TEDInsight

Apprendimento Personalizzato dai TED Talks

Trasformare il modo in cui gli utenti interagiscono con i contenuti TED

Job AWS Glue

Job Create_Data_Lake

Lo scopo principale dello script è elaborare, aggregare e archiviare i dati dei dataset TEDx su MongoDB, consentendo un'analisi e una consultazione più efficienti.

- **Data cleaning:** rimozione dei dati nulli e duplicati per tutti i dataset.
- **PySpark Job:** creazione delle collections per identificare ogni video.

```
## READ TAGS DATASET
tags_dataset_path = "s3://tedx-insight-data/tags.csv"
tags_dataset = spark.read.option("header","true").csv(tags_dataset_path)

##### CLEAN TAGS DATASET
print(f"TOTAL TAGS DATASET: {tags_dataset.count()}")
tags_dataset = tags_dataset.dropDuplicates()
#### REMOVE DUPLICATES
print(f"TAGS DATASET without DUPLICATES {tags_dataset.count()}")
```

Job Related_Videos

Lo scopo principale dello script è quello di incorporare all'interno della documentazione su MongoDB di ogni talk, l'array watch_next contenente gli id dei video consigliati.

- **Data cleaning:** rimozione dei dati nulli e duplicati per tutti i dataset.
- **Aggregate model:** Creazione del modello aggregato aggiungendo i dati "watch_next" al dataset aggregato TEDx e Tags.

```
##### READ RELATED VIDEOS DATASET
related_videos_path = "s3://tedx-insight-data/related_videos.csv"
related_videos = spark.read.option("header","true").csv(related_videos_path)

##### CLEAN RELATED VIDEOS DATASET
print(f"TOTAL RELATED VIDEOS DATASET: {related_videos.count()}")
related_videos = related_videos.dropDuplicates()
#### REMOVE DUPLICATES
print(f"RELATED VIDEOS DATASET without DUPLICATES {related_videos.count()}")

# CREATE THE AGGREGATE MODEL, ADD TAGS TO TEDX_DATASET
tags_dataset_agg = tags_dataset.groupBy(col("id").alias("id_ref")).agg(collect_list("tag").alias("tags"))
tags_dataset_agg.printSchema()
tedx_dataset_agg = tedx_dataset_main.join(tags_dataset_agg, tedx_dataset.id == tags_dataset_agg.id_ref, "left") \
    .drop("id_ref") \
    .select(col("id").alias("_id"), col("*")) \

##### CREATE THE AGGREGATE MODEL, ADD RELATED_VIDEOS TO TEDX_DATASET
related_videos_agg = related_videos.groupBy(col("id").alias("id_ref")).agg(collect_list("related_id").alias("related_videos"))
```

Collection su MongoDB

Collection: tedx_data

Struttura dati ottimizzata per l'archiviazione dei talk TEDx

Ogni documento della collezione rappresenta un talk. Ogni elemento è identificato da un id univoco, vengono inoltre fornite varie informazioni riguardo il contenuto e le visualizzazioni del video.

Struttura del documento

Identificatore Univoco
Ogni talk ha un ID specifico

Metadata del Talk
Titolo, descrizione, durata, data

Dati Analitici
Visualizzazioni, engagement

Array di Tags

I tag associati al talk vengono inseriti in un array.

```
"tags": [ "innovation", "technology", "future"]
```

Array watch_next

Contiene gli ID dei talk consigliati come "watch_next".

```
"watch_next": [ 2473, 3142, 1872]
```

Sviluppi Futuri

Miglioramento dei Tag

Inserire una maggiore quantità di tag all'interno dei dati di ciascun talk in modo più attinente per migliorare la qualità dei video suggeriti.

Benefici attesi:

- Raccomandazioni più precise e personalizzate
- Migliore categorizzazione dei contenuti
- Esperienza di scoperta dei contenuti più ricca

Filtro per Data di Pubblicazione

Prendere in considerazione la data di pubblicazione dei talk per consigliare video più recenti e rilevanti per le tendenze attuali.

Vantaggi principali:

- Contenuti più aggiornati e pertinenti
- Bilanciamento tra classici e novità
- Possibilità di creare percorsi tematici evolutivi nel tempo