**League of Legends – Strategy Analysis**

**Team members:**

* Michael Merzlyak (email: michael.merzlyak@mail.huji.ac.il, csid: mikemerzl)
* Guy Ben - Eliezer (email: guy.beneliezer@mail.huji.ac.il, csid: guybene96)
* Yarin Schiller (email: [yarin.schiller@mail.huji.ac.il](mailto:yarin.schiller@mail.huji.ac.il), csid: yarinsch)

**Problem description:**

The Problem:

In Chess, when you learn the game it is very common to learn from the greats. It is very common to take some state of the game and study a pro move and take it as your own (Like: "Albin countergambit", "Amar opening", etc.). We decided take that intuition and see if it transferable. We decided to take the game "League of Legends" and using algorithms from the course to see if a low-level player, using strategies from an experienced one, has a better chance of winning the game (SPOILER ALERT: We succeeded).

We wanted to take this from a recommendation point of view, we tried to build a recommendation system for a low-level beginner, taking into account his own choices in the game.

The Game:

League of Legends is a team-based game with over 140 champions to make epic plays with. In the game, two teams of five players battle in [player versus player](https://en.wikipedia.org/wiki/Player_versus_player) combat, each team occupying and defending their half of the map. Each of the ten players controls a character, known as a "champion". During a match, champions become more powerful by collecting [experience points](https://en.wikipedia.org/wiki/Experience_point), earning gold, and purchasing [items](https://en.wikipedia.org/wiki/Item_(game_terminology)) to defeat the opposing team. In the game's main mode, Summoner's Rift, a team wins by pushing through to the enemy base and destroying their "Nexus", a large structure located within.

Summoner's Rift is the flagship game mode of *League of Legends* and the most prominent in [professional-level play](https://en.wikipedia.org/wiki/League_of_Legends_in_esports). The mode has a ranked competitive ladder; a matchmaking system determines a player's skill level and generates a starting rank from which they can climb. There are nine tiers; the least skilled are Iron, Bronze, and Silver, and the highest are Master, Grandmaster, and Challenger. Summoner's Rift matches can last from as little as 15 minutes to over an hour.

Data:

We gathered data on games from 2 different leagues, the bronze league, and the grandmaster league. Our goal will be to look at games from bronze league and obtain strategies from the grandmaster league.

By using several methods from the RIOT games developer API (LOL's company) we managed to get players names, from both bronze and grandmaster leagues. We didn't enter any parameters and tried to gather data from different part of the globe where the game acts to ensure independence between the players. Then, we get a list of id's from across the world an then we took a game from each player. After that we obtain game id's and take a "timeline" for each game.

From the data gathering we obtain bronze users and grandmaster users, with a size of around 100 kb, this data was a list of tuples with each tuple containing a player name and its region. These are "summoners\_grandmaster.pkl" and "summoners\_bronze.pkl" data.

Then we build a list of games from those players, the list contained game id's weighing at around 45 kb. These are "matches\_bronze.pkl" and "matches\_grandmaster.pkl".

We finally gathered the timelines of each game and ended with 2,000 bronze games timelines and 15,000 grandmaster timelines. This data is a Json format containing a lot of information, the important part is that every timeline contained a list such for every minute we have a list of events. Every event is a Json with keys according to what is the event. Events like "Elite monster kill", "Item purchased", "champion kill", "ward placed", etc.

Here is an example to see a structure for a single game. Each frame is a minute, containing several events:

תמונה שמכילה טקסט, לוחית, צילום מסך

התיאור נוצר באופן אוטומטי

We have a lot more grandmaster games than bronze for this question and we have 450 mb for the bronze and 1.2 GB for the grandmaster.

This is "timelines\_bronze.pkl" and "timelines\_granmaster.pkl".

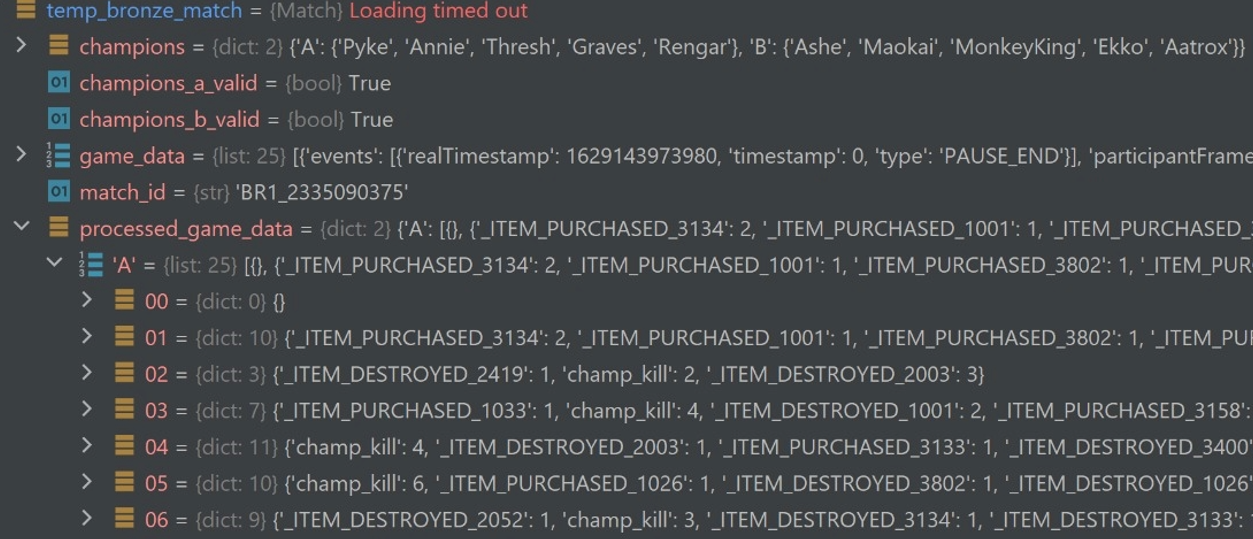
Solution:

To solve our problem we used two tools from course – Jaccard and cosine similarities. By finding for each bronze gameplay a similar grandmaster gameplay, we can recommend the bronze team a strategy with some win probability.

**Pre-processing the data -**

To check if there is a correlation between the grandmaster strategy to the bronze winning state we had to pre-process out data. In order to do so, we created two DataTypes to manipulate the data according to our needs.

The First DataType is **Match** (gameplay events of a complete match):  
Each match contains a list for each team (processed\_game\_data in the example below), each item of these lists is a minute and each minute is a dictionary with event types that occurred during that minute as keys (e.g champ\_kill, ITEM\_PURCHASED, etc…) and values that indicates the number of appearances of each event type during that minute. An example of Match object:



The second DataType is **Team** (gameplay events of a team):|  
Created in order to split each match to minutes events for each team ('A' or 'B') separately (example is not necessary, just look at 'processed\_data[<'A' or 'B'>]' from the example above).

**Jaccard similarity –** תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

In our case, Jaccard similarity is measured over one minutes of two different games. That means that the Jaccard similarity of some minute of two games is the number of common events that occurred in both games in that minute divided by the total number of events that occurred during that minute in both games.

**Cosine Similarity –**

תמונה שמכילה טקסט, שעון

התיאור נוצר באופן אוטומטי

Also measure over the same minute of two different games. We calculate the cosine similarity with the values of events appearances during that minute.

Eventually, in both similarities we calculate the average similarity of all minutes of two games and we get the similarity value of the games.

**Our hypothesis –**

If we take the first minutes of a bronze game, find a matching grandmaster game with high similarity value over these first minutes (which means that we calculate the similarity values over the first minutes only and not over the whole game), and then use the matching grandmaster strategy in the bronze game for the rest of the game, what is our winning probability?

**Testing out hypothesis –**

In order to test our hypothesis, we took