Essbai Wasim (1060652)
Locatelli Matteo (1059210)

TEDx PausaAttiva

01

Obiettivi

Sono stati sviluppati due job con lo scopo di:

+ Data cleaning: I dataset forniti contenevano **duplicati**, **ID non validi** e **record vuoti**.

+ Aggregazione: Ai dati dei talk iniziali verranno aggiunti i campi **tags**, **watch_next** e **num_likes**. Il campo num_likes non è presente nel dataset originale e verrà ottenuto tramite web scraping.

Il dato **watch_next** verrà utilizzato per suggerire talk agli utenti che desiderano esplorare nuovi video.Inoltre, il campo **num_likes** potrà essere utilizzato per migliorare la qualità delle raccomandazioni, privilegiando i talk più apprezzati.

Data Cleaning

```
### READ AND CLEAN TEDx DATASET
tedx_dataset = spark.read \
    .option("header","true") \
    .option("quote", "\"") \
    .option("escape", "\"") \
    .option("multiline", "true") \
    .csv(tedx_dataset_path)
```

Il file .csv viene letto con option("multiline", "true") per evitare che spark consideri i detail multilinea come record sepratati.

Prima di rimuovere i duplicati, il processo di filtraggio elimina i record con ID non validi. Per identificare gli ID non validi, si considerano i casi in cui l'ID è nullo, contiene spazi o non ha una lunghezza corretta.

```
### FILTERING
count_items = tedx_dataset.count()
tedx_dataset_clean = tedx_dataset.filter("idx is not null")
count_items_null = tedx_dataset_clean.count()
tedx_dataset_clean = tedx_dataset_clean.filter(length(col('idx')) == 32)
count_correct_idx_length = tedx_dataset_clean.count()
tedx_dataset_clean = tedx_dataset_clean.filter(~col("idx").contains(" "))
count_no_space_id = tedx_dataset_clean.count()
tedx_dataset_clean = tedx_dataset_clean.dropDuplicates()
count_no_dup = tedx_dataset_clean.count()
```

```
Tedx Dataset: Number of items from RAW DATA: 5096
Tedx Dataset: Number of items from RAW DATA with NOT NULL KEY: 5096
Tedx Dataset: Number of items from RAW DATA with NO SPACES 5096
Tedx Dataset: Number of items from RAW DATA with CORRECT IDX LENGTH 5096
Tedx Dataset: Number of items from RAW DATA with NO DUPLICATES 5096
```

Il dataset era già pulito.

Data Cleaning

Nel dataset *watch_next* sono presenti record con URL non validi, il che potrebbe portare alla presenza di record duplicati che differiscono solo per l'URL non valido. Tuttavia, poiché l'informazione sull'URL è già presente nel dataset *tedx_dataset*, è possibile semplificare il dataset *watch_next* eliminando la colonna "url".

```
## FILTERING
count_items = watch_next_dataset.count()
watch_next_dataset_clean = watch_next_dataset.filter("idx is not null")
count_items_null = watch_next_dataset_clean.count()
watch_next_dataset_clean = watch_next_dataset_clean.filter(length(col('idx')) == 32)
count_correct_idx_length = watch_next_dataset_clean.count()
watch_next_dataset_clean = watch_next_dataset_clean.filter(~col("idx").contains(" "))
count_no_space_id = watch_next_dataset_clean.count()
watch_next_dataset_clean = watch_next_dataset_clean.drop("url")
watch_next_dataset_clean = watch_next_dataset_clean.dropDuplicates()
count_no_dup = watch_next_dataset_clean.count()
```

```
Watch Next Dataset: Number of items from RAW DATA: 87846
Watch Next Dataset: Number of items from RAW DATA with NOT NULL KEY: 87846
Watch Next Dataset: Number of items from RAW DATA with NO SPACES 87846
Watch Next Dataset: Number of items from RAW DATA with CORRECT IDX LENGTH 87846
Watch Next Dataset: Number of items from RAW DATA with NO DUPLICATES 34374
```

Il dataset finale contiene 34374 records, 53.472 in meno.

Numero di like

- Il numero di like viene ottenuto tramite web scraping con Selenium.
- Lo scraper crea un file .csv con coppie (idx,num_likes).
- Link allo Scraper.py

Durante l'esecuzione dello Scraper è stato scoperto che 6 talk presenti nel dataset non sono più disponibili su TED, evidenziando la necessità di un aggiornamento frequente dei dati relativi ai talk.

Aggregazione & Script

```
# CREATE THE AGGREGATE MODEL, ADD TAGS TO TEDX_DATASET
tags_dataset_agg = tags_dataset.groupBy(col("idx").alias("idx_ref")).agg(collect_list("tag").alias("tags"))
tags_dataset_agg.printSchema()

tedx_dataset_agg = tedx_dataset.join(tags_dataset_agg, tedx_dataset.idx == tags_dataset_agg.idx_ref, "left") \
    .drop("idx_ref") \
    .select(col("idx").alias("_id"), col("*")) \
    .drop("idx") \
```

Ad ogni talk in *tedx_dataset* viene aggiunta la lista dei tag associati.

In questo pezzo di script viene aggiunta ad ogni talk la lista dei watch next.

```
# UPDATE THE AGGREGATE MODEL, ADD WATCH NEXT TO TEDX_DATASET
watch_next_dataset_agg = watch_next_dataset.groupBy(col("idx").alias("idx_ref")).agg(collect_list("watch_next_idx").alias("watch_next"))
watch_next_dataset_agg.printSchema()

tedx_dataset_agg = tedx_dataset_agg.join(watch_next_dataset_agg, tedx_dataset_agg._id == watch_next_dataset_agg.idx_ref, "left") \
    .drop("idx_ref") \
```

```
# UPDATE THE AGGREGATE MODEL, ADD NUMBER OF LIKES TO TEDX_DATASET
num_likes_dataset = num_likes_dataset.withColumnRenamed("idx", "idx_ref")
num_likes_dataset.printSchema()

tedx_dataset_agg = tedx_dataset_agg.join(num_likes_dataset, tedx_dataset_agg._id == num_likes_dataset.idx_ref, "left") \
    .drop("idx_ref") \
```

Qui viene aggiunto il numero di like.

Il modello aggregato creato con tutti i dati viene salvato in un'istanza di database su MongoDB.

Schema dati iniziale

```
_id: "3b5f6a108cac226703a717e6ff9692e3"
main_speaker: "Neil R. Jeyasingam"
title: "How do antidepressants work?"
details: "In the 1950s, the discovery of two new drugs sparked what would become…"
posted: "Posted Mar 2021"
url: "https://www.ted.com/talks/neil_r_jeyasingam_how_do_antidepressants_wor…"
num_views: "556,374"
duration: "4:39"
```

Schema dati Finale

```
_id: "3b5f6a108cac226703a717e6ff9692e3"
main_speaker: "Neil R. Jeyasingam"
title: "How do antidepressants work?"
details: "In the 1950s, the discovery of two new drugs sparked what would become…"
posted: "Posted Mar 2021"
url: "https://www.ted.com/talks/neil_r_jeyasingam_how_do_antidepressants_wor…"
num_views: "556,374"
duration: "4:39"
▸ tags: Array
▸ watch_next: Array
num_likes: "112K"
```

Criticità teniche

- Necessità di aggionare periodicamente la lista dei talk. In particolare il numero di like cambia con frequenza elevata.
- Presenza di record con detail multilinea che rendono necessario opportuni accorgimenti.
- Fragilità dello scraper basato su selenium, che diventa molto sensibile ai cambiamenti della pagina web, rendendo necessaria una manutenzione frequente dello script.

Possibili evoluzioni

- Scraping periodico tramite job/lambda function su AWS.
- Valutare l'idea di usare uno script per lo scraping più robusto a cambiamenti della pagina.
- A partire da *tags_dataset* creare una collezione che contiene i talk sotto un certo tag, per fornire un ulteriore servizio di suggerimento.

TEDxPausaAttiva

Board GitHub

Thank you!