CS523-BDT-Final Project on SOCCOR ANALYSER

SUBMITTED BY:

PUSKAR BUDHATHOKI, MHRETEAB ADHANOM BERHE , NATNAEL BERHE

SUBMITTED TO: PROF. MRUDULA MUKADAM

Project Overview

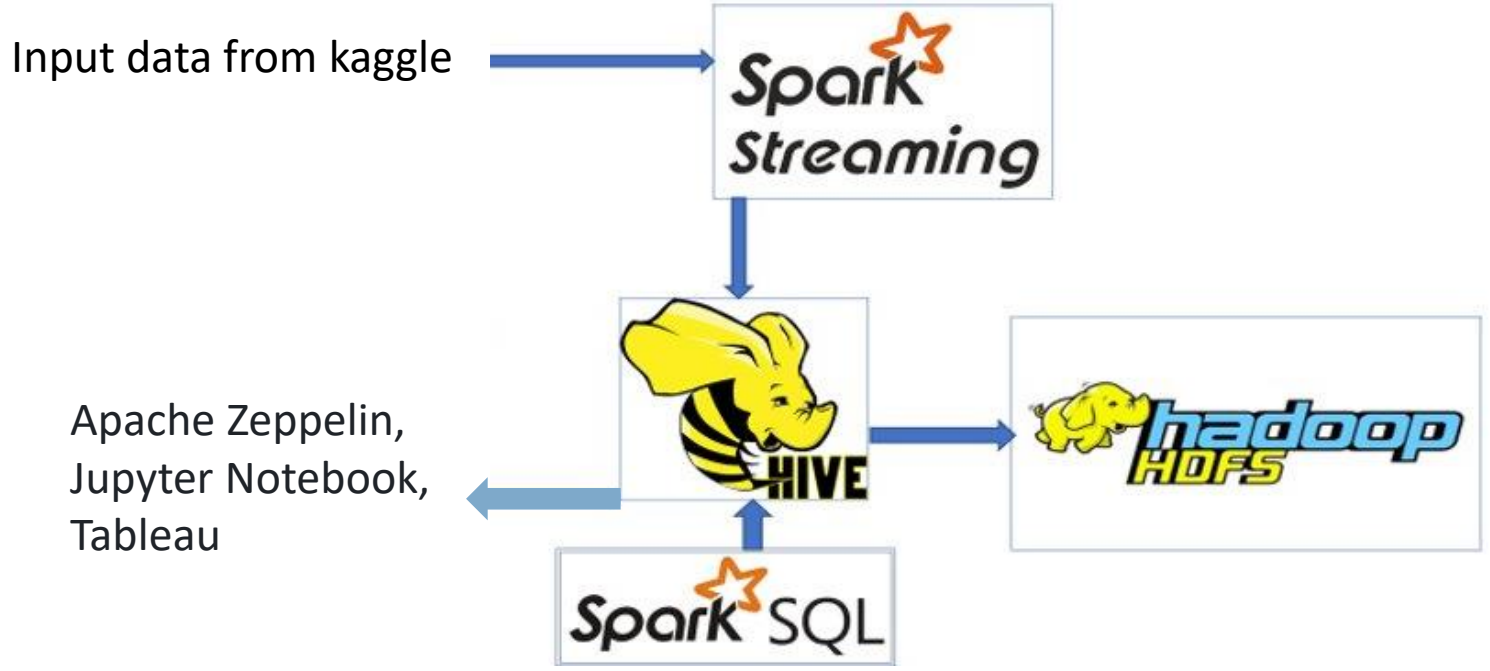
Dataset: we have used International Football Results from 1872 to 2023 dataset from popular website kaggle (https://www.kaggle.com/datasets/martj42/international-football-results-from-1872-to-2017?datasetId=4305)

Kafka: Distributed messaging system which is used for streaming analytics and data integration(Demo Project)

Spark: It is used for getting the dataset and performing some filter and aggregation

Hive/Hbase: The enriched result is saved in Hbase/Hive.

Project Overview

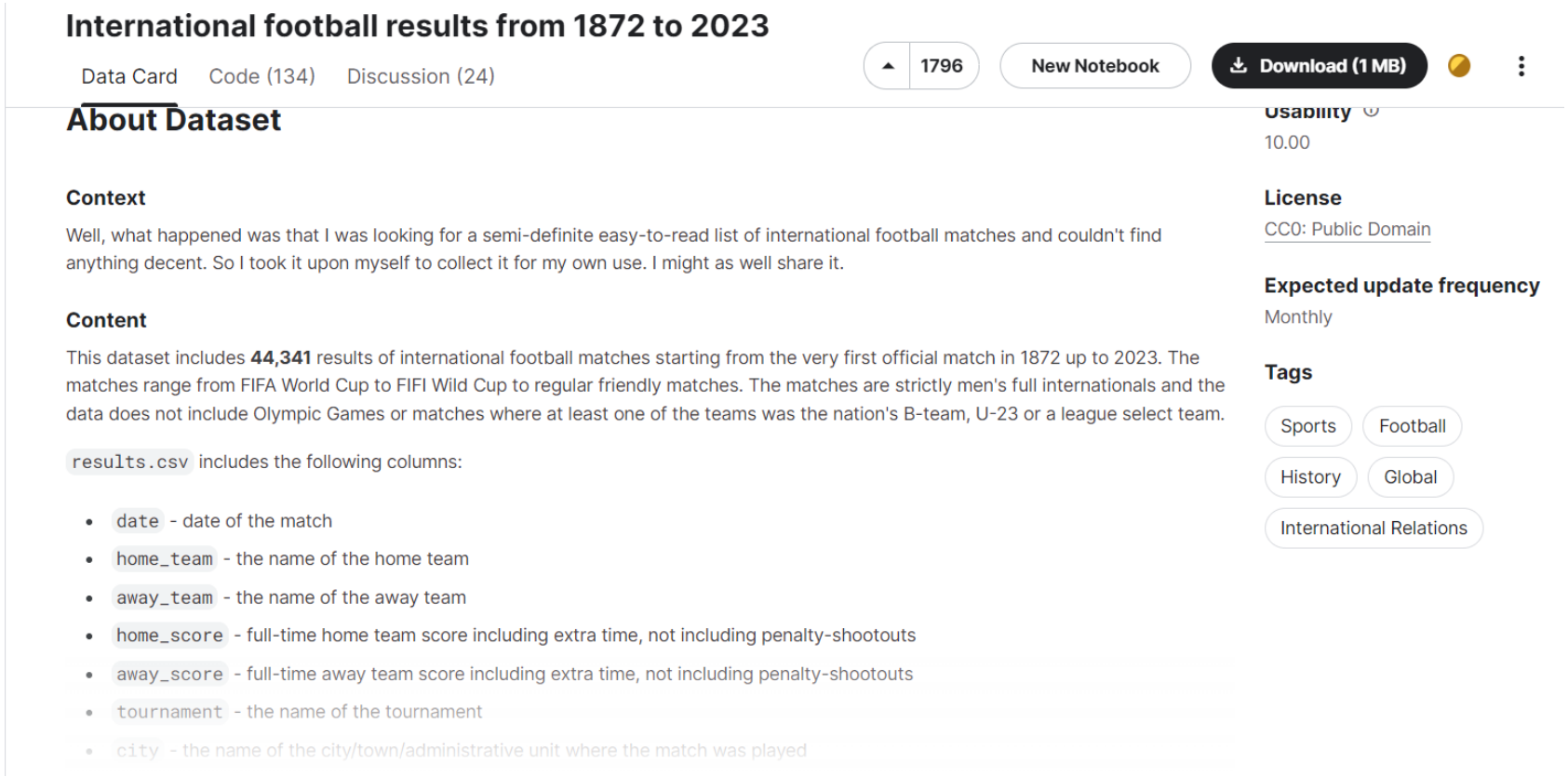


Technology stack used

- Kafka – xxx
- Spark streaming – 2.4.4
- Spark core – 2.4.4
- Java - 8
- Hadoop – 2.6.0
- Hbase/Hive – xxx

Dataset used

- Used static football dataset from kaggle
- https://www.kaggle.com/datasets/martj42/international-football-results-from-1872-to-2017?datasetId=4305
- Created a program to split the whole dataset into smaller datset to use apache data streaming



International football results from 1872 to 2023

Data Card | Code (134) | Discussion (24)

1796 | New Notebook | Download (1 MB)

About Dataset

Context

Well, what happened was that I was looking for a semi-definite easy-to-read list of international football matches and couldn't find anything decent. So I took it upon myself to collect it for my own use. I might as well share it.

Content

This dataset includes **44,341** results of international football matches starting from the very first official match in 1872 up to 2023. The matches range from FIFA World Cup to FIFI Wild Cup to regular friendly matches. The matches are strictly men's full internationals and the data does not include Olympic Games or matches where at least one of the teams was the nation's B-team, U-23 or a league select team.

`results.csv` includes the following columns:

- `date` - date of the match
- `home_team` - the name of the home team
- `away_team` - the name of the away team
- `home_score` - full-time home team score including extra time, not including penalty-shootouts
- `away_score` - full-time away team score including extra time, not including penalty-shootouts
- `tournament` - the name of the tournament
- `city` - the name of the city/town/administrative unit where the match was played

Usability
10.00

License
CC0: Public Domain

Expected update frequency
Monthly

Tags
Sports | Football | History | Global | International Relations

Apache Data Streaming sample

- Received csv file as text stream input and allocate infrastructure to complete the allocated jobs

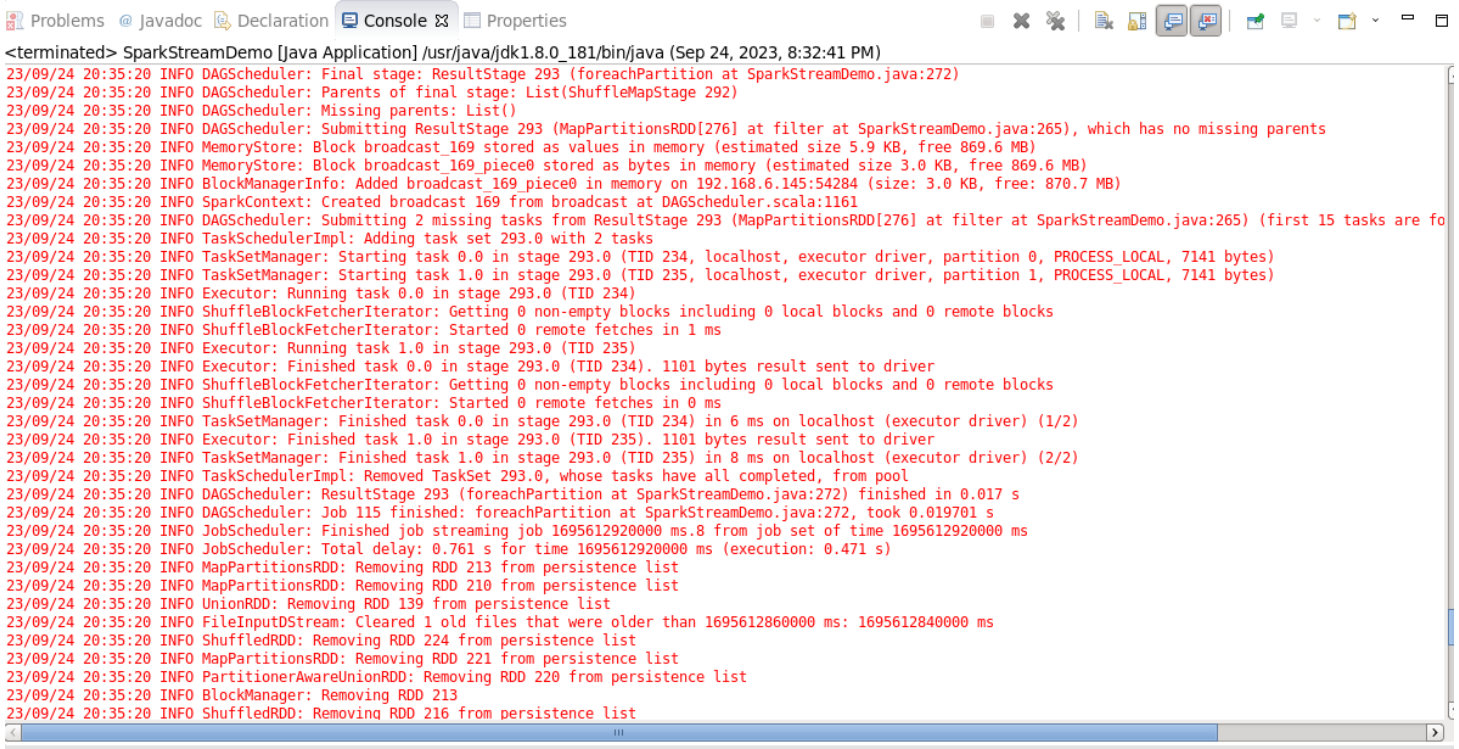


Fig 2: Apache data stream console

Apache Kafka

- Consist xxx brokers (Kafka cluster) and manage by zookeeper
- Detail description and screenshot goes here

Spark Streaming

- Get international football statistics from the split dataset
- Filter required columns like home_team, away_team, home_goal, away_goal, city, match_type etc
- Calculate total scores by each country, frequency of match type, organizing city with frequency
- Filter data based on threshold values for different use cases

Apache HBASE/Hive

- The output data from spark (aggregated) is persist in Hbase/Hive database
- Detail description and screenshot goes here

Project execution steps

Step 1(Data preparation):

- Created a account in kaggle (https://www.kaggle.com/)
- Downloaded static dataset related to international football
- split the dataset into smaller files so that we can apply data streaming and processing operations

Step 2(Data feed to Apache Spark Data Streaming):

- The pipeline is written in shellscript iwhich fetch data from local directory in cloudera VM to the Apache Data Streaming in certain interval of time

Step 3(Real-time Analysis with Apache Spark Streaming):

- Ingest data in every 20 seconds.
- Text file streaming is used to load data.
- Different stream function like filter, maptoPair, Union, reducebykey etc are used.
- complex data structure with multiple Tuple are used to get the aggregate data from input data.

Project execution steps

Step 4(Data Processing with Apache Hive):

- Schema Definition: Defined a Hive schema that matches the structure of the output result from Apache data streaming
- Data Loading: Load the CSV data into a Hive table using Hive's LOAD DATA command or other methods for bulk data loading.
- Data Transformation: Use Hive's SQL-like language, HiveQL, to perform various data transformations and aggregations. we calculated statistics, filter data by date, tournament, or teams, and perform other necessary preprocessing.
- Querying: We ran Hive queries to extract insights from the data. For example, we found the countries with the most international goals, city and country organizing most games, the pattern of football match in international tournament, and more.

Demo steps (Demo project related)

- Please add details here

Future tasks

Output and Visualization:

- We planned to do visualization of some statistics.
- However, due to time constraints, we were not able to do it.
- In future, We plan to visualize the real-time data and insights using data visualization tools like Apache Zeppelin, Jupyter Notebook, or Tableau(connect with Cloudera Hadoop using ODBC client and fetching data from the Hive Tables).
- This can help us provide live updates to football enthusiasts.

Thank You!