метрики для модели показаны на рисунке 27.

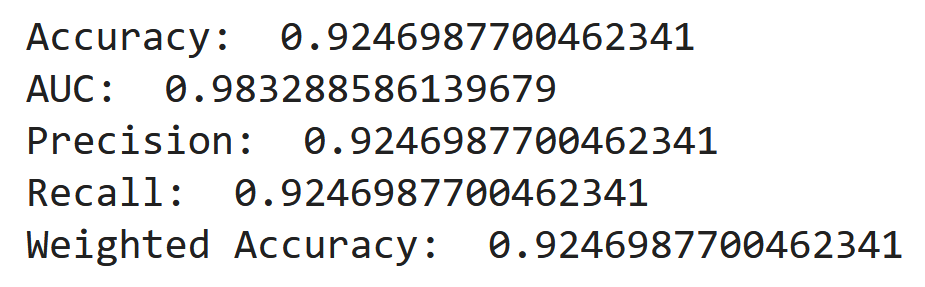


Рисунок 27 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 28).

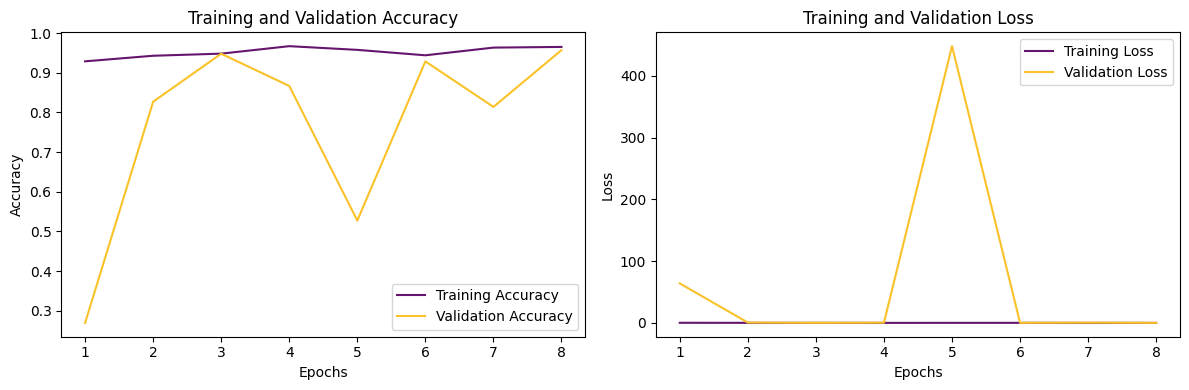


Рисунок 28 — График обучения

График оценки предсказаний (рисунок 29).

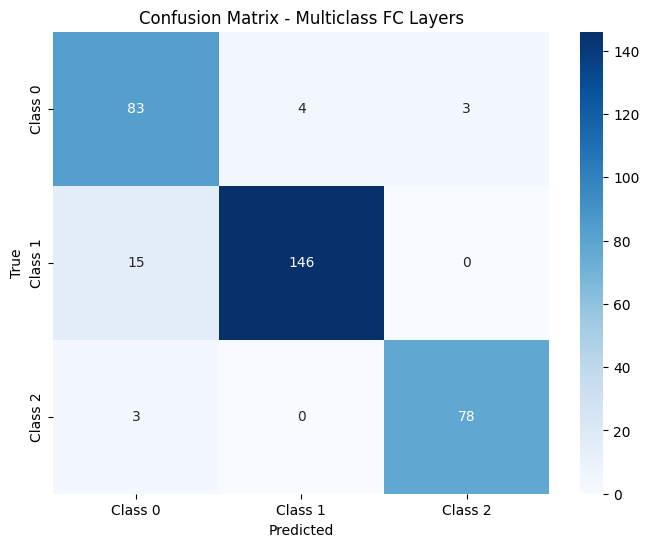


Рисунок 29 — Оценка предсказаний

# 6 Выводы

В задачах бинарной классификации лучшие результаты продемонстрировала модель №2 с двумя сверточными и двумя полносвязными слоями. Добавление слоя Dropout негативно сказалось на процессе обучения. Аугментация позволила добиться повышения точности и сделать процесс обучения более гладким для второй и третьей модели, тогда как первая, наиболее простая из трех, модель на аугментированных данных обучалась хуже.

В задаче мультиклассовой классификации:

А) В модели, где нейронная сеть ResNet не участвует в корректировке весов, удалось добиться улучшения результатов на тестовой выборке, задав коэффициент обучения в оптимизаторе Adam, в противном случае нейронная сеть не обучалась и работала как константный предсказатель. Изменение оптимизатора на RMSprop и параметра слоя Dropout на 0.2, 0.01 не оказало эффекта.

Б) В случае с fine-tuning, модель предсказывает с довольно высокой точностью. Наибольшее число ошибок было допущено при различении класса 0 и класса 1: 15 фотографий из тестовой выборки, принадлежащий классу 1, модель приняла за класс 0.