Team -06: DATA ENCODERS

Version-01

Dataset from the Website

https://m.cricbuzz.com/

Excel Sheet

(For importing the data)

Unstructured

daa

Python for further conversions in the dataset to visualize

Power BI for visualization

Structured data

SQL or Excel (yet to be decided)

**Work Division:**

1. Implementation of Python : Raghunandan
2. Data prep : Deepthi and Nikitha Kethireddy
3. Visualization : Jeevan , Dheeraj

Version-02

Using a package called (pycricbuzz) for generating the live data

A new file with data is generated in the form of JSON file.

We are using python to append the data to the generated file every time the data is updated in the website

Whenever we update the file in power BI, the graphs/bars etc will be updated dynamically.

We use that generated file in Power BI and we have an option to update that file by setting time

**Work Division:**

1. Implementation of Python : Raghunandan
2. Data prep : Deepthi and Nikitha Kethireddy
3. Visualization : Jeevan , Dheeraj

Version-03

Using a package called (pycricbuzz) for generating the live data

We are using python to get the data onto the local system every time the data is updated in the website.

Data is appended to MongoDB as we get live score from the package.

Reports in Power BI will be refreshed as we update the data in MongoDB

We connect to MongoDB to fetch the data and will use Power BI to generate the graphs

**Work Division:**

1. Implementation of Python: Raghunandan
2. Data prep: Deepthi and Nikitha Kethireddy
3. Visualization: Jeevan
4. Commentary Sentiment Analysis: Dheeraj

Version-04

Using a package called (pycricbuzz) for generating the live data

We are using python to get the data onto the local system every time the data is updated in the website.

Data will be in format of JSON and this will be sent to KAFKA

JSON

Fetch from KAFKA

Reports in Power BI will be refreshed as we update the data in KAFKA

We connect to KAFKA to Power BI

Reporting

**Work Division:**

1. Implementation of Python: Raghunandan
2. Data prep: Deepthi and Nikitha Kethireddy
3. Visualization: Jeevan
4. Commentary Sentiment Analysis: Dheeraj

Version-05

1. Score Prediction:

Run zookeeper and kafka in the local system.

Open MongoDB and MongoDb Connector to PowerBI

Run the producer (java) and consumer (java with MongoDB) for the customized project.

Store the streamed data in MongoDB

Connect MongoDB to PowerBI

Data Preparations (Perform aggregation and compute data to get score for next over)

Visualize the data using PowerBI

**Data Source:** We are considering data from the web source known as [www.crickbuzz.com](http://www.crickbuzz.com) to visualize and analyze and predict the score for the succeeding over.

**Data Extraction:**

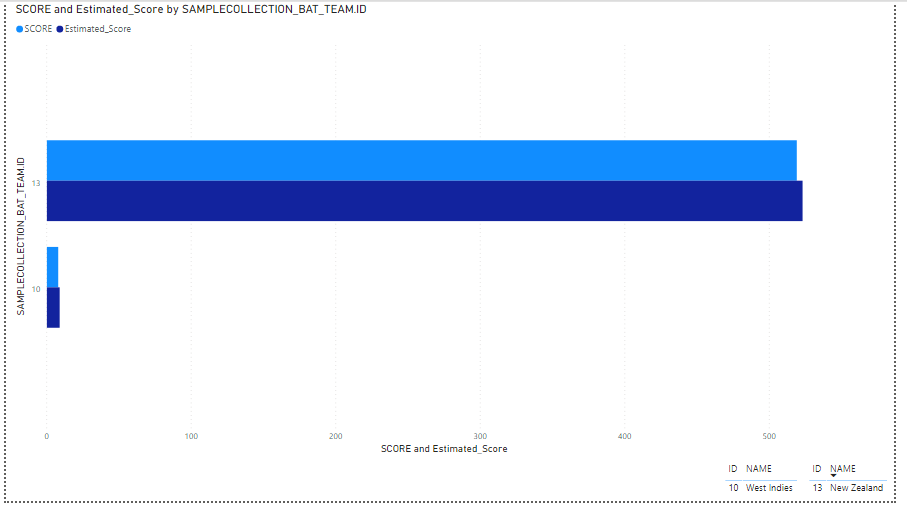
* First we will be running zookeeper and kafka by using the following commands:
  + .\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties
  + .\bin\windows\kafka-server-start.bat .\config\server.properties
* The next process is to open MongoDB which is installed within your local system and MongoDB connector to PowerBI.
* Now run the producer and consumer for the customized project by the name KafkaLiveScoreStream which is present in our Github repository by using IntelliJ IDE.
* Here the producer gets the data from the API of the data source as mentioned above by using Java.
* The reason for using java is because of Serialization and De-serelization of the JSON data is easy.
* And consumer processes the data by using Java and stores the end-result using MongoDB.
* The reason for using MongoDB is that, it’s used to store the streamed data and also it acts as a middleware between Kafka and PowerBI.

**Data Analyzing and transformation:**

* After successful connection establishment from MongoDB to Power BI, we need to open the JSON file (the end result of data obtained by using kafka is stored by using MongoDB).
* Then click on transform data option. A separate file will be opened wherein we will perform the following operations:
  + Firstly we need to perform group-by operation upon the unique id which is generated for every individual teams.
  + Along with the operation specified above, we shall be finding most recent overs and recent score by calculating the maximum of overs and maximum score so as to obtain the most recent overs and runs scored by each team at that point of time.
  + Using score and total number of overs played, we will be run rate of the team. This is done by dividing score with overs. The value thus obtained will be rounded-up.
* By adding the run rate to total score we can obtain the estimated score.
* The estimated score thus obtained will be used to visualize the data accordingly.

**Data Visualizing:**

We will be visualizing the data using Power BI.



The image shown above is the final visualized output for the streaming data for a cricket match.\

**Work Division:**

* Implementation of Kafka: Raghunandan
* Connecting MongoDB to PowerBI : Nikitha Kethireddy
* Data Transformations: Deepthi Chokka
* Data Visualization: Jeevan

1. Commentary Sentiment Analysis:

Import package (pycricbuzz) into jupyter notebook(using python) for generating live commentary

Generate emoji by using NLTK package for the above generated commentaries.

Display end-result in jupyter notebook.

**Work Division:**

* Implementation of Python Programming: Raghunandan
* Analysis of Commentary: Dheeraj

**Data Source:** We are considering data from the web source known as [www.crickbuzz.com](http://www.crickbuzz.com) to visualize and analyze and predict the score for the succeeding over.

**Data Extraction:**

* This website contains an API called as “pycricbuzz” which is exclusively used to extract data using python programming.
* So we will be extracting only the commentary from the web source by using the above mentioned API.

**Data Analyzing:**

Data Analysis is done by using a separate package called as NLTK which is abbreviated as Natural Language Toolkit.

**Data Visualization:**

The data thus obtained will be visualized by using python programming.