

Mini Project 2 - Apache Spark

Overview

League of Legends¹ (abbreviated **LoL** or **League**) is a multiplayer online battle arena video game developed and published by Riot Games. Originally inspired by Valve's Defense of the Ancients (DotA), the game has followed a freemium² model since its release on October 27, 2009. League of Legends is often cited as the world's largest e-sport, with an international competitive scene. The 2019 League of Legends World Championship had over 100 million unique viewers, peaking at a concurrent viewership of 44 million, with a minimum prize pool of US\$2.5 million. Riot Games' 2019 revenue has been recorded at slightly more than 2 billion dollars.

As stated above, League of Legends is a free-to-play, free-to-win game, i.e. paying money won't affect your competitive skills or increase your chances of winning the game. The available purchases are only aesthetic changes and doesn't change the game at all. Riot Games' success can be attributed to the continuous analytical analysis of the player bases to update and refresh the game to be more engaging and addictive.

As in other multiplayer online battle arena (MOBA) games, each player in League of Legends controls a character ("champion") with a set of unique abilities. Most games involve two teams of five players, with each player using a different champion. The two teams compete to be the first to destroy the Nexus structure within the opposing base.

Project Goal

Inspired by Riot Games' success, this project tries to simulate the lifecycle of a real-life analytics project. The project aims to deliver useful information to players or developers that will help make profitable business decisions.³

Data

You are given the online API of Riot Games to use to deliver meaningful, and realistic business insights that will be translated to real-life business decisions. You do not need to run your code to use the online form directly, you can create your own "cache" to run the code off. The online API tests your ability to extract, clean, and summarize real-life data to fit your business needs.

¹ "League of Legends Wiki - Fandom."

https://leagueoflegends.fandom.com/wiki/League_of_Legends_Wiki

² "/dev: On League's Business Model – League of Legends." 25 Jan. 2017,

<https://nexus.leagueoflegends.com/en-us/2017/01/dev-on-leagues-business-model/>

³ "U GG: The Best League of Legends Builds Guides."

<https://u.gg/>

Riot's API

Riot Games has match information publicly available through their online [API](#). It has docs to explain each endpoint, and how to use it. Please be aware of the rate limits which can be found [here](#).⁴

Deliverables

1. [10%] **Proposal**
2. [85%] **Requirements**
 - a. [20%] **Data Preparation** Write “CLEAN”, well-documented **CODE** to collect and cleanse the data to prepare it for the next stages.
(Matches collected must be SOLO/DUO RANKED matches only. min is 75,000 recommended is 100,000)
 - b. [45%] **Data Analysis** Write “CLEAN”, well-documented **CODE** & produce indicative **VISUALISATIONS** for the following:
 - i. [3%] Champion win, pick, and ban rates (bonus if more than 1 patch)
 - ii. [3%] Champion Synergies or duos (bonus if more than 1 patch)
 - iii. [3%] Item win, pick rates (bonus if more than 1 patch)
 - iv. [6%] Item Synergies (item with champion, item with class) (bonus if more than 1 patch)
 - v. [15%] Item suggestion (has to be for at least 2 champions from different classes) (bonus if more than 1 patch, or live in-game suggestion)
 - vi. [10%] **More requirements** to get more insights, information and/or suggestions for questions you are interested in. This is completely up to the students.
(Must be approved by the TAs in the proposal.)
 - vii. [5%] Script to setup/run the pipeline above to be run on a Spark cluster.
 - c. [10%] Write a “CLEAN”, organized **“TECHNICAL” document** detailing the process needed to reach the results with focus on these points
 - i. Data analysis (data problems, patterns, noise, outliers)
 - ii. Challenges faced & how they were solved
 - iii. Optimizations
 - iv. Final design of the code detailing each part of the pipeline
 - v. Any approaches should be written in this document
 - d. [10%] Write a “CLEAN”, organized **“BUSINESS” document** that is to be delivered to a business user with charts and visualizations.

⁴ "Riot API Libraries - Read the Docs."
<https://riot-api-libraries.readthedocs.io/en/latest/>

3. [5%] **Performance** is quite important for business needs. Delivering information as soon as requested is considered a must at times. To simulate this, a percentage of the rubric is set for performance. The fastest team will get the full percentage, the slowest team will get none of the percentage. The rest of the teams will be distributed accordingly
4. [5% bonus] **Discussions** & Executions with the following process:
 - a. Do experiments, trial runs, and/or section runs
 - b. Discuss findings
5. [5% bonus] Useful interactions with TA (questions, suggestions, etc.)
6. [5% bonus] Finding TA's summoner's name (This is for ALL teams, i.e. the 5% will be distributed to teams who find the summoner's name)
7. [-5%] per instruction broken, for any submission/delivery criteria not followed
8. AN INSTANT ZERO (if you are lucky) for plagiarism/cheating/copying for all students involved

Restrictions

1. You have to use Apache Spark
2. NO restriction on language
3. NO restriction on utility libraries
4. NO restriction on size or architecture of the code, however, efficiency is will be taken into account

Delivering Milestones

1. Team assignment (teams of 2)
2. First approach discussion with proposal document
3. Final delivery & presentation