DVC

Open-source
Version Control System
for Machine Learning Projects

ЧТО ЭТО?

- DVC - система контроля версий для дата-сетов и моделей искусственных нейронных сетей.
- Используется только в совокупности с основной системой контроля версий – GIT.

ЗАЧЕМ ЕЕ ИСПОЛЬЗОВАТЬ?

- Для сохранения результатов обучения моделей искусственных нейронных сетей;
- Для сохранения дата-сетов;
- Для сохранения модификаций дата-сетов и обученных моделей

ПОЧЕМУ НЕ GIT?

- DVC оперирует файлами больших размеров, добавление в GIT файлов более 20 МБ может существенно «раздуть» репозиторий;
- Изменение тяжелого файла в гите, особенно если он носит бинарный характер (картинка, видео, музыкальный файл и т.д.) влечет отход от инкрементного принципа сохранения изменений файла в GIT и существенного увеличения размера репозитория.
- Вышеописанные причины могут привести к парализации работы сервиса GIT-а.

КАК ХРАНИТ ФАЙЛЫ DVC?

- DVC не использует инкрементный подход к хранению изменений, а хранит все версии файлов целиком.
- Для хранения может использоваться множество способов: облачные сервисы, распределённое хранение файлов, хранение на жестком диске, хранение на сетевом диске.
- Мы используем принцип хранения на облачном сервисе. В роли сервера облачного хранения используется MinIO, совместимый с Amazon S3.

ПОЧЕМУ НЕ HDFS?

- Требует выделение трех отдельных юнитов для адекватного хранения;
- Каждый юнит использует тяжеловесные библиотеки на Java, что приводит к огромному потреблению оперативной памяти;
- Разворачивание каждого юнита в докер-контейнерах на одной машине возможно, но нецелесообразно в условиях ограниченной оперативной памяти.
- HDFS лучше использовать при наличии нескольких серверов для хранения информации и при хранении больших объемов данных (более 20 ТБ).

DVC. ОСНОВНЫЕ КОМАНДЫ

- dvc init
- dvc remote add
- dvc remote modify
- dvc add
- dvc push
- dvc pull

DVC INIT

- dvc init
- Команда для инициализации DVC в данной директории.

DVC REMOTE ADD

- dvc remote add -d myremote /path/to/remote
- Команда для добавления удаленного репозитория.

DVC REMOTE MODIFY

- dvc remote modify newremote endpointurl https://object-storage.example.com

- Команда для модификации настроек удаленного репозитория.

DVC ADD

- dvc add [-h] [-q | -v] [-R] [--no-commit] [-f <filename>]
  targets [targets ...]

- Команда для отметки файла(-ов) готовых к сохранению состояния. Данная команда создает файлы-метки.
- Файл-метка – описание для DVC какой-именно файл ему забирать. Файл-метка имеет следующее название: <Старое название файла>**.dvc**

DVC PUSH

- dvc push [-h] [-q | -v] [-j <number>]
  [-r <name>] [-a] [-T] [-d] [-R] [--all-commits]
  [targets [targets ...]]

- Команда для загрузки изменения файлов в удаленный репозиторий.

DVC PULL

- dvc pull [-h] [-q | -v] [-j <number>]
  [-r <name>] [-a] [-T] [-d] [-f] [-R] [--all-commits]
  [targets [targets ...]]

- Команда для загрузки изменений с удаленного репозитория.

DVC ОСНОВНЫЕ СТРАТЕГИИ ИСПОЛЬЗОВАНИЯ

- Создание репозитория и его настройка
- Фиксация изменений файла(-ов).
- Загрузка новых версий файла(-ов).

СОЗДАНИЕ РЕПОЗИТОРИЯ И ЕГО НАСТРОЙКА

- Команды:
  - `dvc init`
  - `dvc remote add -d origin s3://ref-info-processing`
  - `dvc remote modify origin endpointurl` https://minio.ies.mrsu.ru

- Первая команда инициализирует DVC репозиторий.
- Вторая команда добавляет новый удаленный репозиторий. Как удаленный репозиторий используется облачное хранилище S3.
- Третья команда указывает где именно находится удаленный репозиторий.

ФИКСАЦИЯ ИЗМЕНЕНИЯ ФАЙЛА (-ОВ)

- Команды:
  - `dvc add data/file1.txt`
  - `dvc add data/file2.xlsx data/file3.mp3`
  - `git add data/file1.txt.dvc data/file2.xlsx.dvc data/file3.mp3.dvc`
  - `dvc push`
  - `git commit –m “Comment”`
  - `git push origin my_branch`

- Первые две команды показывают варианты добавления изменений файла и файлов (соответственно) в DVC.
- Далее мы добавляем в GIT новые версии файлов-меток для DVC (файлы с расширением **.dvc**).
- После этого отправляем изменения в DVC.
- Делаем коммит в GIT и отправляем его на удаленный репозиторий.

ЗАГРУЗКА НОВЫХ ВЕРСИЙ ФАЙЛА(-ОВ)

- Команды:
  - `git pull origin my_branch`
  - `dvc pull`

- Сначала необходимо из GIT-а загрузить новые версии файлов-меток.
- После этого можно получить изменения из DVC.

ОСОБЕННОСТИ ИСПОЛЬЗОВАНИЯ DVC

- DVC в нашем проекте используется вместе с HTTPS подключением к MinIO. Для корректного подключения к MinIO требуется указание ключа доступа и секретного ключа.
- При работе через JupyterHub вы работаете на удаленном сервере, там эти ключи уже прописаны в системе.
- Если же вы хотите получить/загрузить изменения на свой компьютер, то необходимо будет заранее указать эти ключи.

ДАННЫЕ ДЛЯ ПОДКЛЮЧЕНИЯ

- Адрес конечной точки: https://minio.ies.mrsu.ru
- Ключ доступа: UDAm3LAza0LmfJRNIht4
- Секретный ключ: GdKc1nCqBu2zVaA1w7xN

ПОДКЛЮЧЕНИЕ С МАШИНЫ С ОС LINUX

- Необходимо заранее прописать данные ключей.
- Сделать это можно через установку переменных окружения.
- Так же, можно указать эти переменные окружения только для текущей сессии командной строки.
- Например (это **одна** команда):
  - AWS_SECRET_ACCESS_KEY="GdKc1nCqBu2zVaA1w7xN" AWS_ACCESS_KEY_ID="UDAm3LAza0LmfJRNIht4" dvc pull

ПОДКЛЮЧЕНИЕ С МАШИНЫ С ОС WINDOWS

- Необходимо заранее прописать данные ключей.
- Сделать это можно через установку переменных окружения.
- Так же, можно указать эти переменные окружения только для текущей сессии командной строки.
- Например (используется **PowerShell**):
  - `$env:AWS_SECRET_ACCESS_KEY="GdKc1nCqBu2zVaA1w7xN"`
  - `$env:AWS_ACCESS_KEY_ID="UDAm3LAza0LmfJRNIht4"`
  - `dvc pull`

ЧТО НЕ СТОИТ ДЕЛАТЬ?

- Не трогайте папку **.dvc**.
- Не добавляйте в GIT основную версию файла, только его метку, оканчивающуюся на **.dvc**.
- Не сохраняйте ключи в каком-то отдельном файле, который хранится в репозитории.