

PySpark job



Importazione massiva di dati con PySpark e AWS Glue

Cattaneo Andrea - mat. 1040655

Locatelli Matteo - mat. 1041449

Job PySpark

Una criticità in cui ci siamo sin da subito imbattuti è il fatto che in condizioni di cold start i **job AWS GLUE vengono eseguiti con circa 10-15 minuti di ritardo** rispetto l'inserimento dell'attività nel sistema.

In contesti di produzione questo potrebbe non essere un grosso problema (se il job viene eseguito alle 3:30 o alle 3:45 poco cambia), ma **pone un grande limite alla velocità di sviluppo**, poiché si deve attendere un tempo eccessivo prima di verificare il corretto funzionamento dello script.

Per superare il problema in fase di sviluppo abbiamo utilizzato l'immagine docker [jupyter/pyspark-notebook](https://hub.docker.com/r/jupyter/pyspark-notebook)

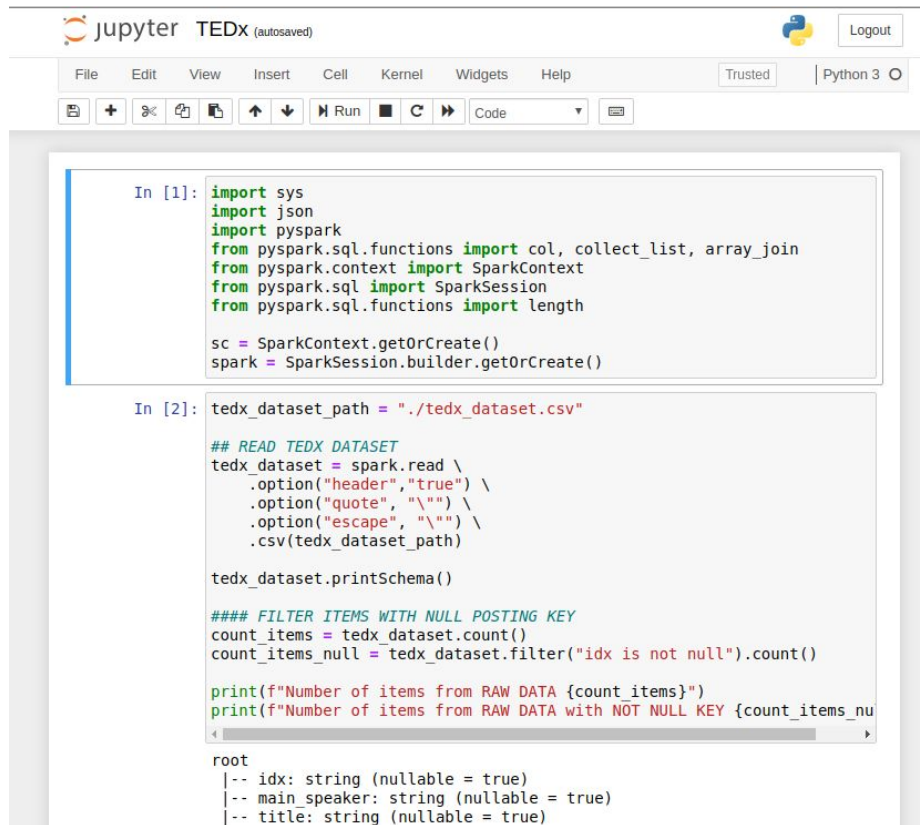
Job PySpark

Con [jupyter/pyspark-notebook](https://github.com/imcatta/tedx_party/blob/master/pyspark/TEDx.ipynb) è possibile istanziare un Jupyter notebook collegato ad un cluster Spark direttamente sul proprio PC con poco sforzo.

```
> docker run -p 8888:8888  
jupyter/pyspark-notebook
```

In questo modo abbiamo a disposizione un ambiente dove poter eseguire rapidamente il nostro codice prima di caricarlo in AWS GLUE

https://github.com/imcatta/tedx_party/blob/master/pyspark/TEDx.ipynb



```
jupyter TEDx (autosaved) Logout  
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3  
In [1]: import sys  
import json  
import pyspark  
from pyspark.sql.functions import col, collect_list, array_join  
from pyspark.context import SparkContext  
from pyspark.sql import SparkSession  
from pyspark.sql.functions import length  
  
sc = SparkContext.getOrCreate()  
spark = SparkSession.builder.getOrCreate()  
  
In [2]: tedx_dataset_path = "./tedx_dataset.csv"  
  
## READ TEDX DATASET  
tedx_dataset = spark.read \  
    .option("header", "true") \  
    .option("quote", "\"") \  
    .option("escape", "\\") \  
    .csv(tedx_dataset_path)  
  
tedx_dataset.printSchema()  
  
#### FILTER ITEMS WITH NULL POSTING KEY  
count_items = tedx_dataset.count()  
count_items_null = tedx_dataset.filter("idx is not null").count()  
  
print(f"Number of items from RAW DATA {count_items}")  
print(f"Number of items from RAW DATA with NOT NULL KEY {count_items_nu  
root  
|-- idx: string (nullable = true)  
|-- main speaker: string (nullable = true)  
|-- title: string (nullable = true)
```

Dataset

Nel corso del nostro lavoro abbiamo individuato alcune criticità nei dataset

- In `tedx_dataset` sono presenti alcune righe “sporche”.

```
+-----+-----+-----+
|          idx|   main_speaker|         title|
+-----+-----+-----+
...
|Elisabeth Pierre ...|          null|          null|
|Elisabeth est zyt...| une des quelques...| chercheurs et cu...|
```

Per filtrarle verifichiamo che la lunghezza di `idx` sia pari a 32

```
> tedx_dataset_agg = tedx_dataset_agg.filter(length(col('idx')) == 32)
```

Con questo filtro abbiamo rimosso 27 righe

Dataset

- In `watch_next` sono stati trovati numerosi duplicati.

...

```
8d2005ec35280deb6a438dc87b225f89,https://www.ted.com/talks/ronald_rael_an_architect_s_subversive_re  
imagining_of_the_us_mexico_border_wall,5bd34fcc55d9e1267f605fa0c060d54e  
8d2005ec35280deb6a438dc87b225f89,https://www.ted.com/talks/ronald_rael_an_architect_s_subversive_re  
imagining_of_the_us_mexico_border_wall,5bd34fcc55d9e1267f605fa0c060d54e  
8d2005ec35280deb6a438dc87b225f89,https://www.ted.com/session/new?context=ted.www%2Fwatch-later,9f7b  
1654e792011b7e1c6f4288520226  
8d2005ec35280deb6a438dc87b225f89,https://www.ted.com/session/new?context=ted.www%2Fwatch-later,9f7b  
1654e792011b7e1c6f4288520226
```

...

Per pulire il dataset abbiamo rimosso i duplicati

```
> wn_dataset=wn_dataset.dropDuplicates()
```

Con questo filtro abbiamo rimosso ben 47.110 righe!

Dataset

Una volta effettuato il join di tutti i dataset (utilizzando come chiave l'id univoco del singolo TEDx talk) la struttura risultante è la seguente:

```
{
  "_id": "7d12cb15860436be16912357f0be08a2"
  "main_speaker": "Johanna Figueira"
  "title": "Simple, effective tech to connect communities in crisis"
  "details": "The world is more connected than ever, but some communities are still ..."
  "posted": "Posted Mar 2020"
  "url": "https://www.ted.com/talks/johanna_figueira_simple_effective_tech_to_co..."
  "tags": Array
    0: "TED"
    1: "talks"
    2: "social media"
    3: "activism"
    4: "technology"
    5: "South America"
    6: "community"
    7: "infrastructure"
    8: "government"
  "watch_next_ids": Array
    0: "989350baed4da5f4e3d52486be7ed5d8"
    1: "8743389cd4b81bbb2f86a877b89358bb"
    2: "9f7b1654e792011b7e1c6f4288520226"
    3: "f14e7d08564e3ffc76e1b0d058170d4a"
    4: "8dc784f2d73e5ca9b45d93b89298be99"
    5: "c4ef6e7ddcd1e2b35d83b236664665c"
    6: "05b2186a8f76d5a34a6c59a185664842"
```

```

  "_id": "8d2005ec35280deb6a438dc87b225f89"
  "main_speaker": "Alexandra Auer"
  "title": "The intangible effects of walls"
  "details": "More barriers exist now than at the end of World War II, says designer..."
  "posted": "Posted Apr 2020"
  "url": "https://www.ted.com/talks/alexandra_auer_the_intangible_effects_of_wal..."
  "tags": Array
    0: "TED"
    1: "talks"
    2: "design"
    3: "society"
    4: "identity"
    5: "social change"
    6: "community"
    7: "humanity"
    8: "TEDx"
  "watch_next_ids": Array
    0: "8576654442b6633b1dc0eb48a989172a"
    1: "5134ae81a27c94354173f38e84289ad5"
    2: "fe35edd737282ab3a325f2387cf1b50b"
    3: "9f7b1654e792011b7e1c6f4288520226"
    4: "078766d6cc461cf71d45dc268b66db95"
    5: "d9896b41b372ec60cdd3c662e57caad3"
    6: "5bd34fcc55d9e1267f605fa0c060d54e"
```

Abbiamo scelto di utilizzare solo gli id per watch_next invece che riportare tutto il documento linkato per non appesantire eccessivamente la dimensione del database

Criticità tecniche

Le criticità principali sono legate al corretto funzionamento dello scraper che fornisce i file csv da importare. Un malfunzionamento può causare errori nell'importazione, o importazioni inconsistenti.

Inoltre, l'inizializzazione del job può rendere questa operazione inadatta se dobbiamo elaborare i dati in tempi ristretti. L'avvio ritardato del job infatti può far slittare l'importazione anche di svariati minuti.

Possibili evoluzioni

Una possibile evoluzione può essere l'introduzione di un ordinamento sui consigliati in base ai tag in comune.

Un'ulteriore miglioria potrebbe essere quella di associare ad ogni video presente in watch_next anche un rating da 1 a 10 che indica quando tale video è coerente con il video appena visualizzato, in modo da poter ordinare i video suggeriti per mostrare prima i video più inerenti a quello visualizzato

Infine per ogni video si potrebbe andare a costruire un rating personalizzato sugli interessi di ogni utente (un po' come i video suggeriti di youtube).