PRI 2021 Information Processing and Retrieval - League of Legends Champions and Items

Miguel Carreira Neves
João Gonçalves
Pedro Coelho
up201608657@edu.fe.up.pt
up201806162@edu.fe.up.pt
up201806802@edu.fe.up.pt
FEUP - Faculdade de Engenharia da Universidade do Porto
Porto, Portugal

ABSTRACT

In this paper we present a project that consists of making an information retrieval system which acts as a search engine for a given data set. We choose as our data the game called League of Legends which includes its Champions (characters) and Items.

The project is divided into three main phases: Data Preparation, Information Retrieval, Search Engine. Currently only the Data Preparation phase has been completed, as such we have all the data treated and cleaned and stored in proper format.

ACM Reference Format:
Miguel Carreira Neves, João Gonçalves, and Pedro Coelho. 2021. PRI 2021 Information Processing and Retrieval - League of Legends Champions and Items. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

To start off the project our team needed to find a theme and where to retrieve data for the project. After analysing some different alternatives the team settled for the data of the primary aspects of the game League of Legends, champions (different playable characters in the game, each being unique) and items.

We retrieved data from two data sources:

-https://ddragon.leagueoflegends.com (dragontail dataset);

-https://lol.fandom.com/wiki

The first source is a official website and thus it contains credible data from the company called Riot that develops League of Legends (also known as League or LoL for short).

The second source is that of an unofficial website made by fans of the game, although we consider that all the information there can be seen as credible since it is constantly reviewed by the whole community of that website and is usually always up to date.

From the first data source we downloaded a compressed folder containing the dragontail dataset. It was composed of multiple data files of which we selected only data related to champions or items. We choose two files, both in json, called champion.json and item.json.

We then converted the json files to csv using python and the pandas library since we considered this was a preferable format for executing data exploration and preparation.

At first only the dragontail data was going to be utilized however we soon noticed that the champion lore (biography) present there was incomplete and was just a summary. Thus the second data source was scraped for the information relative to the lore of the champions.

Our goal when retrieving the data was to gather it all into two csv files called champions.csv and items.csv each well organized and with one row for each champion/item respectively.

As a side note it is important to remember that due to format some figures might be too small the be read, however it is always possible to zoom in since they have great resolution.

2 DATA PROCESSING PIPELINE

This section is a summary of our pipeline and all the steps taken, each being further explained in the next sections.

The following image represents the pipeline used in our project.

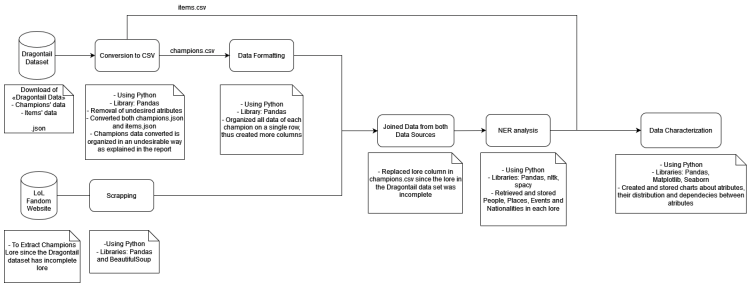

Figure 1: Pipeline

As seen in the pipeline image, after extracting the data from both data sources and converting it to csv files we cleaned it and organized it to make it easier to work with.

Afterwords we performed entity recognition on the champion lore to extract relevant information and store it in the champion.csv file.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

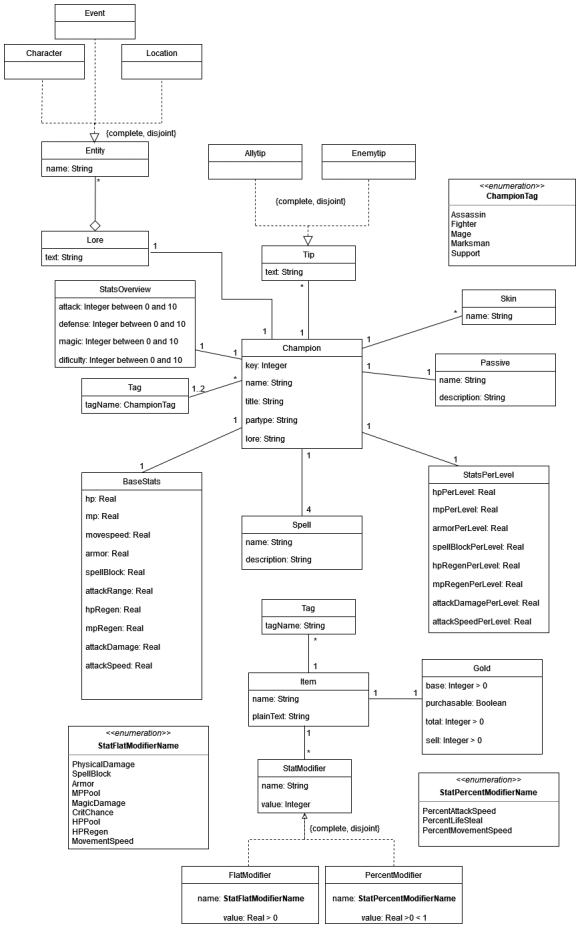

Figure 2: UML

And finally data characterization was made using python in order to create charts and analyse the data we extracted.

3 RELATIONAL DATA MODEL

Our Relational model consists of two main blocks, the champions and items as seen in ***figure2***.

4 DATA GATHERING

As discussed before we gathered data from two data sets, the dragontail one was stored in json format and the other one was in the web and thus needed scraping.

4.1 Conversion from JSON to CSV

This conversion was made using a python file json_to_csv.py and namely a library called pandas. However we verified that the data of the champions file in particular was disorganized, being vertically organized for each champion in some columns (ex: skins) as seen in ***figure3***.

data__|__key	data__|__name	data__|__title	data__|__skins__name
266	Aatrox	the Darkin Blade	default
			Justicar Aatrox
			Mecha Aatrox
			Sea Hunter Aatrox
			Blood Moon Aatrox
			Blood Moon Aatrox Prestige Edition
			Victorious Aatrox
			Odyssey Aatrox

Figure 3: Champion Data set after conversion to csv

key	name	title	skins__name01	skins__name02
266	Aatrox	the Darkin Blade	Justicar Aatrox	Mecha Aatrox
103	Ahri	the Nine-Tailed Fox	Dynasty Ahri	Midnight Ahri
84	Akali	the Rogue Assassin	Stinger Akali	Infernal Akali

Figure 4: Champion Data set after formatting data

4.2 Scraping

We scraped https://lol.fandom.com/wiki for each champion using a python file called scrap.py and a library called BeautifulSoup. After we scraped the lore for each champion we then inserted it into the already existing lore columns on champions.csv thus substituting the old values which contained only a summary of the lore.

5 DATA PROCESSING

5.1 Data Formatting

In order to tackle the problem of the disorganized champions data, we used a Python file called organize_some_cols_to_rows.py to fix these columns, extending the values for each champion along their row. As such new columns were created (ex: skins01, skins02, etc.) and so all the information of each champion became organized in each line as seen in ***figure4***

5.2 Performing NER (Name Entity Recognition)

Using the new lore obtained from scraping, an algorithm for extracting entities is used with the objective of better representing the lore and its content.

In order to perform this operation a python file called lore_ner.py is utilized, which uses nltk, pandas and spacy libraries to perform NER, it then stores the entities recognized in new columns organized in persons, locations (gpe), nationalities (norp) and events involved in that lore. In order for these new atributes to be stored in the same fashion as the others (in one row per champion) we created and filled multiple columns going from 0-N (where N is the number of persons for example in that lore).

6 DATA CHARACTERIZATION

In order to characterize the data we used the pandas and matplotlib python libraries to make plots that summarize the general characteristics and features of the data set.

6.1 Champions

6.1.1 Tags. The champions are game characters which have a multitude of attributes, there are several types of champions and various ways to divide them. We have 6 tags given by the game for characterization

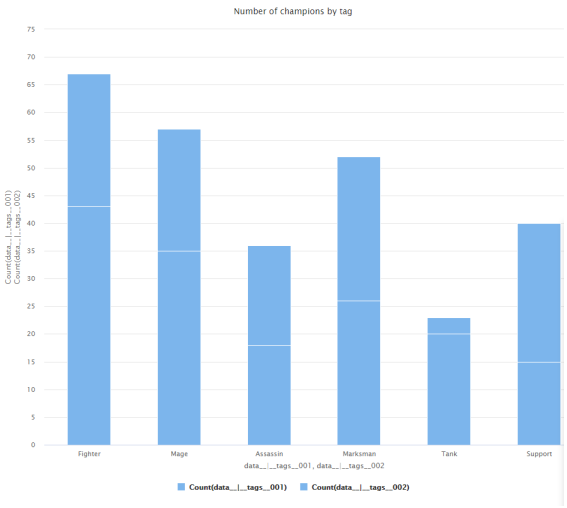

Figure 5: Number of champions depending on both their tags

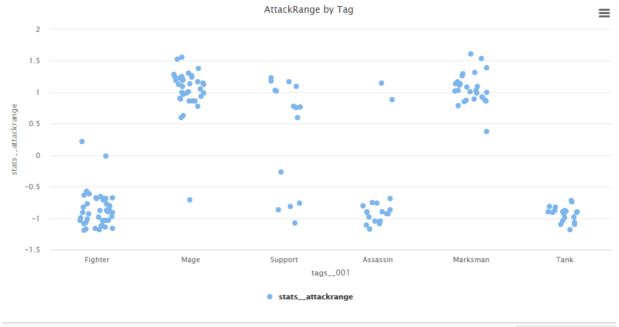

Figure 6: Variation of attack range depending on first tag

Each champion has 2 tags, a primary and a secondary and in the following graphic we can see the distribution of those tags in ***figure*5**.

The most popular tag is Fighter and the least popular tag is Tank. Supports are the type of champions that a allow for more variation. A majority of the champions considered as a possible support only have it as their second tag which is the opposite of the tanks.

6.1.2 Champion Stats. Champions are designed to be of a certain play-style, that being represented most times by their tags, but their base stats can also reveal that.

In ***figure*6** we compare the relation between the tag and the base stat called attack range of the champions.

As we can see, fighter, assassin and tanks have a shorter attack range as they are melee. Mages and marksman have a higher range, although there are a few exceptions.

This only helps the notion that the champions stats distribution are quite different among tags. In ***figure*7** we can see this same phenomenon happening relative to other base stats as well.

By looking at the graph we can see that Mages have a high magic stat. Also Marksman champions are clearly inclined towards attack and are weak in magic and defense. The clearest difference can be

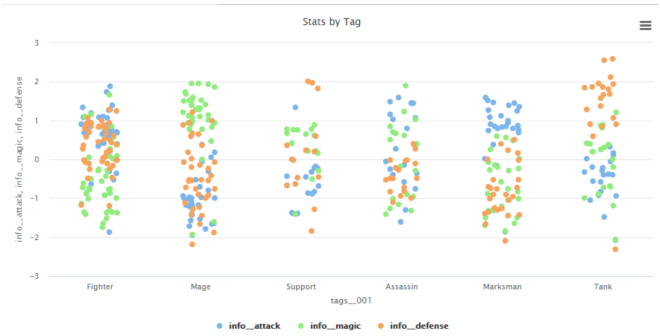

Figure 7: Variation of Champion base stats depending on first tag

Figure 8: Variation of Champion base attack depending on the base magic, colored by defense

seen within the fighters, these have mostly defensive attributes and but they versatile in their damage type, being able to have either more attack or more magic stats. If we look into the assassins base statistics we can clearly see that they have generally low defense has they were created to be champions with high damage and mobility but low resistances.

Each champion has a base attack and a base magic value. Normally the champions tend to be inclined towards one or the other as verified in ***figure*8**.

From the coloring of the graph we can also see that champions that have low magic and attack have more defense, these are the tanks. The champions that have a higher value of both or either offensive stats generally have a substantially lower defense value.

There is a clear pattern showing in this scatter plot, if a champion has 2 high values of base stats it will generally have 1 low value stat or the inverse or a champion may have 3 mediocre stats. This is intended by the game designers in order to try to keep champions balanced and leveled with each other, having them sacrifice something in order to gain another thing.

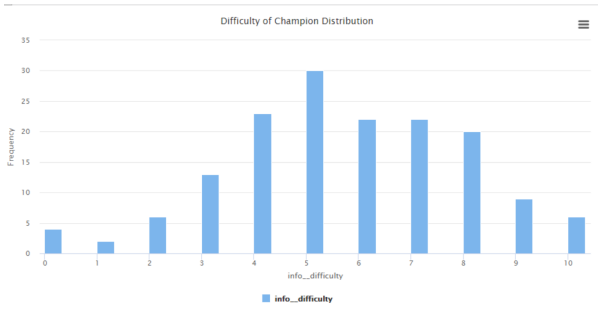

Figure 9: Distribution of Champion difficulty of use

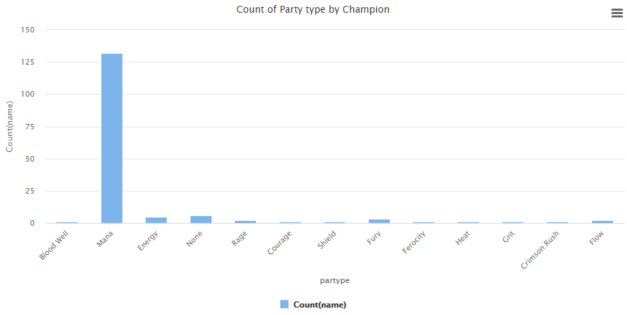

Figure 10: Number of Champions by resource used

6.1.3 Champion Difficulty. Each champion has a difficulty associated with it. It is measured by the amount of practice (theoretically) you need with the champion to be good with it and use its full potential. Champions that have a higher difficulty are intended to be played by better players and average difficulty ones are for the average player.

Assuming the player's skill level follows a normal distribution (which is a rather safe assumption), it would only make sense that the difficulty level of champions also follows or tries to replicate the same distribution. This is intended so that there are champions available for every type of player as shown in ***figure*9**

6.1.4 Champions Resource. For the champions to use abilities they need to expend some sort of resource, most of the champions use Mana but some have their own specific way of casting spells as seen in ***figure*10**.

When doing data analysis over these values we can expect some hardships around the attributes with the lowest users. This can also have an upside because most of the champions use the same thing making it simpler both for the players and the developers of the game.

6.2 Items

6.2.1 Item Costs. The chart on ***figure*11** regarding items relates items by their cost. As we can see the most popular costs for items are 400, 2900, 3200 and 1100. On the opposite end we have quite a few gold costs with only 1 item associated. This makes sense once 400 normally represent starting item costs, 1100 represents medium items/components costs and 2900/3200 represents full item costs.

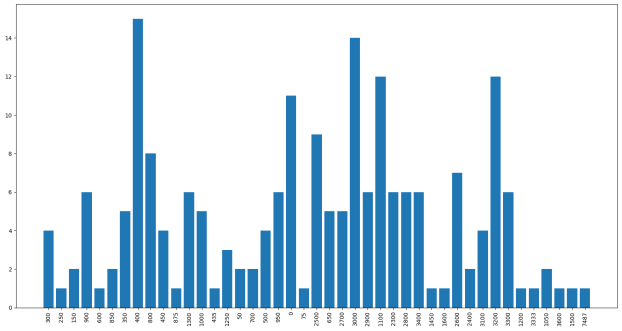

Figure 11: Number of Items by gold cost

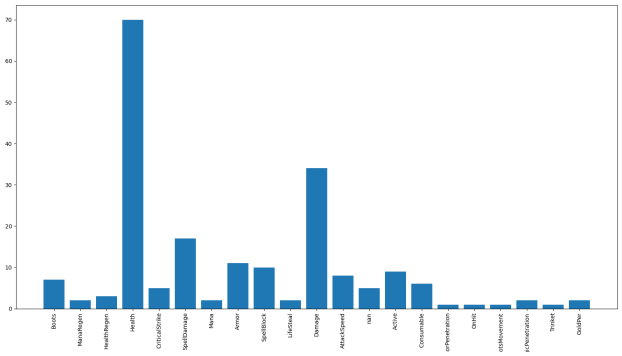

Figure 12: Number of Items by tag

It is important to mention that items have a upgrading tree and as such normally starting items are needed to create medium ones and the same for full items, which may help us understand this distribution of values.

6.2.2 Item Tags. Items have tags associate with which one of them.

The ***figure*12** represents the number of items per tag. Items have various tags and some are more prevalent then others as we can see through the chart. The most common tag is Health followed by Damage. Many items end up giving stat modifiers to Health even if its not their main focus since Health is one of if not the most important statistic in-game.

7 CONCLUSION

In conclusion, after finishing data retrieval, cleaning, preparation and characterization we realized that we have data with many interesting aspects to explore and make relations necessary to implement the search engine towards the end of the development of the project. One of the aspects we are most excited about is seeing the search engine working properly on the Champions lore since many of their biography's are connected to each other or the same places or events.