**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ**

**Ордена Трудового Красного Знамени**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Кафедра «Математическая кибернетика и информационные технологии»

ЛАБОРАТОРНАЯ РАБОТА №2

по дисциплине

НЕЙРОННЫЕ СЕТИ

на тему:

«Fine-tuning предобученной модели ResNeXt для классификации изображений»

Выполнил: студент группы

БВТ2201

Шамсутдинов Рустам

Москва, 2025 год

**Введение**

**Цель работы:** Ознакомиться с архитектурой ResNeXt, её отличительными особенностями и научиться проводить fine-tuning предобученной модели ResNeXt на новом датасете.

**Задачи:**

- Изучить теоретическую базу архитектуры ResNeXt и понять её отличие от классической ResNet.

- Ознакомиться с особенностями датасета CIFAR-100

- Загрузить предобученную модель ResNeXt

- Заменить последний полносвязный слой для адаптации модели к новому числу классов

- Провести fine-tuning модели на выбранном датасете

- Визуализировать динамику обучения (например, выводить значение потерь по эпохам)

- Выполнить оценку модели на тестовом наборе с построением отчёта по метрикам и матрицы ошибок

- Визуализировать предсказания и проанализировать ошибки модели

**Ход работы**

1. Изучаем теоретическую часть и переходим к практической. Устанавливаем и импортируем нужные библиотеки.

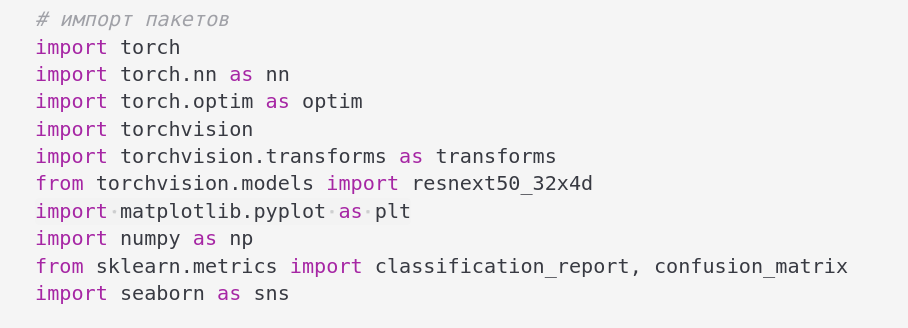


Рис 1. Импорт пакетов

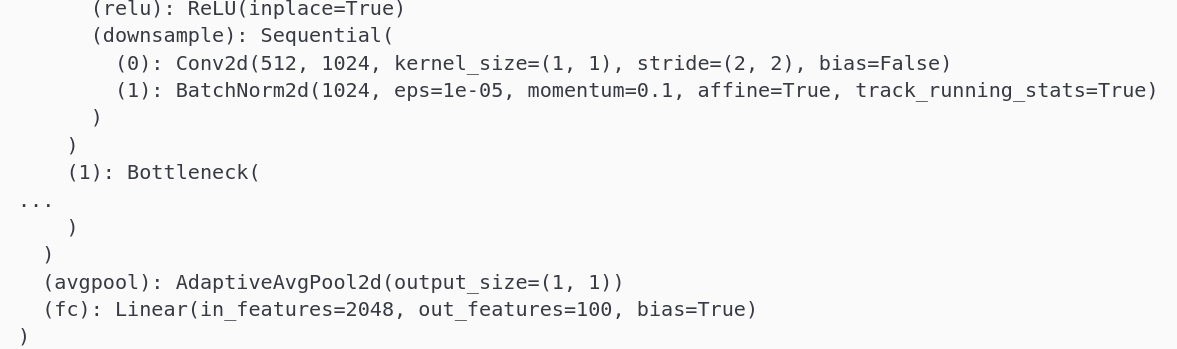
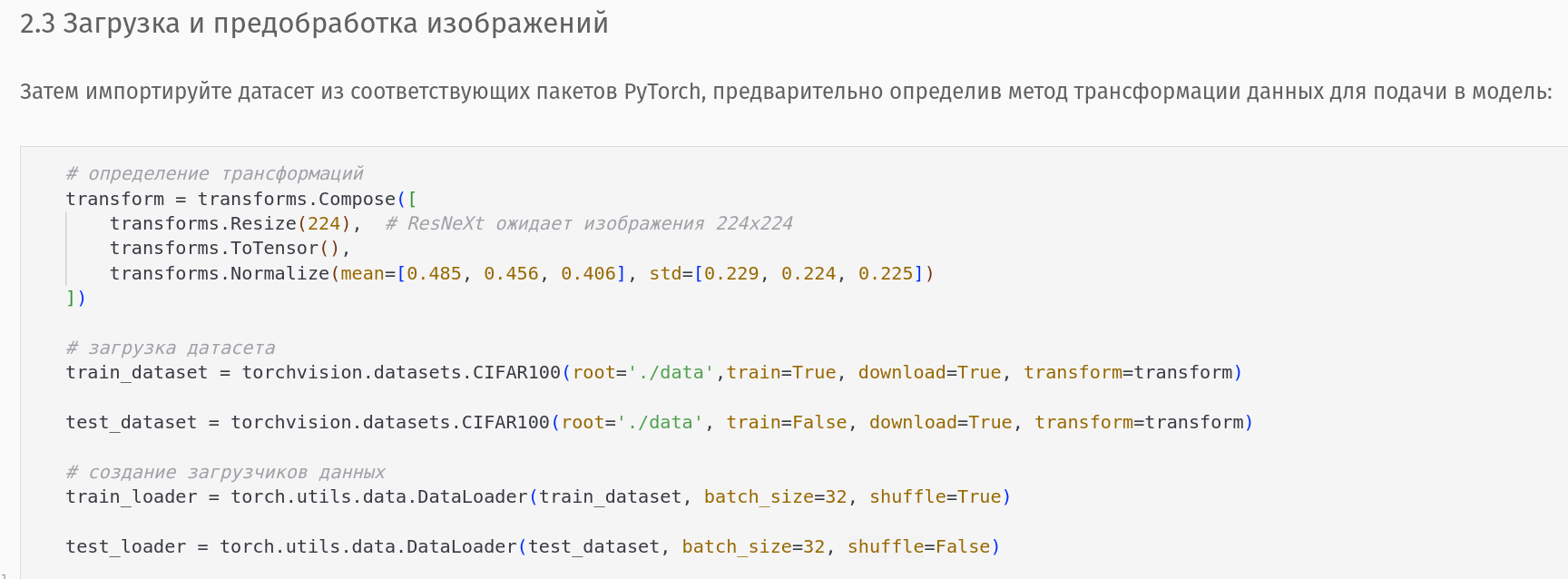
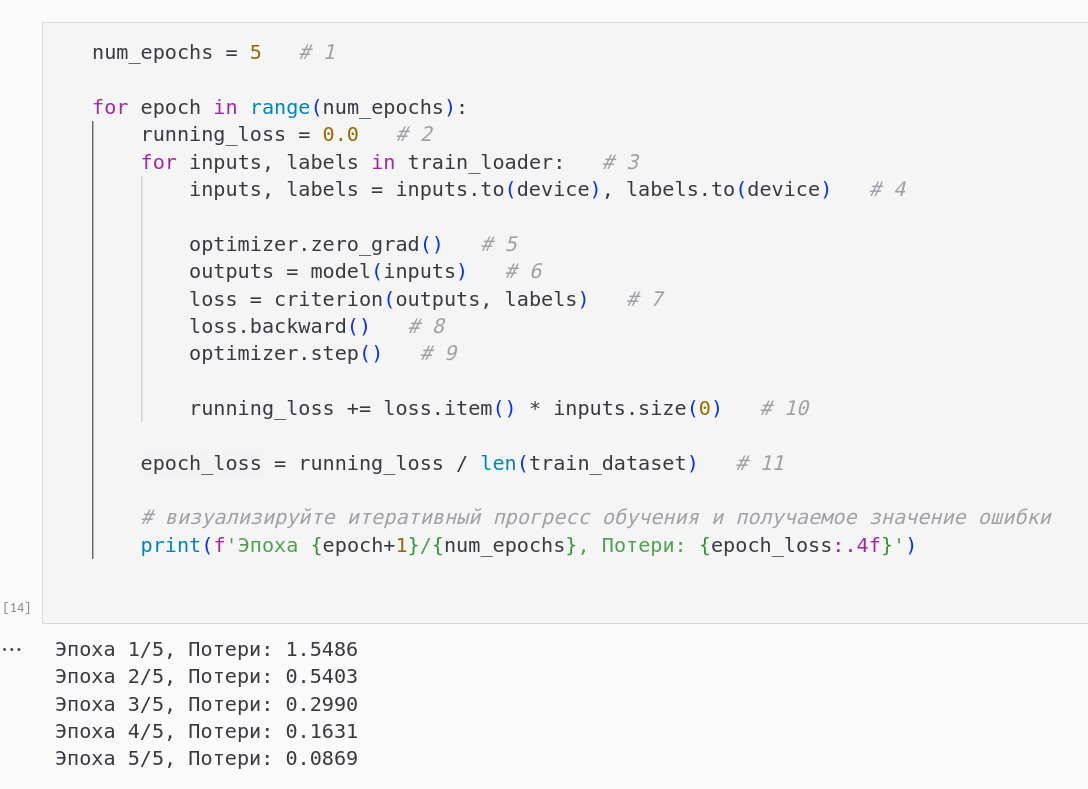
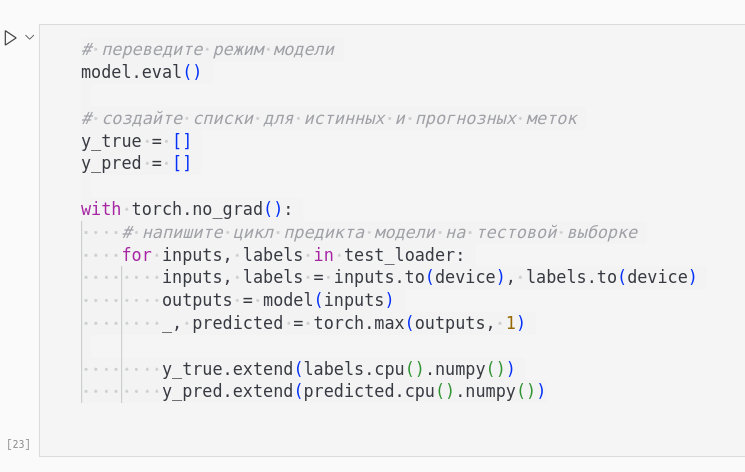
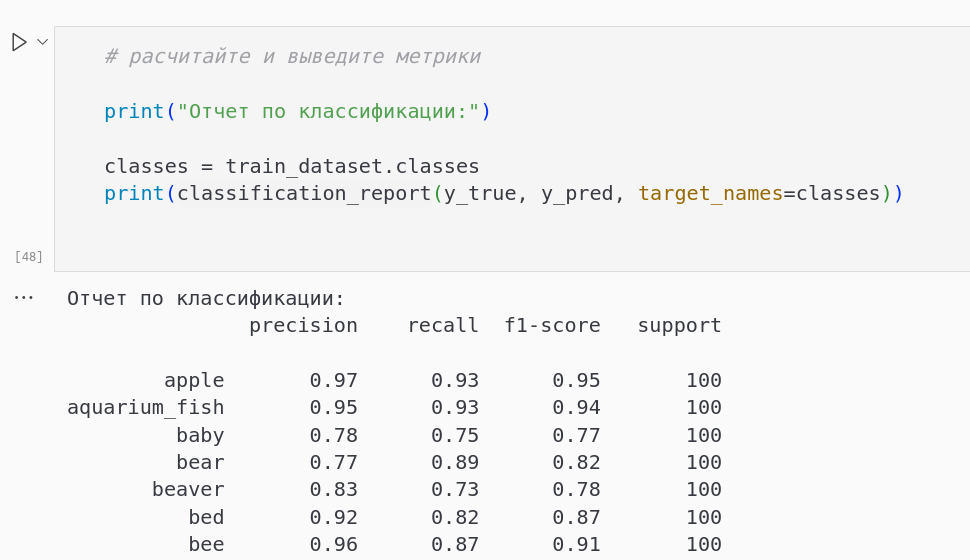
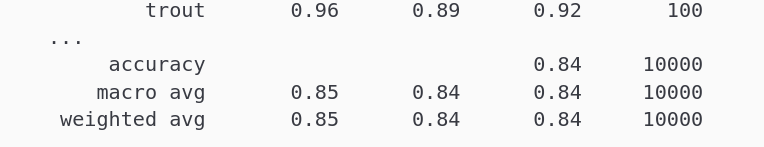
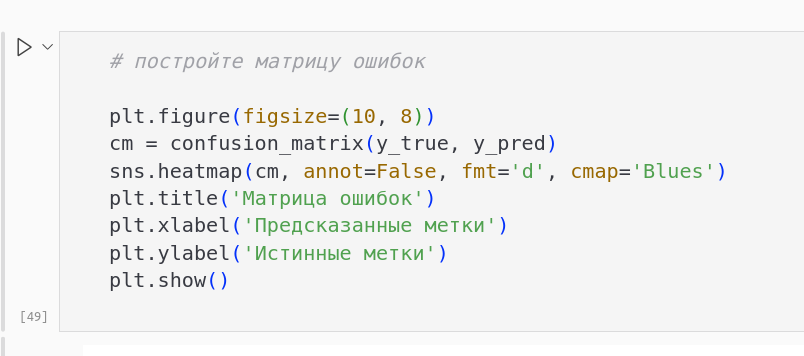
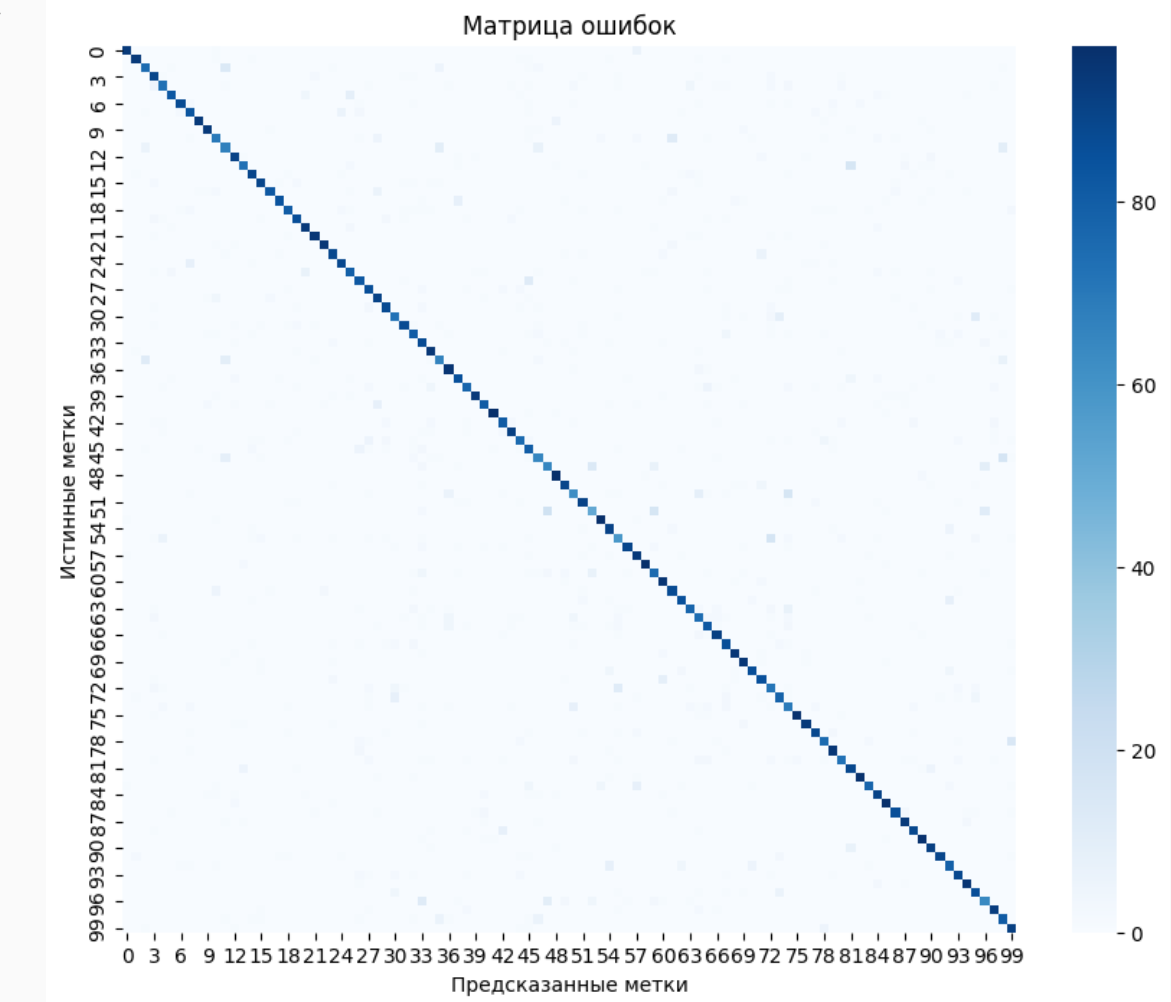
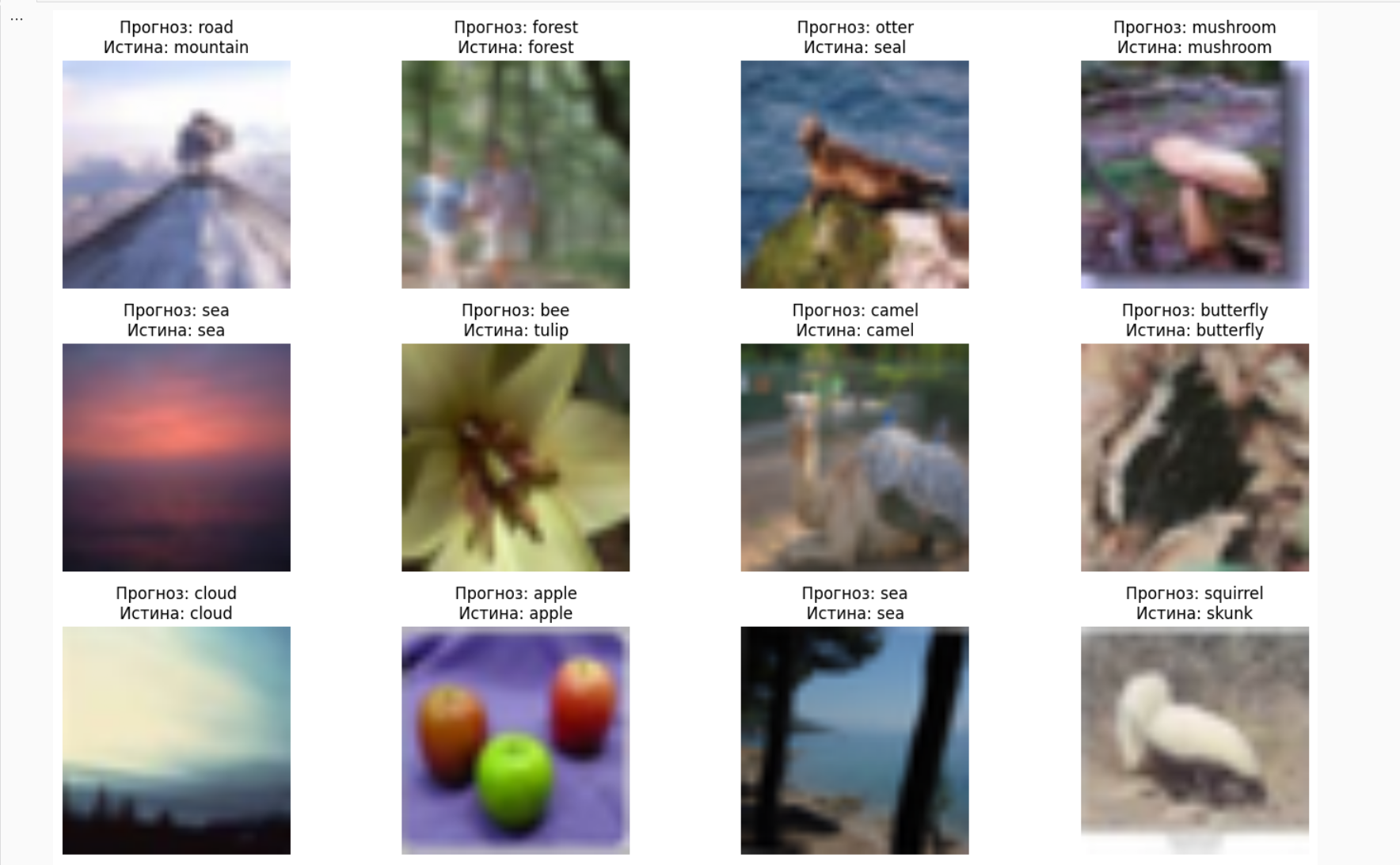
1. Определяем на каком устройстве будет выполняться программаРис 2. Использование cuda
2. Загрузка предобученной моделиРис 3. Загрузка модели
3. Замена последнего слоя модели ResNeXt-50 на линейный слой с 100 выходными нейронами Рис 4. Замена последнего слоя моделиРис 5. Отображение архитектуры модели
4. Определяем трансформацию изображений, загружаем датасет и создаем загрузчики данных Рис 6. Преобразование и загрузка данных
5. выбираем функцию потерь и оптимизатор Рис 7. Функция ошибки и оптимизатор
6. дообучаем модель на тренировочном датасете cifar-100Рис 8. Процесс дообучения модели
7. оцениваем модель на тестовом датасетеРис 9. Прогон тестовых данных через модель
8. получаем метрики моделиРис 10. Отчет классификации по классамРис 11. Общие метрики по всем классам
9. строим матрицу ошибок Рис 12. Код матрицы ошибокРис 13. Матрица ошибок
10. визуализируем предсказания модели на одном батче тестового датасета, денормализировав изображения

Рис 14. Код визуализации одного батча, метод денормализации

Рис 15. Предсказания модели и вывод изображений

**Заключение**

**Вывод:** проделав работу, мы познакомились с архитектурой модели ResNeXt, датасетом CIFAR-100, дообучили модель использю метод fine-tuning, заменив последний слой модели для классификации изображений нового датасета, визуализировали динамику обучения, оценили результат и предсказали классы одного батча тестовых данных для визульной оценки резальтата дообучения.