Meta Search: A Tool for Mass Analysis of Game Strategy

Ben Janzen*
UBC Dept. CS

Giovanni Viviani†
UBC Dept. CS

Abstract

In recent years, competitive gaming, also known as e-sports has become more and more popular. Crowd funded tournaments such as the DOTA 2 International have raised millions of dollars in prize money. As is common in many competitive sports, fans and players enjoy discussing and debating strategy. This design study introduces Meta Search, a tool designed to support mass audience analysis of strategies in DOTA 2. Meta Search specifically focuses on providing an easy to use visualization that helps players understand trends and anomalies in the Hero selection meta game of DOTA 2. Heroes were classified according to the roles they can potentially fulfil, and trends in these taxonomic groups were visualized using simple aesthetically pleasing juxtaposed overview and detail views. This paper also presents strategies for scraping DOTA 2 match results from the official API, and suggests methods for providing responsive end user access to such a large amount of data.

Index Terms: K.6.1 [Management of Computing and Information Systems]: Project and People Management—Life Cycle; K.7.m [The Computing Profession]: Miscellaneous—Ethics

1 Introduction

DOTA (Defence of the Ancients) 2 is a fast paced, free to play action game based upon a custom map from Warcraft 3. Since getting a stand alone release DOTA 2 has become one of the most popular online games in the world, clocking in over 600 thousand unique players online in a single day (stat taken Tuesday 28th October 2014). DOTA 2 has two teams of five players compete. Each team has an "ancient" building that they have to defend from the other team. The ancient is defended by a series of stationary automated guard towers. Both teams also have armies of non-playable soldiers called "creeps" that slowly stream towards and attempt to attack the enemy base. Every game has each player chose one hero out of a pool of over a hundred to play as. All heroes have unique sets of abilities and can fulfill one or many different roles within the game. Some heroes try to kill enemy heroes, others try to support their allies and yet more are designed to ignore the other players and take down the enemy towers as fast as possible. As a single match progresses players earn money and experience to upgrade their heroes with various items and skills. While these low level decisions of what to buy and how to play a specific hero matter, We focus our analysis on the simpler problem of hero selection. On a broad level, because each hero has a unique and unchanging set of skills, most are best suited to one or two particular roles, and we argue one can gain reasonable insight about player strategy based on the heroes chosen.

Even if you remove consideration for how specific item and skill choices may effect your strategy, the set of trade offs that go into a winning team composition is immense. With matches usually lasting half an hour or more, it is simply impossible for any one player to play the hundreds of games required to test out all possible strategies for all possible heroes and see for themselves which strategies are powerful. This issue is complicated further by the fact that the game is also under constant development and strategy balance tuning. The recent 6.82 Rekindling Souls update is perhaps the most extreme example of this. The game map layout was changed significantly, two heroes were remade almost entirely, and almost every hero or item had at least some minor changes. Any one or combination of these changes could have a profound impact on how strong specific heroes are, could enable new team compositions or fundamentally change how people play the game. Thus many players turn to online tools and guides in order to learn about the hero selection meta game. In addition, DOTA 2 has also become a very popular competitive e-sport with several tournaments managing to crowd fund prize pools in the millions of dollars. Expert level players have the added incentive of competing for real money and thus need fast and reliable access to detailed match statistics and tools to optimize their strategic choices.

To support strategy development and discussion, the developers of DOTA 2 provide a public web API to access detailed statistics about every match ever played on their game servers. We found this API to be extremely slow, inefficient and lacking many key features. Thus we present a variety of coping techniques to improve our end user's experience of interacting with the match history results. A variety of tools and projects have already attempted to take advantage of this ready source of data, but we argue that they have failed to provide a sufficiently usable and useful tool.

This paper also presents Meta Search, a tool to enable both casual and detailed analysis of trends in the DOTA 2 hero selection meta game by a lay audience. Meta search was based upon the well known stream graph to provide an easy to use and extremely useful analysis framework for trends in hero selection. A juxtaposed small multiples detail view allows users to drill down into more interesting statistics of the various heroes. Importantly this helps users to understand not only what heroes are currently strong, but also why they are strong. This understanding is key to allow players to move beyond simply playing the accepted meta game, to the formation of their own novel strategies.

2 Related Work

The core of our problem is the analysis of player choices and strategic decision making in games. In a broader sense, these sorts of questions are at the core of fundamental topics in psychology and more specifically game theory. This discipline investigates rational individuals who think strategically about what they expect other individuals to do as a means to argue about a wide variety of human behaviours. [7] This paper, however, is not interested in the broader implications on human behaviour but instead analyzes game strategy as a goal in and of itself.

An investigation into a specific game's strategy from a humanities or psychology perspective would likely include questionaries, guided play sessions and interviews. [5] These approaches are excellent for getting a deep understanding of a small set of players to form broad psychological hypotheses. However, they do not capture a sample size anywhere near large enough to understand broad trends in the meta game. For the purposes of this paper, meta game will be defined as the high level strategic choices players make. This is consistent with emic usage and understanding of the term within our target mass audience, but does not encompass all as-

*e-mail: bjanzen@cs.ubc.ca
†e-mail: vivianig@cs.ubc.ca

pects of the academic definition. [3] From the authors experience as members of the DOTA 2 community, this includes what heroes players tend to pick, what items they buy, what order they purchase skills in and how they play there characters on a moment to moment basis.

Electronic games have opened up a variety of new methods for studying and understanding game strategy in both academic journals and community fan sites. In particular, readily available log data, match statistics, combined with a large, enthusiastic and tech savvy audience has resulted in a number of data mining, machine learning [10, 1] and visualization [4, 8] based tools originating from within and outside of academia. DOTA 2 players will be very familiar with websites like DOTA Buff that process and aggregate all matches played in the last month into simple spark lines and ordered lists. Generally speaking Every few months and after major patches or tournaments analysts within the community will compile these statistics to create a hero strength tier list. These are categorized ordered lists that bin heroes based on their popularity and strength in high skill level competitive games. Many new players who are overwhelmed with the amount of choice available to them will simply pull up the most recent tier list from a fan site, and pick one of the top rated heroes.

While these techniques have a number of strengths, we argue that they do not provide a sufficient framework for easily accessible meta game analysis by a lay audience. Machine learning techniques in particular are powerful at answering specific hypotheses, but tend not to generate new hypotheses and the results can be hard to disseminate to a mass audience. Well designed, aesthetically pleasing, and, easy to understand visualizations are an excellent way to reach a broad section of users and ignite discussion. [2, 9] The standard hero tier list is a simple, useful and effective visualization tool for communicating the state of the meta game. However, it does not communicate trends over time or give understanding as to why certain heroes are strong. We argue that sites like DOTA Buff provide fantastic detail about trends in the meta game, but often fail to provide useful curation and visualization of the statistics for a lay audience. It's telling that even with all of the statistics available a click away, DOTA Buff continues to publish their own set of tier lists monthly, rather than using an integrated solution.

Meta Search attempts to integrate the quick and easy discovery of powerful heroes provided by tier lists with an elegant visualization of trends over time. By allowing user's to drill down into curated and useful statistics about hero performance we help interested users go from a simple understanding of what heroes are strong to a more nuanced understanding of what they bring to the table. By keeping the visualizations simple and aesthetically pleasing we increase our ability to appeal to a mass audience, and hope to bridge the gap between passionate analysts pouring over detailed stats on DOTA Buff and the casual observer chatting with there friends about the most recent tier list.

3 Task and Data Domain

In order to support community development and discussion, Valve, the company who developed DOTA 2 provides access to a public web API that provides users detailed statistics about every match played on their servers. For privacy reasons, users must opt in to this program by checking off a box in the options menu of the game itself. Many players are incentivized to do this through the personal statistic tracking application on DOTA Buff. For the purposes of Meta Search we have sampled matches every month for the last two years. This time series match record includes

1. The skill league of the match
2. When this match was played
3. The length of the game
4. The state of the objectives at the end of the game
5. Which team won
6. Which players participated

Each player record is also associated with a detailed set of statistics about their performance in that game. At the moment this includes for each player

1. The hero this player chose
2. How many other players this player killed or assisted in killing
3. How many times this player died
4. How much gold and experience this player earned
5. How much damage and healing this player did
6. How many NPC characters did this player kill
7. Which team this player belonged to

On a broad level, our application will focus on two main tasks. The first is abstracted as outlier detection in hero selection trends. With over a hundred heroes, players are often overwhelmed with the amount of choice given to them. Many players will turn to online tools such as a tier list to find a strong hero to play that gives them the best chance of winning. These players will search through the match history and find heroes and strategies that have unusually high win rates, popularity or ease of use. This feeds into a more broad trend analysis task. More competitive players will also want to understand, analyze and potentially predict how the meta game will evolve in the future. These players will wish to dig more deeply into detailed performance statistics and understand why the outliers occur. Within the context of our Meta Search system, we formalize these two abstract tasks into the following

T1 Discover trends and patterns of interest in the performance and popularity of specific hero roles

T2 Identify the most popular heroes within a specific role of interest

T3 Discover trends and patterns of interest in the performance and popularity of specific heroes within a role of interest

T4 Drill down into an interesting hero to gain more detailed statistics and understand why these trends occurred

T5 Identify sudden shifts and anomalies in hero selection that may correspond to improper hero balance

3.1 Scalability Concerns

It is important to note that for every match analyzed, the data contains ten highly detailed and space consuming player records. With millions upon millions of Dota 2 games being played within our set visualization period, the amount of data to process and record expands at a truly alarming rate. Providing the infrastructure to access such a large data set is simply impossible for Meta Search. However, one must note that the tasks Meta Search seeks to support are mostly focused on finding trends that would be reflected in a popular hero being played and picked consistently. Instead of trying to visualize the statistics in every match, Meta Search instead aggregates and transforms the data to reflect a statistical sample of total games played. Specifically we take counts and averages of the available statistics for a sample of one hundred matches at set time intervals to visualize. This data is processed in order to provide average popularity and result metrics for each hero and role during the time period in question. While the sample size seems small at first, one hundred games contains a thousand hero choices. If each of the approximately one hundred heroes were played evenly we would get approximately ten match results per hero to get an idea of average performance. However, they are not played evenly, so the more interesting, more frequently played heroes will get larger, more representative samples. Players don't generally seek to find and play unpopular or weak heroes. Thus we are not overly concerned with having a few statistical anomalies where a hero is almost never played in a given month and there detailed statistics may be skewed by sampling error.

4 Tool Design and Justification

The entry view of Meta Search shows our Hero Role Overview stream graph to complete task T1.

4.1 Hero Role Overview Stream Graph

This stream graph encodes the relative popularity of different hero roles over time on the Y axis using position on an unaligned scale and is shown in figure 1. This form of the stream graph was chosen as a simple, easy to understand trend metaphor with a proven track record of igniting popular discussion and analysis of visualization results. [2] It is noted that relative comparison of stream sizes are somewhat inaccurate due to the lack of a common scale. However, the defined tasks focus more on finding shifts over time and clear outliers in the trends than comparing two average items. It's not likely that it will be useful or required to decide which of two similar streams is larger. Users will only truly need to see that the relative size of a stream is becoming larger or smaller. Thus we accept the loss of precision as a cost of reinforcing the change over time metaphor in our application.

Ideally this view would use hue to categorically encode what each stream represents, but there may be arbitrarily many roles a hero could perform. Categorical hue encoding is recommended to not exceed six to twelve bins. [6] We found having borders between the streams to visually distracting, so decided upon a repeating series of hues is to differentiate the different streams within the graph. The main drawback of this approach is that users are unable to talk about example streams by name like they could with categorical hue. However, once we ruled out categorical hue as an option due to scalability concerns we decided upon repeating hues as the least problematic option available. As is common practice with large area marks, the colours chosen are de-saturated in order to avoid distracting the users. [6] Rather than clutter the stream graph with a large number of labels, they are displayed using tooltips as users mouse over outliers of interest. To support task T2 a single click on a role of interest will populate a tier list on the right side of the graph. This gives a quick listing of the most popular heroes that can perform that role. Users may click and drag the mouse over the display of the available months to zoom the view of the graph into a particular area of interest. To enforce the trend analysis nature of the task, at least four months must be selected. The system will automatically add additional months to the display if needed. A drop down list of metrics to visualize trends for is displayed on the right of the stream graph, and the system supports animated transitions between them to help the user track a role of interest. This feature is particularly useful as a quick way to check if a particularly popular hero is just a fan favourite, or legitimately has a high win rate.

4.2 Hero Role Details Stream Graph

To support task T3 and T5 users may double click on any role within the Hero Role overview to bring up a stream graph of all the heroes with the selected role. This stream graph is shown in figure 2. While largely similar to the hero role overview, this detail view supports another interaction scheme to complete task T4. A single mouse click on a hero of interest will swap out the tier list with a box displaying the average results of that hero in the month the mouse was hovering above. In addition this also adds that date to the set of dates being compared in the detailed small multiples matrix view. This allows users to drill down into deeper detail about a heroes' performance in a month of interest and try to understand why they got more or less popular. Like the hero role overview, you can also visualize a variety of statistics about the heroes, and animated transitions between different comparison metrics are supported.

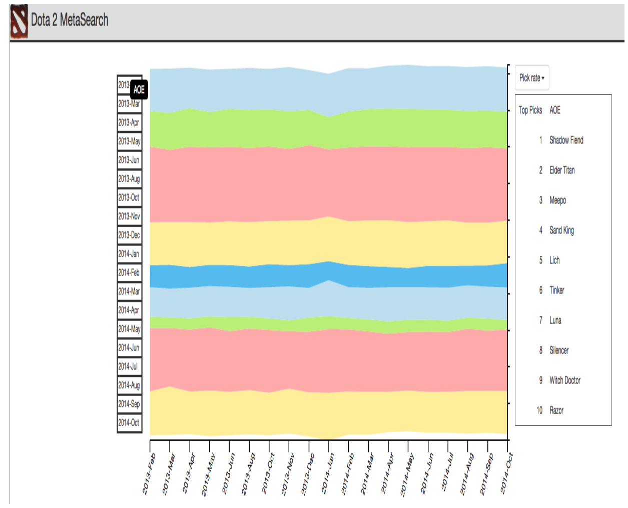

Figure 1: The overview of hero roles over time within our hero role overview. A moused over category of interest has a tooltip displayed, and has been clicked to bring up a tier list of heroes within this category

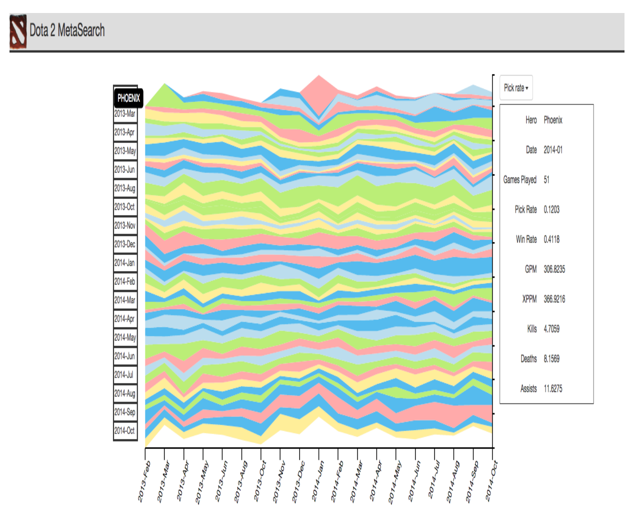

Figure 2: The stream graph of hero popularity within a role. A hero is moused over showing a tooltip of there details. This hero was clicked to populate the details view to the right.

4.3 Hero Stats Small Multiples

While stream graphs are an effective and simple metaphor for understanding how a single variable changes over time, they are unable to convey patterns within multivariate data. Even with the implemented animated transitions it was slow and cumbersome to compare trends in several statistics. Thus a small multiple scatter plot matrix is juxtaposed beneath the stream graphs to display data about many statistics at once. This small multiples view is shown in figure 3. This view is intended for more advanced analysts who will not be put off by having to learn a more complex interface. A scatterplot matrix of variable correlations was also considered and implemented. However, the insights gained from this view did not help users accomplish our tasks. Furthermore, some users found the results confusing so it was ultimately removed from the application to keep our views reasonably user friendly.

Each mark on these scatterplots represents the average value of a hero for one month. Hero ID is encoded at a consistent position along the X axis of every graph. This allows users to quickly glance between the different graphs and instantly know where to find the points for an interesting hero. Again, since we are looking to support outlier identification, details about a data point are provided on demand using a on hover tooltip. The scatterplot matrix compares up to three months worth of data, which was the upper limit before occlusion in our data became problematic. Hue is used to categorically encode which month a specific data point belongs to, and an intuitive legend is provided by colouring the borders of the date selector to the top left. An intuitive set of filtering controls is provided by a linked brushed highlighting approach. On a graph showing an interesting outlier users can click and drag a selection box over elements to investigate. Data points outside of the selected set are rendered with reduced opacity to enable pop out of the selected items. This selection set is shared between all graphs, and automatically updated, allowing users to compare multiple metrics for a set of interesting heroes very quickly. To further support cross graph comparison, the rows of the scatterplot matrix were carefully chosen to visualize similar metrics on each row. Specifically the metrics in each row of the scatterplot matrix were chosen to have similar scales, and the rows most likely to be interesting for analysis are placed at the top.

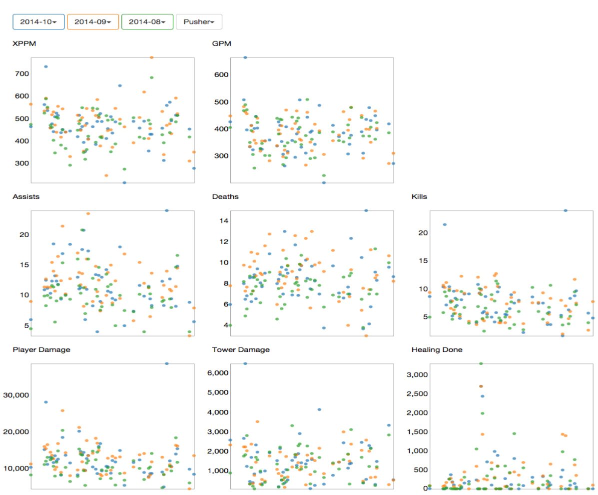

Figure 3: The scatter plot matrix showing statistics for all heroes within one role.

5 Implementation

Meta Search is broadly broken down into two main components. The back end database managing and sampling data from the match history API, and the front end page that displays the visualization.

5.1 Back End

The Meta Search back end builds on top of the python based Django web framework. Django with a clean and well documented query set API to manage a platform independent database out of the box. For development purposes we utilized a simple SQLITE3 database on our computer filesystem and the default Django web server. For production deployment these would have to be swapped out with more advanced solutions like Oracle and Apache Tomcat. All back end code to manage the web server, process requests and serve match history results to the visualization were written in Python. We found that queries on the set of matches were taking an extraordinarily long time to complete with our development back end, so we implemented a rudimentary caching scheme for our match history results. We essentially preprocess the set of anticipated queries, serialize the result into a JSON text file, and serve that result to the client instead of running a dynamic query. This scheme provided near instantaneous loading times on our local machines, but is only usable because our data is updated on a monthly basis by a separate script.

5.2 Data Scraping

In order to speed up client loading we create a local database cache of Match History data from Valve's public web API. A simple set of python scripts repeatedly calls this web API and puts the results into our database. Obtaining valid data from the Match History API proved to be a much greater challenge than anticipated. In order to get a list of games played, most available documentation, guides, and tutorials on the subject suggest the GetMatchHistory call. However, due to performance issues Valve silently remove most of its functionality from this call in the last few months. As of time of writing, you can only get five hundred of the most recent games, and you are prevented from using any of the date filters to search further back. Even the official API parameter documentation does not note this fact. A lesser known, slower API called GetMatchHistoryBySequenceNumber is still functional. However, this API is designed to slowly get a sequential list of every game ever played. There is no ability to select a target date, or otherwise filter the matches. However, the game numbers are sequential and temporally increasing. Our script would take an ID from the most recent month and estimate how far back the next set games we care about is. The resulting query tells us when the games returned were played allowing us to correct our guess and slowly search until we find the data we care about. With millions of games played in the database, and a single query for 100 matches at a time (the maximum amount allowed per query) one cannot hope to analyze every game ever played. Thus the script simply samples a small subset of one hundred matches every month. This seemed to give us a large enough sample size to see a number of expected patterns. Running this process for two years of data could take upwards of an hour. To enable us to transform the data later on without running the script again, the whole match and player records are stored. However, they are later processed to provide the aggregate trend data we visualize.

5.3 Front End

The front end visualization of our site was built as a D3.js application. Base examples for stream graph's and scatterplot matrices were taken from the D3 wiki gallery and then heavily customized for our application and data set. Most of the customization focused on providing the new set of interactions and links between views that were not provided.

6 Results

As part of our evaluation we showed Meta Search to a number of DOTA 2 players on campus and asked them to provide any feedback they could. We also posted the tool on various DOTA 2 fan forum sites in order to get feedback from a broader audience. For the most part responses were positive, but a there were a number of concerns raised around styling and usability. In particular the date filtering widget on the stream graph caused some confusion due to its alignment and thus implied relationship with the Y axis. Also, some users were confused by our choice to start with a stream graph of hero roles. It was suggested that a more intuitive interaction would be starting with a visualization of the heroes and just having a drop down of roles to filter by. Several users were intrigued by seeing how the release of new heroes temporarily effected the meta game and were curious if we could identify the effect of other balance patches. In order to further valuate the tool, we present a series of case studies showing interesting analyses and conclusions we were able to form about the hero selection meta game in DOTA 2 using Meta Search.

6.1 Static Team Composition

One of the most interesting and perhaps unfortunate conclusions we were able to draw is that the overall composition of a team in DOTA seems to be fairly set. Referring to figure 1, we see that there are only a few subtle shifts in the popularity of different roles over time. By and large teams seem to pick the same ratio of pushers to junglers to assassins and so on. Drilling down into the initiator category in figure 4, we see a great deal of variance in the popularity of specific heroes. Thus we must conclude that teams generally always need initiators, but specific initiators fall in and out of favour. We also see a few exceptions like Pudge that are always popular, which indicates that he may be a fan favourite hero.

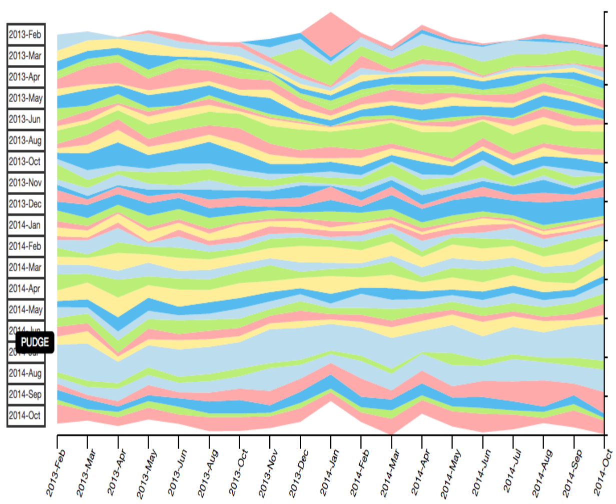

Figure 4: The stream graph of Initiators over time shows a number of heroes coming in and out of favour, the large light blue bars at the bottom indicate a few consistently popular fan favourites

6.2 New Hero Popularity

Another interesting observation is that the stream graph is organized so that heroes that were more recently released are displayed towards the top of the graph. Occasionally one sees a previously unheard of hero having a sudden spike in popularity followed by a dramatic fall off. Referring to the stream graph of heroes in the pusher category shown in figure 5, we see a total of three such spikes. A green one at the start of our recorded data for Bristleback, a red one in the middle for Phoenix and a very small light blue one for techies in the last two or three months. These spikes align with when the heroes were first released to the public and show the perfectly expected trend of a sudden surge in popularity as people try out the hot new hero, followed by a return to normalcy afterwards. Interestingly, the most recent bump for the release of techies is extremely small compared to the other two. This indicates that for whatever reason, people were generally not very interested in playing the hero, even when he was new. From the authors personal experience, techies is an interesting and unique hero, but is very difficult to play well, and thus may not have been very popular upon release.

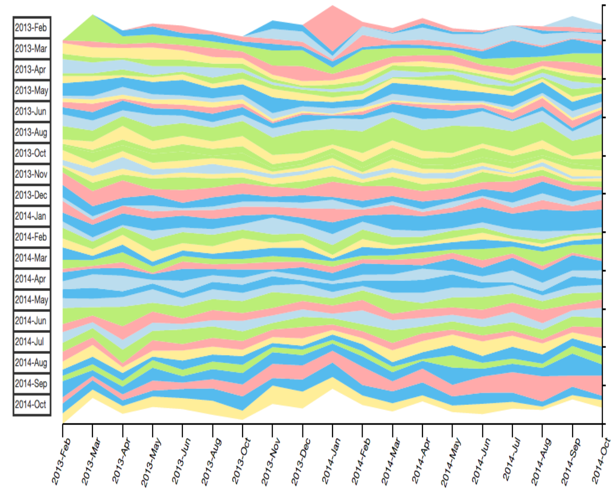

Figure 5: The stream graph displaying heroes that push enemy objectives show a number of sudden spikes in the upper streams that represent new heroes being released

6.3 Hero Popularity Reflects Performance

The default overview of Meta Search displays hero popularity over time. One question we asked was whether trends in hero popularity correlate with actual changes in the effectiveness of a particular strategy, or are due to random fluctuation. To answer this, we investigate the stream graph of all support heroes in figure 6. We see that that in the last two months silencer has seen a significant rise in popularity. We then drill down into the small multiples of statistics for support heroes in figure 7. After finding the marks for silencer, we drag a selection box around them. Now that the values pop out we see a clear pattern. Silencer has seen a significant rise in gold earned, experience earned, kills and assists received, and only a minor increase to the overall deaths. It seems like silencer indeed has become a stronger hero in more recent data. Unfortunately from this data it is impossible to really tell if that is due to a balance change, some new guide for silencer being posted, or some other factor. Even so, Meta Search has provided at least some justification for exploring and practicing the hero, and digging deeper into why he is currently so strong.

6.4 SubCategory Identification

A strength of visualization tools in general is the ability to find outlier values and notice erroneous data. Looking at the small multiple view of support heroes in figure 8 we see an interesting outlier value for the hero Wraith King. We hypothesize this is because while Wraith King provides a great deal of support to his team, he is often played as a carry. One generally expects carry heroes, which obtain as much gold as possible to become a late game threat, to earn more gold and experience than the heroes that support them. Thus, Wraith King's dual, contradictory role causes him to not fit neatly into either category. Perhaps a new taxonomic group needs to be created for such hybrid heroes, or a method needs to be devised to identify which of the two roles a hero is playing in any given game we analyze.

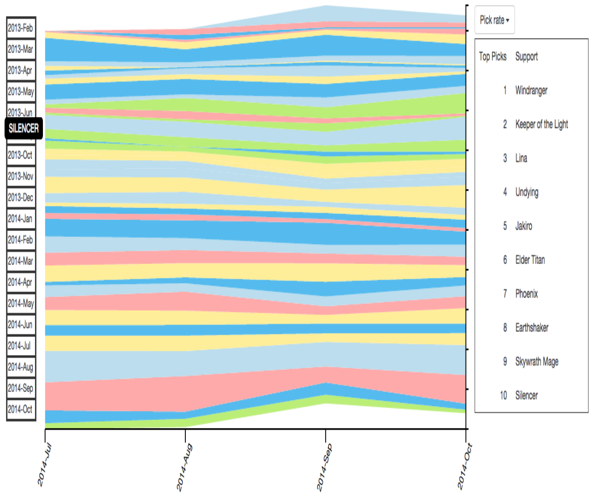

Figure 6: The stream graph displaying support heroes seems to show that the silencer hero is becoming more popular within the last few months

Figure 7: Drilling down into the detail view, we compare the three most recent months of data, and highlight the dots that correspond to silencer. We note that the more recent dots are consistently higher for most metrics

7 Discussion

Overall Meta Search seems to be an extremely useful tool for generating surprising new questions and providing evidence to support conclusions about the DOTA 2 meta game. We argue that it successfully allows novice users to easily understand the current state of the meta game, while providing experts the tools to dig into deeper analysis. Thus Meta Search successfully bridges the gap between your standard hero tier list, and a in-depth statistics application like DOTA Buff. This space is currently underserved in the community, and Meta Search is perfectly suited to fill this hole.

7.1 Overview Stream Graph

The stream graph visualization proved to be a strong and useful metaphor for trends in hero popularity over time. It helps reinforce the temporal nature of the meta game and supports fast, reasonably accurate computation of the most popular heroes at specific points in time. It is sometimes hard to compare two streams of similar size, especially at disjoint areas in the graph because of the lack of a common scale. However, when trying to understand which heroes are the strongest, knowing that two heroes are roughly similar is arguably good enough. The stream graph also scales impressively well as the amount of data increases. The nature of the graph allows it to scale arbitrarily far into the past along the X axis, and our query result caching scheme allows us to provide near instant loading and lag free interaction. The desaturated colours proved to not only be useful for avoiding distraction, but also very aesthetically pleasing.

The biggest issue identified with our stream graph implementation seems to be confusing or unclear interactions and filtering controls. Our taxonomy of hero roles proved to not provide sufficiently interesting trends, and serves mainly as a filter on which heroes to visualize. Some users were confused by this dynamic and would have preferred being able to see at least some set of interesting trends on page load. There are a number of potential solutions to this problem, such as restricting the matches analyzed to a more controlled and skilled population or swapping out the filter with a drop down list of categories. The widget to filter what dates are shown confused a number of users. This is potentially because it is aligned with the Y axis, causing some to think they were actual axis ticks instead of a control. A simple scroll wheel zoom may provide a more intuitive interface. Ideally this would also reduce the time period between meta game samples, but this may be impossible with our current infrastructure.

7.2 Small Multiples Matrix

The small multiples scatterplot matrix proved to be useful for investigating more complex multi-variate relationships in performance metrics. This is naturally a less common, and more complex task than analyzing trends in pick rates. However, this is still highly important for advanced users and transitioning between the metrics on a stream graph would be too cumbersome. The linked selection box proved extremely powerful for finding related points on several graphs. Occlusion sometimes caused it to select more points than intended, but this could be fixed with some simple zooming controls. Using opacity to create a pop out effect was also very intuitive and effective given proper viewing conditions, but it was noted the lower opacity dots were invisible on lower quality monitors. Allowing users the ability to control opacity levels would easily have relieved this issue.

There are a number of shortcomings and limitations with our approach to the scatterplot matrix. Currently the workflow is that users investigate the matrix to find an interesting point and then hover the mouse over it to figure out which hero it corresponds to. However, this view is intricately linked to the stream graph, so it is very reasonable for a user to have already found a hero of interest on the streams. Simply put, the only option to find the dots corresponding to that hero is to painstakingly hover over every column of dots. Linked highlighting between the two views is important and should be implemented. Our attempt to prevent occlusion by limiting the matrix to three days at once helped, but did not do enough. Some views were simply too cluttered to successfully analyze. Lastly the overall size of the matrix was extremely large. For an expert visualization analyst with a large expensive monitor, this is not a problem, but our target mass audience may be greatly reduced by this assumption.

7.3 Lesson Learned

The biggest thing we learned is that it is critically important in a design study to make absolutely sure you have full access to all the data you need before getting too deep into a project. Going into the project, we assumed that Valve's match history API was entirely functional. However, it had been quietly repurposed to only provide the five hundred most recent games. We did a great deal of work

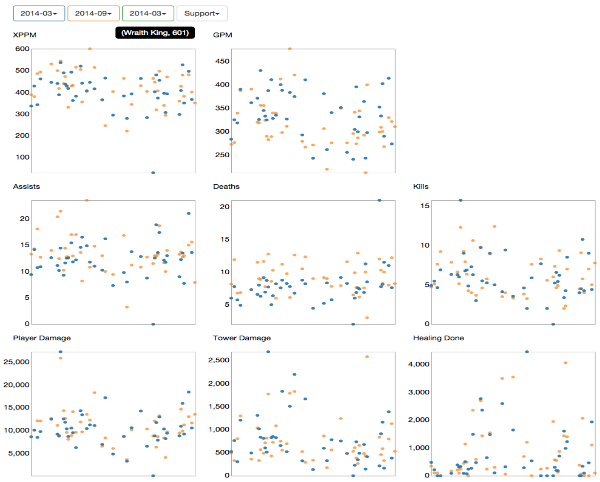

Figure 8: Looking at the detail view of supports we see a dot on the top of the graph showing Wraith King as a clear outlier case

before realizing this, and would not have been able to complete our project if we weren't able to find the undocumented replacement API. Additionally, we learned that it is good practice to develop with a static, local set of data. During the course of this project Valve released two major updates to DOTA 2 that introduced new Heroes and content. These updates briefly broke their web API and required updates to our data collection scripts. These issues with storing and recovering data definitely slowed our progress, and will be avoided in future work. Working with D3 also proved to be a significant obstacle, as neither author had worked with the tool before. However, D3 had a great deal of documentation and sample code available to get our project up and running reasonably quickly.

We also learned that we needed to budget a great deal more time than one would normally expect to data cleaning and analysis. In the second half of the project we started working with an iterative process. We started with a simple implementation of a new feature, then tested how our data worked with this new visualization. In almost every case we would discover anomalies in our data, such as special events impacting our metrics, missing samples, or corrupted days. Reversing our workflow, filtering or otherwise accounting for errors and then restarting development wasted a good deal of time that could have been spent on the visualization itself.

8 Future Work

The design and implementation process of Meta Search raised a number of interesting questions and tasks that should be completed in future work.

8.1 Additional Performance Metrics

The default match history API only records a single snapshot of player performance at the end of the game. This has a number of significant downsides and prevents one from asking many fine grain questions about how players were doing over the course of the whole match. For example, a hero might have a low win rate overall because they perform extremely well in the beginning of the game but drop off later. If a player knew this fact they could potentially devise a strategy to end the game quickly and make that hero successful. Detailed replay data of all the games in the match history API is available from another API but was not computationally feasible to process for a data set of our size. Future work should investigate processing a smaller set of matches and incorporating fine grain analysis of hero performance over the course of the entire game.

A more computationally feasible proxy for the full replay data would be expanding the tool to analyze the items and skills bought by the player. The match history record includes a list of the items the player had at the end of the game and the order in which the player upgraded their skills. The items players buy is often influenced not only by what hero they are playing, but who they are playing against and how they are doing in that match. For example, hand of midas is an expensive item that players consider greedy. It does not provide much strength to help you win the game, but it increases your gold income so you can potentially buy more items later. Buying such an item could be an indication of a match going particularly well for a player and them trying to snowball their character further. Analysis of these choices was outside of the scope of Meta Search, but could be very enlightening.

8.2 Incorporate Analysis of Skill Level

Since getting a full release last year, the DOTA 2 player base has quickly expanded. DOTA 2 is a highly complex game, and the learning curve for new players is extremely steep. Game theory analysis of player choice typically assumes that people are thinking rationally and strategically about their choices in relation to how they guess other players will act. [7] As new players are learning the game and overwhelmed by the sheer number of heroes, abilities and items, they often have very little knowledge about what strategies are strong, or what strategies are possible. Thus their strategic choices will seem illogical and irrational to expert players. This is problematic because our match sampling scheme does nothing to filter out these matches and assigns them the same priority as a match played in a professional tournament. Future work should use the skill league information present in the match history data to infer the skill of the different players. This knowledge should then be used to effect the visualization of trends through a weighting or filtering scheme on the matches. We hypothesize that the meta game in higher skill levels is more well defined, dynamic and changes faster.

8.3 Add Timeline of Game Updates

One of the major questions that was brought up in our evaluation of the tool was whether or not we could spot the impact of balance and content updates on the meta game. In order to answer this question we were forced to open a second page and refer to the DOTA 2 update history blog. While expert players with deep knowledge and experience with the game may remember this data already, this information should be incorporated into the main stream graph visualization. This could be done unobtrusively with a timeline along the X axis of the graph, the pops up greater detail about the update as mouse hovers over it. This will allow players to easily investigate how recent updates effected the meta game, and link their questions to what was actually changed in the update.

We also noticed that some of these patches have a significant effect on the semantic quality of our match statistics. A typical example for this is Diretide 2013 which temporarily modified the gold mechanic and corrupted our statistics for that month [1]. Outliers such as this can be corrected by filtering the matches visualized based on the game mode played, but one could argue they are part of the overall meta game. Having bars or lines related to major patches and events would at least allow users to note and correct for the effect of similar events.

8.4 Improve Hero Role Taxonomy

As noted in the results section, we realized that the relative prevalence of the the different hero roles remains fairly static. Currently,

[1] Diretide is an event that lasted from November 14 to November 28, 2013. During the event, the players were playing a different game mode where they would receive additional gold.

each hero is assigned to one or more categories, depending on the roles they can perform. However, many heroes can perform a hybrid of different roles, and can perform contradictory roles based on how they are played and what items they buy. The best example of this phenomena is a hero like Wraith King, who has a number of abilities that enable him to single handedly destroy the enemy team as a carry, or if built with a different set of items simply stand back and support his allies. We hypothesize that it might be possible to automatically infer which of several roles a hero is playing in a specific game by looking at which items and skills they purchased. Future work should investigate how accurate such a prediction would be, and test how the hero role visualization shifts if heroes are recorded using the role they performed in that specific game, instead of the set of roles they could have performed.

8.5 Incorporate Broader User Feedback

The feedback received so far on Meta Search has been relatively limited. Future work would ideally partner with popular fan sites like DOTA Buff to integrate Meta Search into popular discussion. This would provide a huge amount of feedback that could improve the tool to fit the needs of the community. It may also be useful to do a formal user study comparing the effectiveness of the most recent set of tier lists with Meta Search. Furthermore, the two tools could serve a complimentary purpose where Meta Search serves as the justification for why the current tier lists are valid or need to be updated.

Future work will also need to address the concerns that were brought up regarding usability and styling of Meta Search. The time period filtering control in particular could be scrapped or redesigned to be more intuitive. One suggestion was having simple mouse wheel zooming in addition to having more samples during the months. Another suggestion is allowing users to visualize trends in two disjoint time periods in juxtaposed or perhaps overlaid views. There are also a number of new performance metrics we could derive from the data to incorporate into Meta Search. In particular KDA or the ratio of kills, deaths, and assists would be useful to understand if a hero is providing good value to their team, or simply being killed too often.

8.6 Counter-pick Analysis

At the moment, our visualization assumes that players select heroes without sure knowledge of the enemy team's choices. This would be called a blind pick draft where each team tries to pick an independently strong strategy with the greatest likelihood of winning in the current meta game. However, there is no true blind pick draft mode in DOTA 2. The most popular mode, all pick, allows players to "lock in" their hero after discussing it with their team, which prevents them from changing their choice. However, the enemy team is notified whenever a new hero is locked in, and can adjust their future selections with this new knowledge. It is reasonably common practice to purposefully wait until the last minute to lock in selections in order to "counter pick" the enemy team. Successfully counter picking the enemy team can have a significant effect on the outcome of a game, and is one of the most common reasons to play weaker, situationally powerful heroes.

Future work might analyze possible variations in the pick strategy based on the presence of a specific hero in the enemy team. It would be of particular interest to compare the effectiveness of those variations. It is not yet certain that frantically counter picking the enemy at the last minute improves the success of a players hero selection strategy or if counter picking actually leads people to make more irrational choices when pressed for time.

9 Conclusion

In conclusion, this design study presents Meta Search, a novel visualization of match history and results from the popular video game DOTA 2. A simple statistical sampling and cached query method is used to minimize processing requirements and provide the scalability to visualize two years of data. The visualization itself utilizes juxtaposed overview and detail views to allow lay users to answer simple questions about which heroes are the strongest, while more advanced users are able to drill down into greater detail to understand why those heroes are strong. The overview uses a simple stream graph to improve upon existing concepts of a hero strength tier list with an understanding of trends over time. The linked detail view allows users to explore multi-variate patterns in the data through a small multiples scatter plot matrix. By providing identical X axis position for specific heroes on the graphs and efficient linked highlighting selection we allow users to quickly link related data points between many charts.

Meta Search was shown to a number of DOTA 2 players on campus, and a link to the prototype was posted on various fan forums. Initial user response has been reasonably positive. Most complaints centred around issues with styling and control usability. Most users expressed interest and enthusiasm in the tool and were excited about being able to see the effect of new patches, heroes and updates on the meta game. Meta Search has also provided insights into a variety of key questions about the DOTA 2 meta game. Meta Search showed that team compositions tend to maintain a set ratio of the different roles while swapping out the heroes to fulfill those roles, varying trends in new hero acceptance were observed, and increases in hero popularity were shown to correlate strong with increased effectiveness in that hero.

Future work aims to improve the usability of the tool further, expand upon the set of statistics available, investigate how trends differ according to the skill level of the players, and investigate player reception and widespread use of the tool. In an ideal world, Meta Search could be expanded to also process raw replay data in addition to recorded match statistics. This would enable us to do highly detailed analysis and account for heroes that can play multiple roles depending on which items they buy. As is, Meta Search provides a useful analytical tool that bridges the gap between a simple tier list and advanced statistics applications like DOTA Buff.

References

[1] T. Batsford. Calculating optimal jungling routes in dota2 using neural networks and genetic algorithms. *Game Behaviour*, 1(1), 2014.

[2] L. Byron and M. Wattenberg. Stacked graphs-geometry & aesthetics. *IEEE Trans. Vis. Comput. Graph.*, 14(6):1245–1252, 2008.

[3] M. Carter, M. Gibbs, and M. Harrop. Metagames, paragames and orthogames: A new vocabulary. In *Proceedings of the International Conference on the Foundations of Digital Games*, pages 11–17. ACM, 2012.

[4] A. Drachen, M. Yancey, J. Maguire, D. Chu, I. Y. Wang, T. Mahlmann, M. Schubert, and D. Klabajan. Skill-based differences in spatio-temporal team behaviour in defence of the ancients 2 (dota 2).

[5] G. McEwan, C. Gutwin, R. L. Mandryk, and L. Nacke. Analysing social metrics in an online game site. *GRAND 2012*, 2012.

[6] T. Munzner. *Visualization Analysis and Design*. CRC Press, first edition, 2014.

[7] M. J. Osborne and A. Rubinstein. *A course in game theory*. MIT press, 1994.

[8] J. H. Tom Giratikanon and J. White. Watch 10,000 league of legends games in 30 seconds. *The New York Times*, October 2014.

[9] M. Wattenberg and J. Kriss. Designing for social data analysis. *Visualization and Computer Graphics, IEEE Transactions on*, 12(4):549–557, 2006.

[10] B. G. Weber and M. Mateas. A data mining approach to strategy prediction. In *Computational Intelligence and Games, 2009. CIG 2009. IEEE Symposium on*, pages 140–147. IEEE, 2009.