Sistemi e Architetture per Big Data - A.A. 2018/19
Progetto 2: Analisi dei commenti di articoli pubblicati sul New York Times con Storm/Flink

Docenti: Valeria Cardellini, Fabiana Rossi
Dipartimento di Ingegneria Civile e Ingegneria Informatica
Università degli Studi di Roma "Tor Vergata"

Requisiti del progetto

Lo scopo del progetto è rispondere ad alcune query riguardanti un dataset relativo ai commenti di articoli pubblicati sul New York Times, utilizzando il framework Apache Storm o, in alternativa, Apache Flink.

Il dataset contiene dati relativi ai commenti (diretti e indiretti) di articoli pubblicati sul New York Times dall'1 gennaio 2018 al 18 aprile 2018 ed è composto da un unico flusso di input, fornito come file di testo in formato CSV e disponibile all'indirizzo `http://www.ce.uniroma2.it/courses/sabd1819/projects/prj2_dataset.tgz`

Il flusso di input, contenuto nel file `comments.csv`, contiene diverse informazioni riguardanti i commenti; in particolare, ogni riga del file ha il seguente formato:

```
approveDate, articleID, articleWordCount, commentID, commentType,
createDate, depth, editorsSelection, inReplyTo, parentUserDisplayName,
recommendations, sectionName, userDisplayName, userID, userLocation
```

dove:

- `approveDate` è il timestamp che indica quando il commento è stato approvato dalla redazione del New York Times; il timestamp è espresso in Unix time (offset in secondi rispetto alla mezzanotte (UTC) dell'1 gennaio 1970).
- `articleID` è l'identificatore dell'articolo a cui il commento si riferisce (stringa alfanumerica);
- `articleWordCount` è il numero di parole dell'articolo a cui il commento si riferisce (intero senza segno);
- `commentID` è l'identificatore del commento (intero senza segno);
- `commentType` è il tipo di commento (`comment` oppure `userReply`); `comment` indica un commento diretto, mentre `userReply` indica la risposta di un utente (o commento indiretto)
- `createDate` è il timestamp che indica quando il commento è stato inserito dall'utente; il timestamp è espresso in Unix time (offset in secondi rispetto alla mezzanotte (UTC) dell'1 gennaio 1970).
- `depth` è la profondità del commento (1 se commento diretto, 2 o 3 se commento indiretto);

- `editorsSelection` indica se il commento è stato selezionato o meno dagli editor (booleano);
- `inReplyTo` è l'identificatore del commento che viene commentato (intero senza segno), il campo è vuoto se la riga rappresenta un commento diretto;
- `parentUserDisplayName` è il nickname dell'utente che ha inserito il commento rispetto al quale la riga rappresenta un commento indiretto, il campo è vuoto se la riga rappresenta un commento diretto;
- `recommendations` è il numero di like ricevuti dal commento;
- `sectionName` è la sezione del giornale in cui è stato pubblicato l'articolo;
- `userDisplayName` è il nickname dell'utente che ha inserito il commento (stringa alfanumerica);
- `userID` è l'identificatore dell'utente che ha inserito il commento (intero senza segno);
- `userLocation` è l'ubicazione dell'utente che ha inserito il commento (stringa alfanumerica).

Gli eventi sono ordinati in base al timestamp di creazione (`createDate`).

Il progetto è dimensionato per un gruppo composto da **2 studenti**; per gruppi composti da 1 oppure 3 studenti, si vedano le indicazioni specifiche. Supponendo di effettuare il replay del dataset (accelerando la scala temporale), si chiede di rispondere alla seguenti query in tempo reale:

1. Fornire la classifica aggiornata in tempo reale dei 3 articoli più popolari, ovvero che hanno ricevuto il maggior numero di commenti (sia diretti, sia indiretti). L'output della classifica ha il seguente schema:

   ```
   ts, artID_1, nCmnt_1, artID_2, nCmnt_2, artID_3, nCmnt_3
   ```

   dove

   ```
   ts                  // timestamp di inizio statistica
   artID_1          // id dell'articolo classificato primo
   nCmnt_1   // numero di commenti dell'articolo classificato primo
   ...
   artID_3       // id dell'articolo classificato terzo
   nCmnt_3 // numero di commenti dell'articolo classificato terzo
   ```

   La classifica dovrà essere calcolata sulle finestre temporali:

   - 1 ora (di event time),
   - 24 ore (di event time),
   - 7 giorni (di event time).

2. Ottenere informazioni sulla fascia oraria in cui vengono inseriti commenti. In particolare, l'output della query riporta il numero complessivo di commenti diretti che vengono inseriti nell'arco temporale di 2 ore, secondo lo schema:

   ```
   ts, count_h00, count_h02, ..., count_h20, count_h22
   ```

   dove

```
ts          // timestamp di inizio statistica
count_h00   // numero di commenti diretti inseriti da 00:00:00 a 01:59:59
count_h02   // numero di commenti diretti inseriti da 02:00:00 a 03:59:59
...
count_h22   // numero di commenti diretti inseriti da 22:00:00 a 23:59:59
```

Tali statistiche aggregate dovranno essere calcolate sulle finestre temporali:

- 24 ore (di event time),
- 7 giorni (di event time),
- 1 mese (di event time).

3. Fornire la classifica aggiornata in tempo reale dei 10 utenti più popolari. Un utente viene definito popolare in base ai like che ricevono i commenti diretti da lui inseriti ed al numero di discussioni che è in grado di generare. Il grado di popolarità di un utente viene quindi definito come $w_a a + w_b b$ dove a è il numero di like ricevuto dai suoi commenti diretti, b il numero di commenti indiretti collegati ai suoi commenti e $w_a = 0.3, w_b = 0.7$. Se un commento diretto è stato selezionato dagli editor, il numero di like deve essere incrementato del 10%.

   L'output della classifica ha il seguente schema:

   ```
   ts, user_1, rating_1, user_2, rating_2, ..., user_10, rating_10
   ```

   dove

   ```
   ts          // timestamp di inizio statistica
   user_1      // id dell'utente classificato primo
   rating_1    // punteggio complessivo dell'utente classificato primo
   ...
   user_10     // id dell'utente classificato decimo
   rating_10   // punteggio complessivo dell'utente classificato decimo
   ```

   La classifica dovrà essere calcolata sulle finestre temporali:

   - 24 ore (di event time),
   - 7 giorni (di event time),
   - 1 mese (di event time).

Gli output delle query devono anche essere memorizzati in opportuni file e consegnati.

Si chiede inoltre di valutare sperimentalmente i tempi di latenza ed il throughput delle tre query durante il processamento sulla piattaforma di riferimento usata per la realizzazione del progetto, riportando tali tempi nella presentazione (e nell'eventuale relazione). La piattaforma di data stream processing può essere un nodo standalone con Apache Flink o Apache Storm oppure in alternativa è possibile utilizzare un servizio Cloud per Hadoop (ad es. Amazon EMR o Google Dataflow), usando i rispettivi grant a disposizione.

Opzionale: Rispondere ad una query a scelta tra le tre sopra descritte usando Kafka Streams oppure Spark Streaming e confrontare, sulla stessa piattaforma di riferimento, le prestazioni in termini di tempo di latenza e throughput delle query ottenute dai due framework.

Per gruppi composti da 1 studente: si richiede di rispondere alle query 1 e 2.

Per gruppi composti da 3 studenti: in aggiunta ai requisiti sopra elencati, si richiede di utilizzare Kafka Streams oppure Spark Streaming per rispondere alle tre query e di confrontare, sulla stessa piattaforma di riferimento, le prestazioni in termini di latenza e throughput con quelle ottenute dal primo framework scelto.

Svolgimento e consegna del progetto

Comunicare alle docenti la composizione del gruppo entro **lunedì 24 giugno 2019**.

Per ogni comunicazione via email è necessario specificare *[SABD]* nell'oggetto (subject) dell'email. Il progetto è valido **solo** per l'A.A. 2018/19 e deve essere consegnato **entro venerdì 12 luglio 2019** per poter raggiungere il punteggio massimo.

La consegna del progetto consiste in:

1. link a spazio di Cloud storage o repository contenente il codice del progetto;
2. lucidi della presentazione orale, da inviare via email alle docenti *dopo* lo svolgimento della presentazione.
3. *opzionale*: relazione di lunghezza compresa tra le 4 e le 6 pagine, usando il formato ACM proceedings (`https://www.acm.org/publications/proceedings-template`) oppure il formato IEEE proceedings (`https://www.ieee.org/conferences_events/conferences/publishing/templates.html`).

Le presentazioni si terrano **lunedì 15 luglio e martedì 16 luglio 2019**; ciascun gruppo avrà a disposizione **massimo 15 minuti**.

Valutazione del progetto

I principali criteri di valutazione del progetto saranno:

1. rispondenza ai requisiti;
2. originalità;
3. architettura del sistema e deployment;
4. organizzazione del codice;
5. efficienza;
6. organizzazione, chiarezza e rispetto dei tempi della presentazione orale.