Проект-ДЗ по Эксплуатации

В данном проекте вы выполните комплексное домашнее задание по подготовке, оптимизации и развертыванию модели машинного обучения с использованием современных инструментов. Цель проекта – освоить процессы обучения, конвертации, оптимизации и интеграции моделей в продакшн-среду с применением Triton Inference Server, Docker и микросервисной архитектуры.

Задачи проекта

1. **Обучение модели**
    - Обучите самописную модель на базе **Torch** или **TensorFlow**.
    - Используйте стандартные слои, избегая кастомных решений.
2. **Конвертация в ONNX**
    - Экспортируйте обученную модель в формат **ONNX**.
3. **(Опционально) Конвертация в TensorRT (TRT)**
    - При необходимости, конвертируйте модель в формат **TensorRT** для повышения производительности инференса.
4. **Оптимизация модели средствами Torch/TensorFlow**
    - Примените встроенные методы оптимизации (например, quantization или pruning) для улучшения эффективности модели.
5. **Оптимизация модели инструментами ONNX и (опционально) TRT**
    - Используйте оптимизирующие инструменты для ONNX (например, ONNX Runtime) для повышения производительности.
    - (Опционально) Оптимизируйте модель в формате TensorRT.
6. **Разработка микросервиса предобработки данных**
    - Спроектируйте микросервис для предобработки данных, необходимых для работы вашей модели.
    - Реализуйте сервис с использованием **Flask**, **FastAPI** или другого подходящего фреймворка.
    - Оформите микросервис в виде **Docker-контейнера**.
7. **Развёртывание моделей с помощью Triton Inference Server**
    - Запустите **Triton Inference Server**.
    - Задеплойте следующие версии модели:
        - Оригинальная модель.
        - Оптимизированная модель (на базе Torch/TensorFlow).
        - Модель в формате ONNX.
        - Оптимизированная модель ONNX.

        - (Опционально) Модель в формате TRT.

8. **Настройка мониторинга метрик**

    - Настройте сбор и визуализацию метрик с помощью **Grafana** и **Prometheus**.
    - Обеспечьте мониторинг как для Triton Inference Server, так и для микросервиса предобработки.

9. **Оркестрация сервисов с помощью docker-compose**

    - Поднимите весь стек сервисов через **docker-compose**, включающий:
        - Микросервис предобработки данных.
        - Triton Inference Server.
        - Систему мониторинга (Prometheus и Grafana).

10. **Тестирование и формирование отчёта**

    - Проведите тестирование всех версий модели, отправляя запросы на инференс.
    - Соберите логи работы сервисов.
    - Сформируйте итоговый файл с отчётом, включающим результаты тестирования, собранные логи и анализ производительности.