The League of Mythics

Reed Semmel, Allen Hughes, Greg Brown, and Paul Velas

***Abstract*— In this document we will detail how our team, the League of Mythics, used data from the massively popular Multiplayer Online Battle Arena (MOBA) League of Legends (LOL) to analyze a major part of its design, mythic items. Riot Games is keen on the idea of mythic diversity, the idea that champions should have a diverse set of mythic items they build across the board. We analyze publicly accessible data to conclude that mythic diversity is rather poor, and has decreased as time progressed.**

I. BACKGROUND

League of Legends is a video game where two teams of five players play in matches on a battlefield with the ultimate goal of destroying the enemy's base. This is easier said than done, multiple structures, objectives, and enemies stand in the way. As the game progresses, each player controlling playable characters called "champions" earns gold through passive and active means (claiming objectives, killing enemies, *etc*). Gold is used to purchase items, which improve the power of your champion. Specifically, we will be looking into mythic items in detail. These are a special group of items that generally provide the best value and therefore are usually purchased first. They have a constraint that you may only have one mythic item. This creates a choice, and where humans have choices, there is data to be had. In our case, League of Legends has 157 champions and 23 mythic items as of November 2021, giving us thousands of possibilities to analyze.

II. OBJECTIVE

Our objective, as described above, is to analyze the entirety of mythic items in League of Legends and how they affect different aspects of the game. This comes in two parts: first was the challenge of collecting data from Riot Games' API (Riot Games is the developer of League of Legends), and second was analyzing this data to see what conclusions we could draw. One of the key topics Riot Games talked about this year was the idea of mythic diversity Riot's exact goal with the item changes in the 2021 season is to ensure that "no champion chooses the same mythic [item] in 75%+ of games" [1]. As of that blog post in February, 88% of champions reached that goal. They did not want players building the same item every single time for a given champion. They created the mythic items in hopes that they would give options on which to pick each game in order to give an advantage over the opponent. We wish to explore how successful this venture was and see what other conclusions we could draw from the raw data that might not be available on aggregate data analysis sites for the game such as `u.gg`, `op.gg`, `lolalytics.com`, *etc.*

III. OVERVIEW OF DATA

A. API Layout

The source of our data is the Riot Games' API. This API provides a variety of endpoints for match and player statistics. Since we want to analyze mythic items, we want to get details of many individual matches. The API has an exact endpoint for this: the match API. This endpoint returns a list of all ten participants in a given match with which champion they played as, what items they built, how well they performed, *etc*. However, we still match ids to get this data. In order to get these match ids, we have to use a few additional endpoints. The first of which is the league entries endpoint. This is the source of our data, as all it requires to call is a rank, division, and page, which can be simply iterated through. This endpoint returns roughly two hundred users which currently are ranked at that rank and division. From here, we need to convert their summoner id to a PUUID since the Riot API is half way between versions while they are restructuring their infrastructure. Currently, the league entry endpoint is on v4, which uses summoner ids to identify users, while the match endpoint is on v5 which only supports PUUIDs. Getting a list of all ids associated with an account is as simple as calling an endpoint in the summoner section of the API with the id you have. Once we have the PUUID, we can get the match history for a given player, which is simply a list of match ids. We can finally use these match ids to get the data we are after.

Additionally, the API has a rate limit. We are limited to one hundred requests every two minutes. This is concerned as we want to collect hundreds of thousands of matches. With the time-frame we have, this would be greatly limiting. We need to put great care into collecting data efficiently.

B. Distributed Collection

Since we are a team of four, each of us can get our own API key with separate rate limits. While we can quadruple our throughput, this increases the data collection complexity as we now have to keep track of a distributed system. We decided on uploading collected data to a MongoDB collection while all concurrently scraping from the API.

C. Limited API Call Overhead

With low rate limits, we must be careful with our API calls to not waste them. Only one API call gives us the actual data we want: the match id to match data. All other ones are just overhead to get the match ids. We want to limit the amount of overhead calls to improve collection rates. The path described above suits us best. The major benefit of the league entry endpoint is a single API call gives us two hundred users who have played ranked games. For each of these users, we need to make two API calls: one to get the PUUID, and one to get the match histories. This gives us an overhead of roughly 2.005 API calls per user.

Using this method is not enough, because now we need to optimize the matches we collect. Since there are ten people on a team, we will get match ids shared across multiple users, which are useless to us because we already have that data. To limit the possibilities of duplicate match ids, we collect users from only ten percent of entries.

The match history API lets us query with great detail, which allows us to make decide how exactly we should go about getting the most match ids for a user. Since we are using a ranked leader board to source our users, it is very likely they have played plenty of ranked matches. The benefit of this is two-fold: our user sourcing method gives us users who are likely to have many matches in a particular game mode (ranked), and ranked is competitive so people will be trying their best to win. Using the league entry endpoint, we can get certain ranks. We chose to use everything from Gold IV and above. This represents about forty percent of the ranked player base according to rough estimates from number of pages for each rank and division. Using the top forty percent of players gives us a high quality data set since the players generally have good knowledge of the game. This method will also decrease the amount of trolls because ruining matches can get you banned from the game.

D. Collection Process

With this knowledge, the task of data collection no longer looks as daunting. The largest road block now would be the unreliability of the API. The API will frequently return HTTP errors such as 429 or 503 at no fault of our own, so our collection process has to account for these and work around this. After much trial and error of encountering any and all errors that would pop up. We created a system that would almost never crash unless the entire API is suffering an outage for multiple minutes. The collector script is extremely simple to use as well. Each of us have a JSON snippet represent a range of league entries to work our way through, and if the script ever exits, simply restarting it will make it pick up right where it left off. We were very happy with how well the script turned out, and most of the human demand came from replacing the API key which expires every 24 hours.

E. Collection Results

Our data, in its rawest form, was all processed into a SQLite3 database after using MongoDB to store the API response. Since we are only interested in what each participant did, we can flatten the array of participants in each match and store rows of participants in a single table. However, this data would have been very hard for a lot of the analysis, as many areas such as identification of mythic items were abstracted, with the mythic data being reduced to numeric IDs. We processed it into much more usable data, fitting IDs and everything into a second, much more usable aggregate table. We used SQLite3 to analyze it all, and used the "SQLite DB Browser" application to great effect, as it made it massively more easy to construct queries that spanned multiple lines, and to save both them and their data for future reference. These queries gave us tables of data that we used both in that form and by processing them further through Python to create graphs and plots and whatnot.

Ultimately, we were able to collect over the course of roughly a week almost four hundred thousand unique matches, totaling to nearly four million instances of players.

IV. ALGORITHMS AND MODELS

No special algorithms or models were used in the analysis. For our purposes, we did not need to “predict” anything since all of the data is already here. We collected a sample of the data and we use traditional analysis techniques to discover findings.

We purposely do not want to clean or filter any data returned from match endpoint. After verifying that the data is most likely correct, we want to analyze all matches collected because even if people had a bad day or ruined the match for everyone else, player choices were still made, which is the primary topic we are analyzing.

V. RESULTS

A. The Importance of a Mythic Item

Fig. 1. Mythic vs No Mythic

Champion	Mythic	Wins	Games	Winrate
All	None	96707	289459	0.334096
All	All	2032263	3968491	0.512100

Fig. 2. No Mythic Winrate Filtered

Champion	Mythic	Wins	Games	Winrate
All	None	23847	71648	0.33284
All	None	39251	107426	0.36538

Firstly, We wished to examine how the purchasing of a mythic might correlate to whether a match is won. Of all games, as expected, it is reasonable to say that there was a winrate of exactly one half. It makes sense: 10 players enter, 5 leave victorious, for the most part. Leavers, griefers (taking action to put your own team at a disadvantage), and AFK players (Away From Keyboard, AKA players that are not presently playing the game) can be reasonably assumed to not affect overall winrate, as it is equally likely to occur on either team.

However, these three categories of players that are much more likely to cause a loss will almost always not buy a mythic, either through choice or refusal to play the game. A player that has earned enough gold from minion kills, player kills, and other objectives and goals can be assumed to be playing the game as regular.

Figure 1 shows all games in total, with no filtering for any of the three categories. Figure 2, above, shows a simple filter of goldSpent > 500 on the bottom row. 500 gold is the starting amount of gold, and it can be reasonable assumed that a player actively participating will buy that and use any extra gold they earn on items. Players who are actively participating but who do not refuse to but just forget to or are too inexperienced to buy more than that should be statistically insignificant enough, and could be counted as a griefer potentially as there is a comprehensive tutorial, in-game suggestions on what items to buy next dependent on lane and champion, and having no items or only the starting items puts you at a significant disadvantage that only grows as the game continues. We could not find any other stats that I could in good faith use to try to gauge whether a player is not playing or griefing actively other than the bare minimum of totalDamageDealt > 300, as I do not want to mistakenly label players just having a bad game, especially on low-damage support champions. The goldSpent filter eliminated 180k of the 290k players who did not purchase a mythic, and the totalDamageDealt filter eliminated a further 30k. However, there was actually not a significant change in winrate for either, with the last filter actually decreasing it below the original. (Refer to Figure 2 above, the first row is both filters applied and the the second in only the goldSpent filter applied)

From this, we can conclude that for games where you are able to purchase a Mythic, there is a very small increase in the odds that the match will be won (1.2% is decently significant over a sample size of 4 million).

On the flip side, games where a player is not able to purchase a Mythic, even taking into account potential griefers and leavers, still has almost only a third (36.5%) of a chance to result in a win.

B. Diversity Across the Board

Fig. 3. A graph of the pick rate of the most commonly picked mythic on each champion.

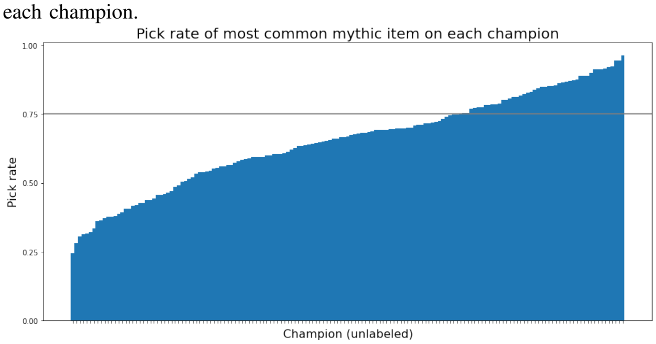


Figure 3 shows the pick rate of the most common mythic on each of the 157 champions. This graph is a far cry from Riot's vision. 46 of the 157 champions do not meet Riot's goal of having a mythic item being picked no more than 75% of the time. This is much fewer than the previous 88% as reported by Riot in February [1].

Let us take a look at how many choices champions really have. "Commonly" purchased items is rather subjective,

Fig. 4. Number of champions that build a number of mythic items at least five percent of the time.

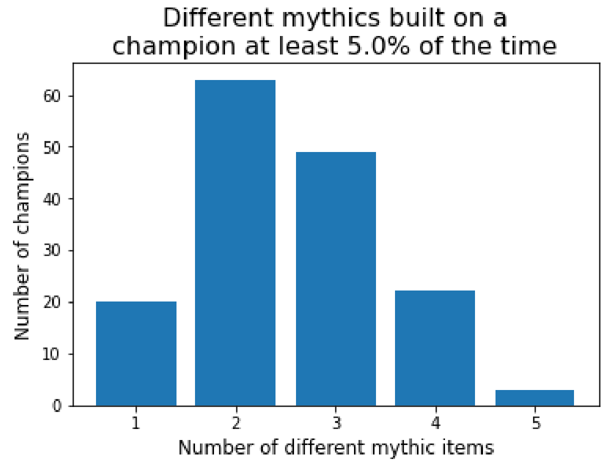
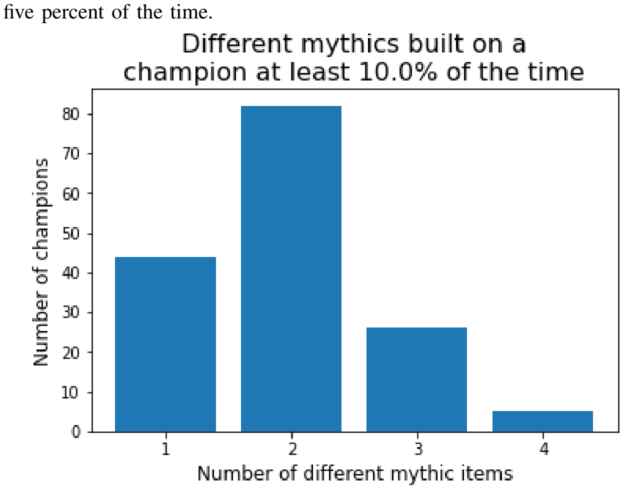


Fig. 5. Number of champions that build a number of mythic items at least five percent of the time.



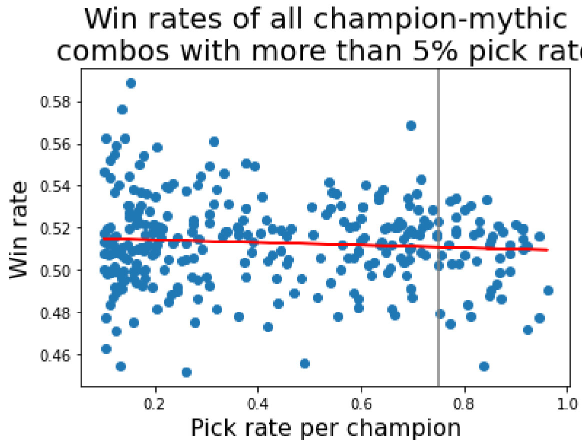
so we present two charts. One that shows the number of champions that purchase a certain number of mythic items at least five percent of the time, and one for ten percent. Chart 4 shows that nearly twenty champions only build a single mythic item more than five percent of the time, showing that these champions have almost no other choices than that one item. Most champions have either two or three available options if we define common to be five percent. When we move up to defining common at ten percent, we see a significant difference. The vast majority of champions have only one or two mythic items that would be commonly found on them.

Finally, we will take a look at the scatter plot of all champion-mythic composites (Figure 6). This graph shows very little difference as a whole between common and uncommon item picks.

C. Diversity in Enchanter Items

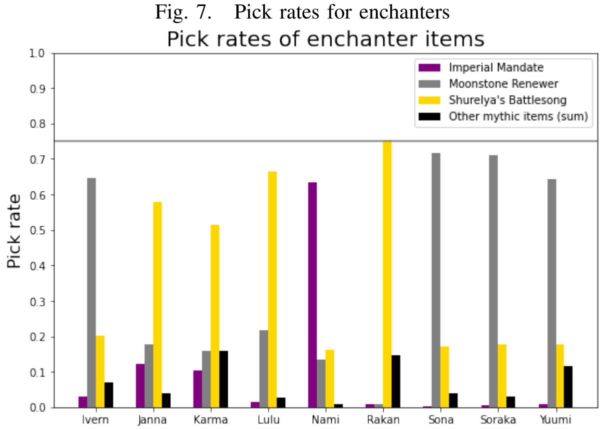
To dive into more detail, we will look at a specific group of nine champions who mostly only have three viable mythic items tailored to them. The exact details of the champions

Fig. 6. Scatter plot and regression of all common champion-mythic composites pick and win rates



and items aren't relevant - all that is important is they are related and perform similar tasks on a given team. The enchanter role provides value to the team by healing and empowering allies instead of doing damage directly to the enemy. We will take a look at the graphs of the pick rates and win rates of these three items on these nine champions.

Fig. 7. Pick rates for enchanters

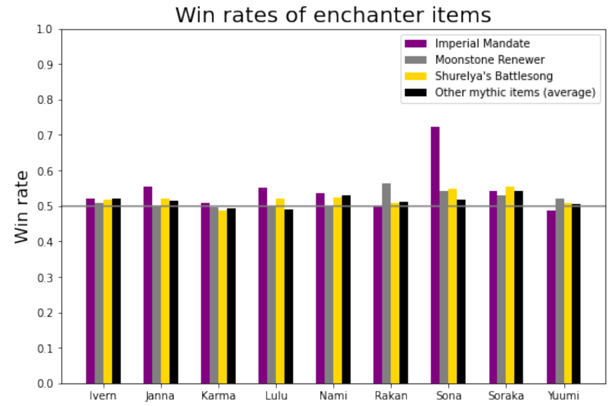


Based on Figure 7, it is pretty clear that each champion has a significant preference, and the preferred mythic item is rather varied, with the exception of Imperial Mandate - only one champion prefers this item. The highest "second place" item ranks in at a pick rate in the low twenties percent. On the bright side, only one of the champions do not meet Riot's diversity goals, and it is by a very thin margin.

It gets more interesting when we consider Figure 8. There is a noticeable outlier of Imperial Mandate on Sona. However, the pick rate for this champion-mythic composite is near zero, so the win rate is not statistically significant. All win rates are very close to the fifty percent average, indicating that the choice of the mythic item is not as important as the disparity in the pick rates would make it seem. The non-enchanter items which are rarely picked seem to also have relevance.

We also wanted to analyze whether diversity was impacted

Fig. 8. Win rates for enchanters



by role chosen in game. League of Legends has 5 roles: Top Lane, Mid Lane, Jungle, ADC, and Support. We chose to analyze two of the most popular roles, Mid and Jungle, in order to find data to support our query. To go about this we chose to look at the five most popular champions for each role. In Mid Lane the champions were as follows from Figure 9: Yasuo, who had 48k matches played, Yone, who had 41k matches played, Zed, who had 35k matches played, Leblanc, who had 32k matches played, and finally Sylas who had 30k matches played. We then took these heroes, and looked at the amount of games where they had chosen their most popular mythic item in order to determine the percentage of matches where they chose that mythic in the mid role. The percentages and mythics were as follows from Fig3: Yasuo chose Immortal Shieldbow 91% of the time, Yone chose Immortal Shieldbow 92% of the time, Zed chose Duskblade of Drakthar 44 % of the time, Leblanc chose Ludens Tempest 89% of the time, and Sylas chose Everfrost 88% of the time. So, throughout the Mid Lane's 5 most popular heroes, the goal of the same item being built less than 75% of the time was met only on one hero, Zed.

Fig. 9. Top 5 Most Played Champions in the Mid Lane

champion	mythic	gamesMid
Yasuo	All	48431
Yone	All	41847
Zed	All	34667
Leblanc	All	31505
Sylas	All	30080

In the jungle role however, things look a bit more hopeful for Riots case. The 5 most popular heroes for the Jungle role were as follows from Figure 10: Kayn with 55k games, Lee Sin with 46k games, Viego with 36k games, Graves with 47k games, and Khazix with 45k games. The champions most popular mythics and pick rate for those mythics are as follows: Kayn chose Goredrinker 71% of the time, Leesin chose Goredrinker 81% of the time, Viego chose Divine Sunderer 60% of the time, Graves chose Immortal Shieldbow 55% of the time, and Khazix chose Duskblade of Draktharr 68% of the time. From these numbers we can see that the mythic diversity of the Jungle role is in a considerably better place than the Mid Lane role. In fact out of the top 5 Jungle champions, only one champion's mythic pick rate does not corroborate Riot's goal.

Fig. 10. Top 5 Most Played Champions in the Jungle

champion	mythic	gamesJungle
Kayn	All	54940
Graves	All	46723
LeeSin	All	46538
Khazix	All	44638
Viego	All	36761

D. Conclusions

Overall, there is a large disparity in purchase rates of mythic items across the majority of the champions. Riot has failed to make their goal of having no champion pick the same item in more than three fourths of games, with the champion failure rate more than double the rate presented in their February blog post [1]. This, however, may not be any fault of their own. The win rates of all common champion-mythic composites are very close to 50-50. These items are viable despite their low pick rate. This puts Riot in a tough spot because if the items are balanced, making an item stronger will most likely make that item far too strong.

VI. PRIMARY ISSUES ENCOUNTERED

One primary issue was with the time limited API keys. Riot Games' API uses API Keys that are only valid for twenty four hours, which made extended use of our scraper a bit harder. We designed the scraper with this limitation in mind, and restarting collection was as simple as editing the environment file to update it with the new API key, and restart the script. It will automatically restart were it left off.

Another issue was that the scraper has a chance to shut down at random points. We had to keep an eye on it and check back in every now and then to restart it if it had ceased to function. Thankfully, this was intermittent until near the end of our collection period when the API suffered much downtime.

VII. FUTURE WORK

Since the primary idea of our work was to test whether Riots goal of mythic diversity was present in League of Legends, the most obvious step forward from our project would be to continue to assess the diversity of mythic items throughout further patches to the game. One of the common ideas generated throughout the entire team's analysis was that even if champions do achieve Riots numerical goal for mythic diversity, *i.e.* the same item is built less than 75 percent of the time, most champions remained very close to this metric. Our research shows how close Riot is to invalidating their claim. Moving forward, our work could be used to suggest changes as well as balance other issues to game play. For a game developer as large as Riot, a project as small as ours would likely go by the wayside, but we believe that our analysis reflects the current state of the game. Going forward, our work could be used to analyze a longer period of time to more accurately reflect how the state of the game changes over a greater time frame. Since we only analyzed a small number of matches over a short period of time, a larger sample size over many different patches (iterations) of the game may generate new ideas that we were not able to see with our smaller sample size. Additionally, with a larger data set, instead of looking at individual champions, we can look at match ups of one champion vs another.

VIII. TIMELINE AND RESPONSIBILITIES

Each member in our group was responsible for several aspects of the project. Reed wrote the data scraper and helped in the analysis of the data. Paul and Greg both analyzed the data and wrote this paper. Allen helped analyze the data and organized the presentation. Reed finished the data scraper around the middle of October when it was initially expected. After this we ran into some problems with Riots rate limiting issues so the data took significantly longer to gather, and finished sometime around early November. We took most of November to export the data from MongoDB to a SQL database, and then finished up our analysis in late November - early December. Our hard deadline for final analysis was November 30th and the final presentation was finished on December 3rd.

REFERENCES

[1] "Quick gameplay thoughts: Feb 26 - league of legends," leagueoflegends.com. [Online]. Available: https://www.leagueoflegends.com/en-us/news/dev/quick-gameplay-thoughts-feb-26/. [Accessed: 28-Sep-2021].