NU Hackathon x Beeline BigData

Немного про Beeline BigData

Beeline Kazakhstan — лучшая большая компания страны для работы data-специалиста в 2022 году. Это показал независимый опрос в профессиональном сообществе Новости Казахстанского ДС (Main ML_KZ). Команда Big Data состоит из 7 стримов: adTech, NLP, CV, CVM, Internal, Fintech, Devices. Наши сотрудники регулярно выступают и делятся лучшими практиками на крупнейших казахстанских конференциях.

Проблема

По мере роста потребностей бизнеса в все более новых и технологичных data-продуктах, в нашем хранилище данных пропорционально растет и объем генерируемых витрин. На хранение и обработку этих данных уходит огромное количество ресурсов, которое чаще всего расходуются неэффективно.

Несмотря на то, что источников данных в компании много, и их количество непрерывно растет, основные уникальные типы можно сосчитать по пальцам. Зачастую, одни и те же данные дублируются во многих витринах, занимая при этом место в хранилище и продлевая среднее время готовности просчета витрин (в связи с утилизацией ресурсов на обработку этих данных).

На данный момент, мы умеем выстраивать зависимости между витринами, стейджингами и источниками данных (DAG зависимостей в разрезе проектов). И мы:

1. Знаем какой объект зависит от другого. (DAG)
2. Знаем что и когда запускать. (Scheduling)
3. Умеем мониторить все просчеты. Реагировать в случае поломки источника. (Monitoring)
4. Умеем выстраивать оптимальные пайплайны. (Optimization)

Было бы здорово спуститься на один абстрактный уровень ниже.

Требования

Необходимо распарсить предложенные скрипты(.ipynb files)
Описание задачи в Readme

Прикладное применение для работы предложенной задачи:

- Понимать вычислительную зависимость: логическую зависимость построения показателей внутри витрин от источников
- Понимать утилизацию полей: долю использованных полей от каждого источника при расчете данной витрины
- Графическое представление зависимостей
- Полезно в случае внезапной потребности переноса источников в альтернативный контур. Например: из hadoop есть потребность перенести расчет витрины в реляционные БД и нам необходимо понять какие источники необходимо перенести
- Оптимизация кода перед выводом в продуктив и исключение неиспользуемых источников из скриптов. Посчитать количество операций внутри скрипта - если много джоинов, то это значит, что нужно упрощать скрипт.

Handouts

Вам дана ссылка на репозиторий проекта. Репозиторий состоит из композиции нескольких docker-контейнеров (docker-compose):

1. **spark_cluster** – мини кластер для обработки данных
2. **workspace_server** – серверная среда. Она состоит из:
    - **data/**
    - **projects/** – кодовая база проектов
    - **<solution>/** – пустой python-проект в котором необходимо реализовать решение
3. **web_service (optional)** – веб-сервис, для визуализации зависимостей.

Критерии оценки

Для проверки финальной версии вашего решения имеется отложенная выборка в виде одного дополнительного скрипта.

Полезные материалы

https://spark.apache.org/docs/3.1.2/api/python/reference/api/pyspark.sql.DataFrame.explain.html

https://towardsdev.com/decoding-spark-query-physical-plan-9b9682815173