Периодический запуск процедуры очистки датасета мошеннических финансовых транзакций

Домашнее задание № 4

Цель работы. В данном домашнем задании Вы потренируетесь в организации *периодического запуска* процедуры очистки данных с помощью инструмента `Apache Airflow`, познакомитесь с концепцией *ориентированных направленных графов* (DAGs), с помощью которых организуется последовательность запуска задач по расписанию, научитесь *разрабатывать собственные* графы с помощью языка `Python` для `Apache Airflow`.

Уважаемый ML-инженер!

Итак, данные очищены и загружены в хранилище. В принципе, можно бы уже начинать их анализ, однако, Вас беспокоит одна проблема. Антифрод-система продолжает работать и накапливать данные, а очистку этих данных Вы выполнили лишь единожды. Мошенники продолжают искать новые уязвимые места в системе защиты, а это означает, что модель, обученная на уже собранных и не обновляемых данных, скоро устареет и будет неспособна к качественному анализу транзакций...

Из сказанного выше вытекает необходимость создания системы, способной периодически получать новые данные из озера компании, проверять их качество, очищать и добавлять к уже существующим в Вашем хранилище. Скрипт для очистки датасета у Вас уже есть; теперь нужно с помощью `Apache Airflow` обеспечить его периодический запуск на требуемой порции новых данных из озера.

Обратите внимание!

Систему `Apache Airflow` желательно запустить в `Docker`-контейнере, подмонтировав ему в качестве `volume` директорию с разработанным DAG. Это позволит обеспечить переносимую конфигурацию системы. Однако, в этом случае, возможность работы со `Spark`-кластером изнутри контейнера будет ограничена. Поэтому, вместо `Airflow Spark Operator` рекомендуется использовать следующие операторы:

а) `SFTPOperator` для передачи на `Spark`-кластер исполняемого скрипта и связанных с ним файлов по протоколу `SFTP`;

б) `SSHOperator` для удаленного запуска скрипта на исполнение на `Spark`-кластере в рамках открытой `SSH`-сессии.

Подобный подход позволяет также запустить `Apache Airflow` на *любой* системе, с которой доступны узлы `Spark`-кластера по `SSH`.

Вам предлагается на основе представленной информации:

1. Запустить систему `Apache Airflow`, открыть её веб-интерфейс и в нём создать `SSH Connection`, в котором указать параметры доступа к `Spark`-кластеру. `Airflow` может быть запущен на одном из узлов кластера или на отдельной виртуальной машине Яндекс Облака.

Обратите внимание!

Если скрипт для очистки датасета содержит *зависимости*, реализованные в других файлах, то все эти файлы необходимо поместить в один архив или пакет, который должен быть скопирован вместе до скриптом на кластер и указан в параметрах команды `spark-submit` при запуске процедуры очистки данных.

2. Создать DAG для *ежедневного* запуска скрипта очистки датасета и разместить его в директории для DAG'ов, доступной `Apache Spark`. В графе следует прописать этапы *копирования* скрипта и необходимых ему файлов на `Spark`-кластер, а также его *запуска* на кластере посредством `spark-submit`.

3. Убедиться, что граф загрузился в систему и отображается в графическом интерфейсе. Файл(-ы) с DAG необходимо разместить в Вашем `GitHub`-репозитории и предоставить для проверки.

4. Разрешить периодическое исполнение разработанного DAG в `Apache AirFlow` и протестировать его работоспособность. Требуется дождаться *не менее трёх* успешных запусков процедуры очистки датасета по расписанию. Снимок экрана, подтверждающий успешную работу системы, необходимо привести в `README`-файле Вашего `GitHub`-репозитория.

5. В соответствии с достигнутыми результатами, изменить статус ранее созданных задач на `Kanban`-доске в `GitHub Projects`. Возможно, некоторые задачи нужно будет скорректировать, разделить на подзадачи или объединить друг с другом.

6. Полностью удалить созданный кластер, чтобы избежать оплаты ресурсов в период его простаивания.

Обратите внимание!

Для более *эффективного* управления ресурсами имеет смысл автоматизировать создание и удаление `Spark`-кластера с помощью `Apache Airflow`, включив данные действия в `DAG` процедуры очистки данных.

Для получения положительной оценки за работу необходимо выполнить *минимум* первые четыре вышеприведенных задания.

Желаем успехов!