HOMEWORK2

CRISTIAN LOCATELLI – 1041279
ANDREA PAGANESSI – 1040464
STEFANO VILLA – 1040633

Job PySpark

```
## READ TAGS DATASET
tags_dataset_path = "s3://unibg-tedx-data-99/tags_dataset.csv"
tags_dataset = spark.read.option("header","true").csv(tags_dataset_path)


# READ TAGS DATASET
watch_next_path = "s3://unibg-tedx-data-99/watch_next_dataset.csv"
watch_next_dataset = spark.read.option("header","true").csv(watch_next_path)


# 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("*")) \
```

Job PySpark

```
19  # CREATE THE AGGREGATE MODEL, ADD WATCH NEXT TO TEDX_DATASET
20
21  zipper = udf(lambda x, y: [list(z) for z in zip(x, y)], ArrayType(StringType()))
22
23  watch_next_dataset_agg = watch_next_dataset.groupBy(col("idx").alias("idx_ref")) \
24      .agg(collect_list("watch_next_idx").alias("next"), collect_list("url").alias("next_url")) \
25      .withColumn("next", zipper(col("next"),col("next_url"))) \
26      .drop("next_url")
27
28  watch_next_dataset_agg.printSchema()
29
30  tedx_dataset_agg = tedx_dataset_agg.join(watch_next_dataset_agg, tedx_dataset.idx == watch_next_dataset_agg.idx_ref, "left") \
31      .drop("idx_ref") \
32      .select(col("idx").alias("_id"), col("*")) \
33      .drop("idx") \
```

Dati e Schema Finale


```
_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"
next: Array
  0: "[5bd34fcc55d9e1267f605fa0c060d54e, https://www.ted.com/talks/ronald_ra..."
  1: "[5bd34fcc55d9e1267f605fa0c060d54e, https://www.ted.com/talks/ronald_ra..."
  2: "[9f7b1654e792011b7e1c6f4288520226, https://www.ted.com/session/new?con..."
  3: "[fe35edd737282ab3a325f2387cf1b50b, https://www.ted.com/talks/megan_cam..."
  4: "[fe35edd737282ab3a325f2387cf1b50b, https://www.ted.com/talks/megan_cam..."
  5: "[9f7b1654e792011b7e1c6f4288520226, https://www.ted.com/session/new?con..."
  6: "[d9896b41b372ec60cdd3c662e57caad3, https://www.ted.com/talks/julia_dha..."
  7: "[d9896b41b372ec60cdd3c662e57caad3, https://www.ted.com/talks/julia_dha..."
  8: "[9f7b1654e792011b7e1c6f4288520226, https://www.ted.com/session/new?con..."
  9: "[5134ae81a27c94354173f38e84289ad5, https://www.ted.com/talks/anna_heri..."
  10: "[5134ae81a27c94354173f38e84289ad5, https://www.ted.com/talks/anna_heri..."
  11: "[9f7b1654e792011b7e1c6f4288520226, https://www.ted.com/session/new?con..."
  12: "[8576654442b6633b1dc0eb48a989172a, https://www.ted.com/talks/alex_honn..."
  13: "[8576654442b6633b1dc0eb48a989172a, https://www.ted.com/talks/alex_honn..."
  14: "[9f7b1654e792011b7e1c6f4288520226, https://www.ted.com/session/new?con..."
  15: "[078766d6cc461cf71d45dc268b66db95, https://www.ted.com/talks/will_hurd..."
  16: "[078766d6cc461cf71d45dc268b66db95, https://www.ted.com/talks/will_hurd..."
  17: "[9f7b1654e792011b7e1c6f4288520226, https://www.ted.com/session/new?con..."
```

Criticità

- ✓ Tempo di debug:

  usando AWS Glue è necessario attendere dai 10 ai 20 minuti, ovvero tutto il costo legato all'istanziazione dell'infrastruttura, anche quando ci sono errori nel codice; a questo si aggiunge

- ✓ Impossibilità di eseguire il codice offline:

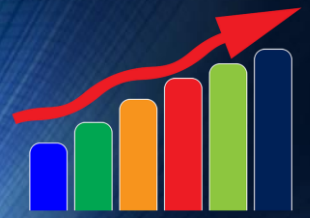
  le funzionalità di AWS non è possibile importarle/utilizzarle su di un IDE diverso da AWS quindi qualsiasi test si voglia effettuare richiede il tempo d'attesa sopra indicato

Evoluzione

Funzionalità da poter aggiungere:

- «**Circles**»: utilizzando il dataset watch_next sarebbe possibile, partendo da un video a scelta dell'utente, generare un albero *di figli dei figli* dei watch_next finché una delle foglie non corrisponda al video radice dell'albero.

  In quel modo il ciclo sarebbe *chiuso* e l'utente avrebbe a disposizione una playlist che parte da un video e vi ritorna percorrendo solo video tra loro correlati



- **«Tendencies»**: questa funzione permette di vedere quali sono i tag più popolari di sempre effettuando una sommatoria delle visualizzazioni di ciascun video contenente il tag