
A REVIEW OF DATA ANALYSIS AND VISUALIZATION OF OLYMPICS USING PYSARK AND DASH-PLOTLY

**Harshal S. Kudale*¹, Mihir V. Phadnis*², Pooja J. Chittar*³,
Kalpesh P. Zarkar*⁴, Prof. Balaji K. Bodhke*⁵**

*^{1,2,3,4}UG Scholar, Department Of Computer Engineering, Modern Education Society's College Of Engineering, Pune, Maharashtra, India.

*⁵Professor, Department Of Computer Engineering, Modern Education Society's College Of Engineering, Pune, Maharashtra, India.

ABSTRACT

Big data Analytics is becoming increasingly popular in our day to day life. Everyday, tons of data is generated and analyzed using big data tools to extract useful and meaningful information. This study aims to analyze and visualize the Modern Olympic Games from 1896 to present. This project shows Analysis of Medals and Athletes participated in Modern Olympics for 120 Years and gives insight on the same. Apache Spark is arguably the most popular Data Analytics framework, with multiple supporting libraries like SparkSQL, Spark Streaming, GraphX, MLlib etc. this experiment uses Python API for Spark that is, PySpark. The Visualization of data is done using Plotly-dash library and its tools for Interactive Visualisation.

Keywords: Big Data, PySpark, Plotly-Dash, Pandas, Olympics.

I. INTRODUCTION

Big data Analytics is a process of handling and processing huge volumes of data. Nowadays, about 2 Quintillion Bytes of data is generated everyday from the likes of Websites, IOT Devices, Transactions, Logs, Social media etc. Thus, Generates huge need to manage, process this data and extract useful information to gain insights. To manage these kinds of data, we need efficient, robust tools and Algorithms to handle it properly with minimum loss of Information. The Olympics is one of the most Prestigious International sport events in the world. Athletes and People participate from every corner of the world, with only one goal in mind, Winning. There are Tons of events managed by the Olympics association covering various Sports categories like Athletics, Racket Sports, Water Sports, Martial Arts, Timed events, Group Events etc. hence they generate high Amount of data from events, needs to be processed to find out becomes the winner and takes gold medal. Such data generated is also analyzed real time to decide the winner considering the millimeter gap in competitiveness. The winners in Participated Athletes get their Name written in History for their achievement.

Here the historic data comes in mind to process and analyze to get interesting Insights on progress of athletes. More accurate analysis gets better results in decision making.

In this paper we will get a brief overview of data analysis done on the historic Olympics data set to get some interesting Information from it, and it is visualized to get more interactive output.

For Compatibility and Easy to use reasons we have choose Apache Spark framework to do the data Analytics work, since it is about 100 times faster than the Apache Hadoop in in-memory data processing, and has wide variety of language API support like Java, Python, Scala, Ruby, R etc. whereas Hadoop mostly uses Java as default language.

For this Project we have chosen Python API for Apache Spark ie. PySpark,[12] which Comes with Amazing Ecosystem of Analytical tools like SparkSQL for Query handling, Spark MLlib for Machine Learning, Spark GraphX for graph processing etc. For the visualization part, we are using Dash-Plotly [13] library to Create high quality, Interactive web apps.

In this Survey Paper we will be going through the previous work done using Apache Spark, Big data analytics and visualization and relevant topics.

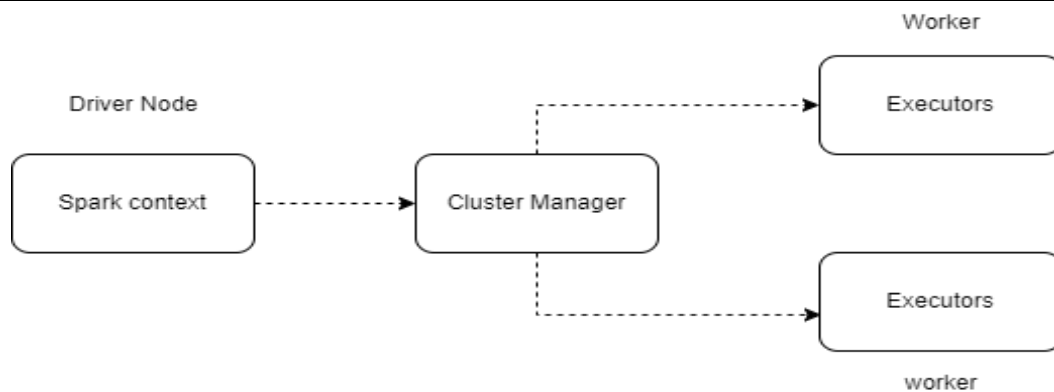


Figure 1: Apache Spark Architecture

II. LITERATURE SURVEY

Kavita S., Ranjana B.[1] have created a Hybrid recommendation System using Apache Spark, they have used Collaborative and Content Based Filtering Approach to create a Movie recommendation website. Their Recommendation model is based on MovieLens 100K dataset and created and trained using Spark MLLib libraries.

HooYoung Ahn, Hyunjae Kim, Woongshik You [2] Have Performance tested Apache Spark framework on a Framework level using YARN, and HiBench Suite. There they have tested and measured different usage Scenarios depending on the execution time, throughput, CPU usage, memory usage, and disk throughput of various workloads. They also have Applied and tested Performance of Spark in Fully Distributed Mode for Large Scale Datasets and clusters.

Jia Yu 1, Mohamed Sarwat [3] have created a tutorial on Geospatial Data management in Apache Spark. They have tested Apache spark on large Scale Spatial-temporal data, streaming data. they have aimed to bridge gap between Distributed computation in spark and Spatial data analysis. They aim to develop further research in Spatial Data management Systems.

Muhammad Junaid, Shiraz Ali Wagan, Nawab Muhammad Faseeh Qureshi, Choon Sung Nam, Dong Ryeol Shin [4] have shown use case of Spark MLLib, A Machine Learning Library in Apache Spark ecosystem, by applying Machine learning on UCI data set. In the paper they have given overview of Apache Spark ecosystem in Machine Learning Perspective, describing various tools in Ecosystem Such as MLLib, SparkSQL, SparkStraming, GraphX etc.

Eman Shaikh, Iman Mohiuddin, Yasmeen Alufaisan, Irum Nahvi [5] have also given an overview on Whole Apache Spark Ecosystem. They have described the Multi Processing and batch processing Capabilities of Spark , How Spark is Better than Hadoop. In the paper they have given an overview of Core Architecture and Data processing in Spark, also shown use cases and applications where Apache spark can be useful.

Deepika D Mishra, Salim Pathan, CSRC Murthy [6] Have created a Analysis system for Squid Proxy Logs using Apache Spark. They have used realtive technologies like Apache Hadoop HDFS for storage and Apache Zepplin for result visualization. They have tested Saprk performance using changing in data volumes and parameters like number of executors, number of executor cores and executor memory. They generated statistics like top domains accessed, top users etc by querying these logs which helped in understanding traffic characteristics within organization and also detect threats.

Silvina Ca´mo-Lores, Jes´us Carretero, Bogdan Nicolae, Orcun Yildiz, and Tom Peterka [7] have created a Framework called Spark-DIY to create a Parallelism and High Performance Computing based framework for Integrating High performance computing with a batch processing model. This lets user to create high Performance applications without Compromising on Scalablity, frmo the results, This framework shows good performance and scalability for communication-intensive operations in comparison to Spark, and enables the integration of elements from both the Big Data and HPC ecosystems for applications with diverse requirements without sacrificing productivity.

Yogesh Kumar Gupta, Nidhi Sharma [8] have created a paper to showcase Comparison between Apache Spark and Apache Hadoop’s performance capabilities. They have applied these framework processing on Stock market Nifty 50 data and Covid 19 data set. They have also Collected all the Apache Hadoop and Spark’s Literature reviews in One paper.

Jiwo Bang, Mi-Jung Choi [9] have performed a Benchmark test in Native and Docker Environment for Apache Spark, and Apache Storm. They conducted a test of performing real time Processing of JSON data in native and Docker environment, in single user and batch processing mode. In the end, only with 2-10% performance deficit, Docker is worthy option for testing than directly applying on native hardware.

Anveshrihaa S, Lavanya [10] performed a real time traffic analysis using Short term memory networks in Apache spark. They Developed an efficient data processing system using Apache Spark, Kafka and Hadoop. Using Apache Spark, a traffic forecast model that leverages a deep learning model called Long- Short Term Memory network for predicting the flow of traffic on roads in real-time is implemented. With the use of Apache Kafka and Spark streaming, real-time predictive analysis of the traffic data was carried out and good performance of the model was observed in analyzing the data and predicting the flow of Vehicular traffic.

III. UML DIAGRAMS

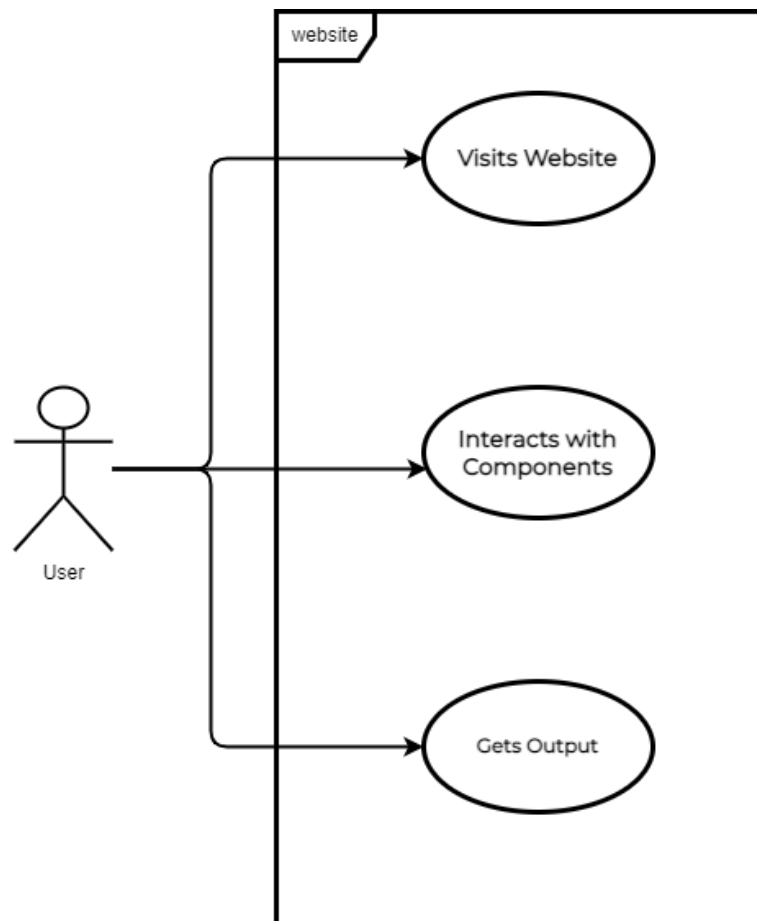


Figure 2: Use Case Diagram

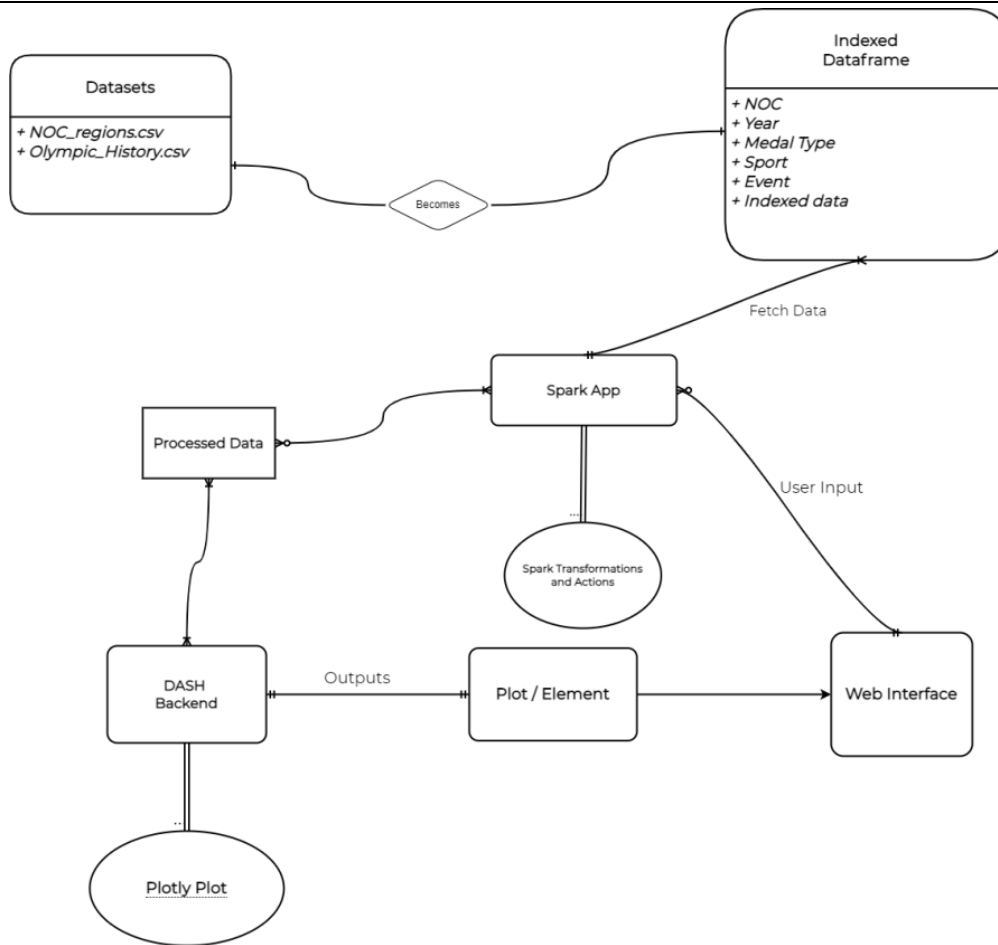


Figure 3: Entity Relationship Diagram

IV. TECHNOLOGIES OVERVIEW

A. PySpark:

PySpark is the Python API for Apache Spark, an open source, distributed computing framework and set of libraries for real-time, large-scale data processing. If you're already familiar with Python and libraries such as Pandas, then PySpark is a good language to learn to create more scalable analysis and pipelines. The key data type used in PySpark is the Spark dataframe. This object can be thought of as a table distributed across a cluster, and has functionality that is similar to dataframes in R and Pandas. If you want to do distributed computation using PySpark, then you'll need to perform operations on Spark dataframes and not other Python data types.

B. PySpark Components:

PySparkSQL - A PySpark library to apply SQL-like analysis on a huge amount of structured or semi-structured data. You can also use SQL queries with PySparkSQL.

MLlib - A wrapper over PySpark and Spark's machine learning (ML) library. MLlib supports many machine learning algorithms for classification, regression, clustering, collaborative filtering, dimensionality reduction, and underlying optimization primitives.

GraphFrames - A graph processing library that provides a set of APIs for performing graph analysis efficiently, using the PySpark core and PySparkSQL. It is optimized for fast distributed computing.

C. Dash-Plotly:

Dash is the original low-code framework for rapidly building data apps in python, R, Julia, and F#(experimental). It is open source python library released under permissive MIT license. It is ideal for building and deploying data apps with customize user interface. It is enough that you can bind a user interface to your code easily. It is python library for creating reactive, web-based applications.

Dash is a user interface library for creating analytical web applications. Those who use Python for data analysis, data exploration, visualization, modelling, instrument control, and reporting will find immediate use for Dash. It abstracts away all of the technologies and protocols that are required to build a full-stack web app with interactive data visualization. Dash makes it dead-simple to build a GUI around your data analysis code. Plotly develops Dash and also offers a platform for writing and deploying Dash apps in enterprise environment.

V. CONCLUSION

This Survey Paper Includes all the documentation required and prerequisites for understanding project. With help of the Survey we will be able to perform the analysis and visualization of Olympics using Spark. With the help of The architecture, we can also create an web app for another sports tournaments like Sports websites for Cricket, Football Premier League etc. Using the Processing Libraries like PySpark, Data analysis has become more fast and feasible to even small developers. In the proposed Project, the analysis will be done using Popular PySpark framework, with very useful ecosystem Libraries like SparkSQL and MLlib. We can effectively process data real time using these libraries to get thorough insights. Using Dash we can create fast and beautiful web apps in no time, and with the help of Plotly library we can convert this processed data into charts, graphs, diagrams etc.

VI. REFERENCES

- [1] Kavitha S, Ranjana Rajesh Badre, "Towards a Hybrid Recommendation System On Apache Spark", 2020 IEEE India Council International Subsections Conference (INDISCON).
- [2] HooYoung Ahn, Hyunjae Kim, Woongshik You "Performance study of Spark on YARN cluster using HiBench", 2018 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia).
- [3] Jia Yu 1, Mohamed Sarwat, "Geospatial Data Management in Apache Spark:A Tutorial", 2019 IEEE 35th International Conference on Data Engineering (ICDE).
- [4] Muhammad Junaid, Shiraz Ali Wagan, Nawab Muhammad Faseeh Qureshi, Choon Sung Nam, Dong Ryeol Shin, "Big data Predictive Analytics for Apache Spark using Machine Learning", 2020 Global Conference on Wireless and Optical Technologies (GCWOT).
- [5] Eman Shaikh, Iman Mohiuddin, Yasmeen Alufaisan, Irum Nahvi, "Apache Spark: A Big Data Processing Engine", 2019 2nd IEEE Middle East and North Africa COMMUNICATIONS Conference (MENACOMM).
- [6] Deepika D Mishra, Salim Pathan, CSRC Murthy, "Apache Spark Based Analytics of Squid Proxy Logs", 2018 IEEE International Conference on advance networks and telecommunication systems.
- [7] Silvina Ca'ino-Lores, Jes'us Carretero, Bogdan Nicolae, Orcun Yildiz, and Tom Peterka, "Spark-DIY: A Framework for Interoperable Spark Operations with High Performance Block-Based Data Models", 2018 IEEE/ACM 5th International Conference on Big Data Computing Applications and Technologies (BDCAT)
- [8] Yogesh Kumar Gupta, Nidhi Sharma, "Propositional Aspect between Apache Spark and Hadoop Map-Reduce for Stock Market Data", Proceedings of the Third International Conference on Intelligent Sustainable Systems [ICISS 2020]
- [9] Jiwon Bang, Mi-Jung Choi, " Docker environment based Apache Storm and Spark Benchmark Test", KICS 2020.
- [10] Anveshrihaa S, Lavanya K, "Real-Time Vehicle Traffic Analysis using Long Short Term Memory Networks in Apache Spark", 2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)
- [11] Kaggle Data set Link
- [12] PySpark Documentation
- [13] Dash Documentation