Hackathon Big Data 2015

Data mining on League of Legends Data sets

The DMD Team
Driss El Makhzoumi
Lamya Kejji
Alexandre Rodenas
Clémentine Szalavecz

Coach
Chedy Raissi

League Of Legends

- 3D multiplayer online battle arena
  - Real-time strategy video game
  - Developed and published by Riot Games
  - Free-to-play game

- Rules
  - Player 1 bans a hero — Player 2 bans another hero (x 3)
  - Player 1 picks a hero — Player 2 picks 2 other heroes
  - player 1 picks 2 more — Player 2 picks 2 more
  - player 1 picks 2 more — Player 2 picks 1 more

624$ million

Second for free-to-play games when it comes to revenue in 2013

67 million

Number of players every month in 2014

27 million

Number of players per day in 2014

Why such a study ?

- Few data mining has been done on this kind of games
  - Get progressive statistics

- Players within a team take part in Million dollar competitions
  - as they pick or ban a player a rate of winning is displayed

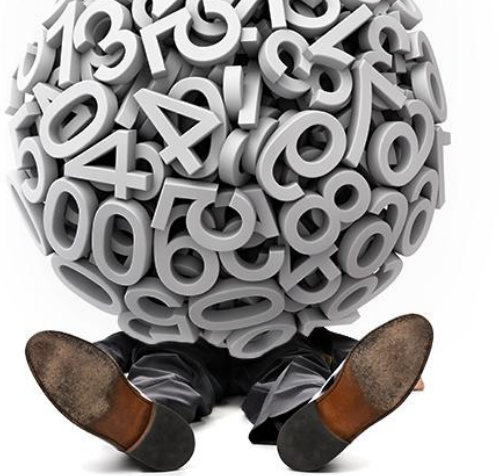
- Ensure fair competition by balancing the game
  - Give the balance for each combination

Departure Goal of the project

- Back-end (PHP-JAVA)
    - Compute statistics on the success of a team as they choose heroes to pick or ban.
    - Compute best combinations of upcoming choices of heroes to pick or ban.
- Front-end (JS-HTML-CSS-AJAX)
    - Develop a web plateforme similar to the game’s plateforme.
    - Allow the communication between the server side and client one.
    - Display the results of the back-end’s computation.


NOW YOU SHOW ME SOME DATA !!!

Data Description



- Available Data
  - Large set of data available
  - Manipulation of only 10% 60 Gb
  - Numerous attributes
- Parsing and Indexing data using Java (Lucene)
- Our Data

MatchID	BannedHeroID	PickedHeroID	TeamID(1,2)	Winner (0,1)

http://liris.cnrs.fr/dm2l/lol/documentation_lol_json.html

Approaches

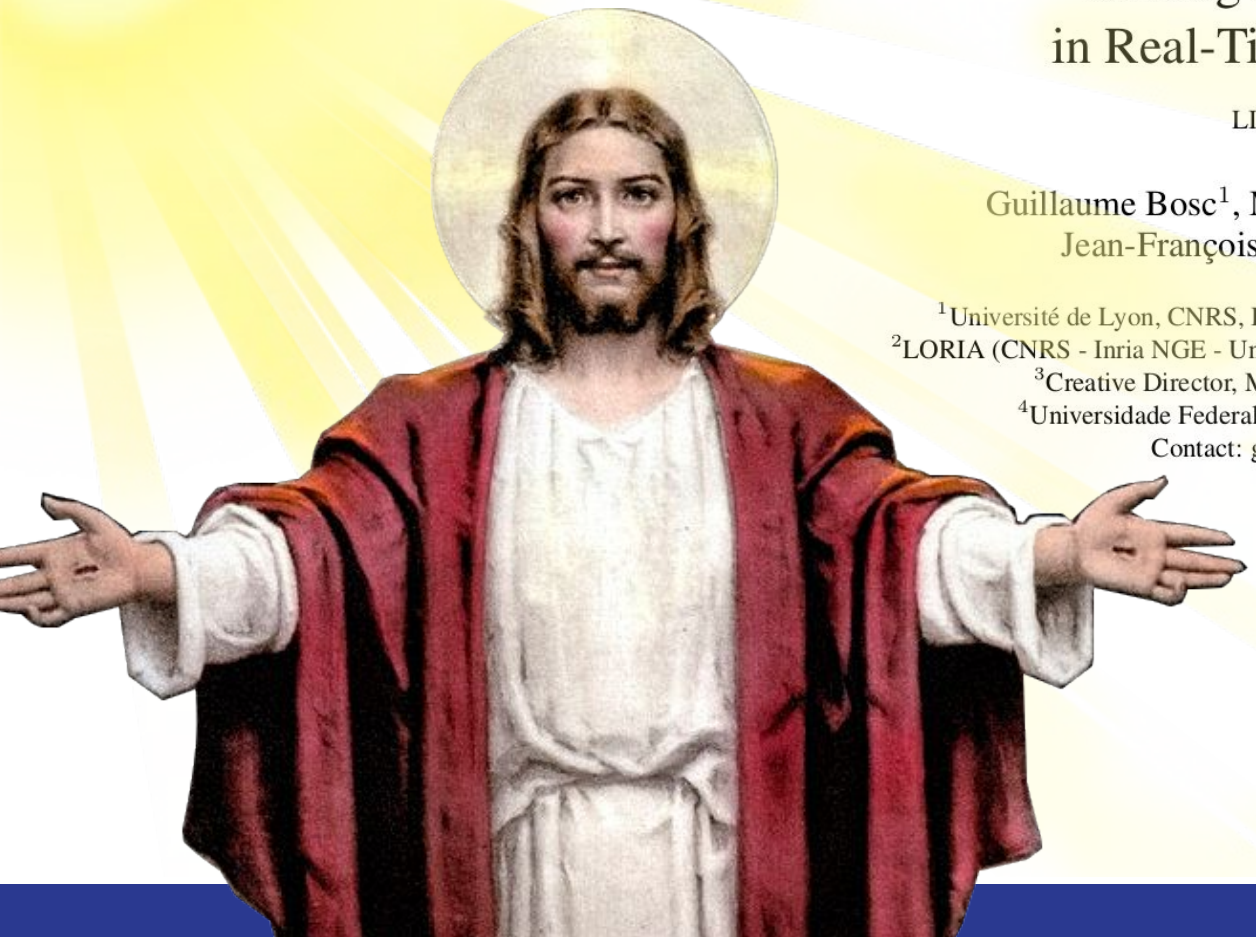
Pattern and Sequential pattern mining

- **numerous sequences**
- **Keep only relevant attributes**
- **Using SPMF**

Deep learning using recursive neural networks

- **Very interesting approach for this field**
- **Andrej Karpathy : Recurrent Neural Networks**

Sequential Pattern Mining



Mining Balanced Patterns in Real-Time Strategy Games*

LIRIS Research Report

Guillaume Bosc[1], Mehdi Kaytoue[1,4], Chedy Raïssi[2]
Jean-François Boulicaut[1] and Philip Tan[3]

[1]Université de Lyon, CNRS, INSA-Lyon, LIRIS, UMR5205, F-69621, France
[2]LORIA (CNRS - Inria NGE - Université de Lorraine), Vandoeuvre-lès-Nancy, France
[3]Creative Director, MIT Game Lab, Cambridge (MA), USA
[4]Universidade Federal de Minas Gerais, Belo Horizonte, Brasil
Contact: guillaume.bosc@insa-lyon.fr

May 2014

Sequential pattern mining

BalanceSpan*

Algorithm that extracts frequent patterns from *StarCraft II replays*.
Adaptation of the datasets to apply it.

- each item is represented as a unique number between 0 and 507 (127*4-1)
- each item labelled as + is represented by an even number (a) and its dual (labelled as -) is represented by the odd number (a+1)
- Result :
  - Frequent subsequences
  - The balance and the support of each sequence

*https://github.com/guillaume-bosc/BalanceSpan

Sequential pattern mining (Data Transformation)

IDs : entiers [1,432] —Dico1→ Name+ suffix ({+,-}{Ban,Pick}) : NewName

NewName —Dico2→ NewName+suffixe(entiers [|0-507|])

Example

Ahri 103 → AhriBan+ 4 AhriBan- 5

AhriPick+ 6 AhriPick- 7

Sequential pattern mining (Process)

User input : <ABCDEF>

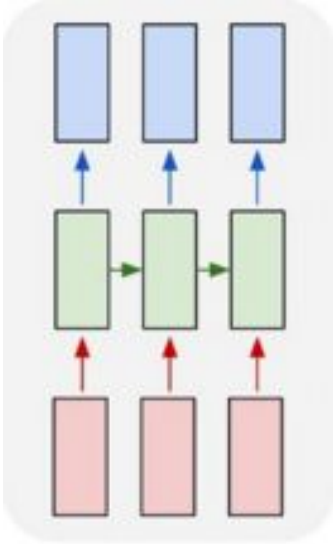
- Get all of the possible combination with a size equals of minimum between length (user input) and stated level
- Start by looking for the longest combination into freqSequences.txt (ex: <ABCD>)
  - If we have a hit : We extract the current combination from the user input and we start again
    ex : user input : <EF>
  - If not : We start again but with a lower level
- If level is null or sequence is empty, just stop

Eventually, our initial sequence is sliced in sub-sequences, where their length is maximized.


WE NEED TO GO
DEEPER

Recursive Neural Networks

Why RNNs ?



RNNs are Very powerful :

- Distributed hidden state that allows them to store a lot of information about the past efficiently.
- Nonlinear dynamics that allows them to update their hidden state in complicated ways

Keeps the order which is important for the game


FOR DATA MINING IS HARD AND FULL OF TERRORS

Actual Reached Goal

- Back-end
  - Compute statistics on the success of a team as they choose heroes to pick or ban.

- Front-end
  - Develop a web plateforme similar to the game’s plateforme.
  - Allow the communication between the server side and client one.
  - Display the results of the back-end’s computation.

Webographie

http://karpathy.github.io/2015/05/21/rnn-effectiveness/
https://github.com/guillaume-bosc/BalanceSpan
https://class.coursera.org/neuralnets-2012-001/lecture/77


Thank you for your attention !!
Any questions?