Описание бизнес-логики использования A2P для разработки и продуктивизации batch ML-моделей

Данный раздел посвящен верхнеуровневому описанию действий Пользователя для создания и вывода в PROD модели машинного обучения. Предположим, что требуется разработать и вывести batch модель в PROD, чтобы эта модель делала предсказание (скоринг) по определенному расписанию (шедуленгу) и записывала результат в БД. Также пусть требуется обеспечить переобучение этой модели, но уже по другому расписанию.

Общий алгоритм работы в этом случае должен состоять из следующих этапов:

1. Создается проект в JupyterLab, содержащий необходимые шаблоны файлов и директорий. Внутри проекта можно создать Jupyter ноутбук, в котором вести написание и отладку модели. Более подробно про создание проекта и работу с JupyterLab смотрите в разделе “Работа с Jupyter”. Для разработки модели может потребоваться подключение к БД для получения данных. Если это подключение к Oracle, то подробнее см. пункт “Подключение к БД Oracle”. Если требуется подключение к Hive, то подробнее см. пункт “Подключение к БД Hive”.
   Также при обучении модели желательно логировать процесс обучения в MLflow (см. п. «Работа с MLFlow»).
   Результатом обучения модели должен стать файл model.pkl, находящийся в директории pkl внутри текущего проекта.
   Кроме файла model.pkl Пользователь должен подготовить файлы main.py, retrain.py (папка models), dag-batch.py и dag-retrain.py (папка gitlab-ci). Более подробно про назначение, особенности и необходимое содержание файлов смотрите в разделах “Файл main.py”, “Файл retrain.py”, “Файл dag-batch.py”, “Файл dag-retrain.py”, также следует ознакомиться с пунктом “Передача переменных в python скрипты”.
   Подводя итог, разработку модели можно считать завершенной, когда готовы файлы model.pkl, main.py, retrain.py, dag-batch.py, dag-retrain.py, если использовались вновь установленные библиотеки, то еще файл requirements.txt.
2. Отправить pickle файл с моделью в удаленное хранилище S3 Minio с помощью команд dvc, более подробно смотрите пункт “Работа с dvc”,
3. С помощью команд git загрузить (“запушить”) подготовленные файлы модели в ветку dev удаленного репозитория. Более подробно смотрите пункт “Работа с git”.
4. Необходимо перейти в GitLab и запустить CI/CD процесс продуктивизации модели в среде dev. Для этого перейти в удаленный репозиторий и выполнить слияние

ветки dev с веткой main. Более подробно смотрите в пункте “Работа с GitLab”.

5. Необходимо перейти в airflow и активировать DAG файл с моделью, который и будет выполнять скоринг модели по заданному в DAG-файле расписанию (шедуленгу).
6. Далее необходимо запустить CI/CD пайплайн для переобучения модели. Для этого необходимо выполнить слияние ветки main с веткой retrain. Более подробно смотрите в пункте “CI/CD процесс для переобучения модели”.
7. Перейти в airflow и активировать DAG-файл с переобучением модели. После этого, согласно расписанию, указанному в данном DAG-файле, модель будет проходить переобучение. Более подробно смотрите в пункте “CI/CD процесс для переобучения модели”.
8. Для перевода модели из среды dev в среду prod требуется выполнить слияние ветки main в ветку prod. Более подробно смотрите в пункте “CI/CD процесс для продуктивизации модели в среде PROD”.

Модель успешно разработана и поставлена на регламентное исполнение!