TED IN FILA

Giulia Migliorati
Luca Frangiamore
Michael Marzella

Job WatchNext

Basandoci sul dataset Watch next fornitoci, abbiamo collegato ad ogni video la propria **lista di prossimi video da vedere**

```
# Query per WatchNext
tags_dataset_agg = tedx_tag.groupBy(col("idx")).agg(collect_list("tag").alias("tags"))
tags_dataset_agg.printSchema()


watch_next_dataset = tedx_next.groupBy(col("idx").alias("idx_ref")).agg(array_distinct(collect_list(col("url"))).alias("watch_next_url"))

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

tedx_dataset_agg.printSchema()

result = tedx_dataset_agg.join(watch_next_dataset, tedx_dataset_agg._id == watch_next_dataset.idx_ref, "left") \
    .drop(watch_next_dataset.idx_ref) \
    .select(col("_id"), col("*")) \

result.printSchema()
```

Job RangeCustomers

Per prima cosa abbiamo fatto una distinzione fra **clienti del mattino e clienti del pomeriggio** creando due differenti job, rispettivamente RangeCustomers Morning e RangeCustomers Afternoon.

Abbiamo poi filtrato i video in base all'**argomento** in modo che colpissero l'attenzione dei clienti della rispettiva fascia oraria.

Script RangeCustomers

```
#Query for RangeCustomers
tags=["beauty","family","femminism","fashion","friendship","garden","health care","love","life","nature","parenting","relationship","women"]

# Filter the id_tags DataFrame to only include rows with the "life" tag
id_tags = tedx_tag.filter(col("tag").isin(tags)).select(col("idx")).distinct()

# Join the tedx_dataset and id_tags DataFrames on the "idx" column, and filter to include only rows with the "life" tag
tedx_dataset_agg = tedx_dataset.join(
    id_tags,
    ["idx"],
    "inner"
).select(
    col("idx").alias("_id"),
    col("*")
).drop("idx","tag")
```

Schema finale WatchNext

Informazioni riguardanti il video e l'array di url dei watch next


```
_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
▾ watch_next: Array
    0: "https://www.ted.com/talks/ronald_rael_an_architect_s_subversive_reimag…"
    1: "https://www.ted.com/session/new?context=ted.www%2Fwatch-later"
    2: "https://www.ted.com/talks/megan_campisi_and_pen_pen_chen_what_makes_th…"
```


Schema finale RangeCustomers


```
_id: "b3072cd11f40eb57fd259555264476c6"
main_speaker: "Elizabeth Gilbert"
title: "It's OK to feel overwhelmed. Here's what to do next"
details: "If you're feeling anxious or fearful during the coronavirus pandemic, …"
posted: "Posted Apr 2020"
url: "https://www.ted.com/talks/elizabeth_gilbert_it_s_ok_to_feel_overwhelme…"
```


Esempio di video selezionato in quanto presenta un tag fra quelli preselezionati

Criticità

- **Dati non sempre consistenti**, elaborati e filtrati tramite l'utilizzo di **distinct**
- Ricerca di informazioni per **scelta dei tag** più adatti ad ogni range di clienti

Possibili sviluppi

- Possibilità di **aggiornare** il dataset
- Aggiungere funzionalità tramite **web scraping,** come per esempio mostrare la **copertina del video**