CS579 Project Report

DOTA SCIENCE

Adithya Sreenath Chandrashekarapuram

Department of Computer Science

Illinois Institute of Technology

April 18, 2018

Abstract:

In this project, various factors which affect a Dota 2 game such as kills, deaths, assists, gold per minute, experience per minute and combination of characters picked are analysed and presented in the manner of graphs. These graphs can be used to see which factors play a major role and what a player must focus on to win a game.

Introduction:

Dota 2 is a popular MOBA (Multiplayer Online Battle Arena) game which has a 5 VS 5 Tower defence game style. There are various factors that affect the outcome of the game. These factors can be analysed and players can look at the graphs and see which areas to focus on to achieve a win.

Data:

All the data is gathered from the Dota 2 Web Api. Real time data can be collected using the getmatches.py, getplayermatch.py. Otherwise the data gathered by me can be used to test the programs. The output is then stored as JSON files for access at a later stage.

Method:

The first method used to calculate the win rates, kills, deaths, assists, gpm, xpm percentages involve

```
Win Perc = total wins/ (total wins + total losses)

Kill perc = Kills when win/(Kills when win + kills when lose)

Death perc = Deaths when win/(Deaths when win + Deaths when lose)

Gold_per_min[i] = (total gold at the end of game/total game time)

for i in Gold_per_min[i]:

    sum = sum + gold_per_min[i]

    total_matches = total_matches + 1

Gold perc = sum/total_matches
```

Similarly Experience ,Kills/Deaths per loss are calculated.

The second method used is to determine the hero combinations that lead to a win in a Dota 2 game.

1. First we create a co-occurrence network:

   For each player, loop through every match

   For each match, increment the edge weight of the hero selected by the current player and heroes selected by the other nine players.

   for each individual match:

   for every player except current player:

   [hero[current_player]][hero[other_player]][weight] =+ 1

2. Creation of graphs based on edge weight is performed:

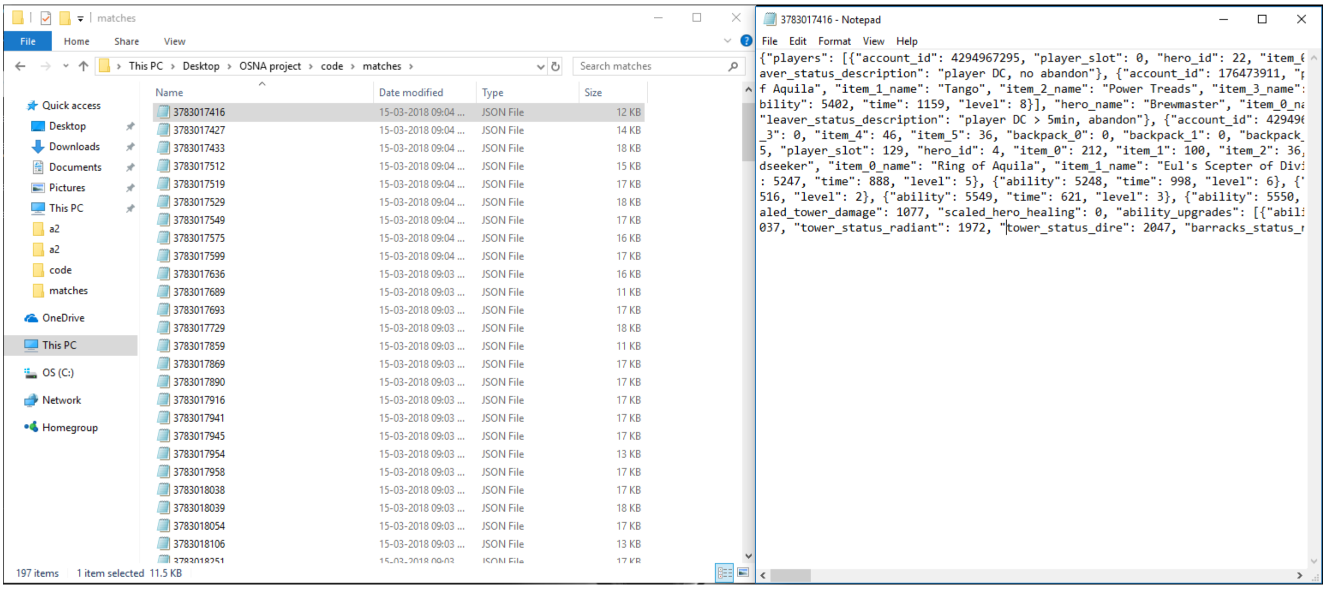
   Nodes are created for each hero. These nodes are represented by their hero id in the graph.

   The graph is converted to a fully connected graph.

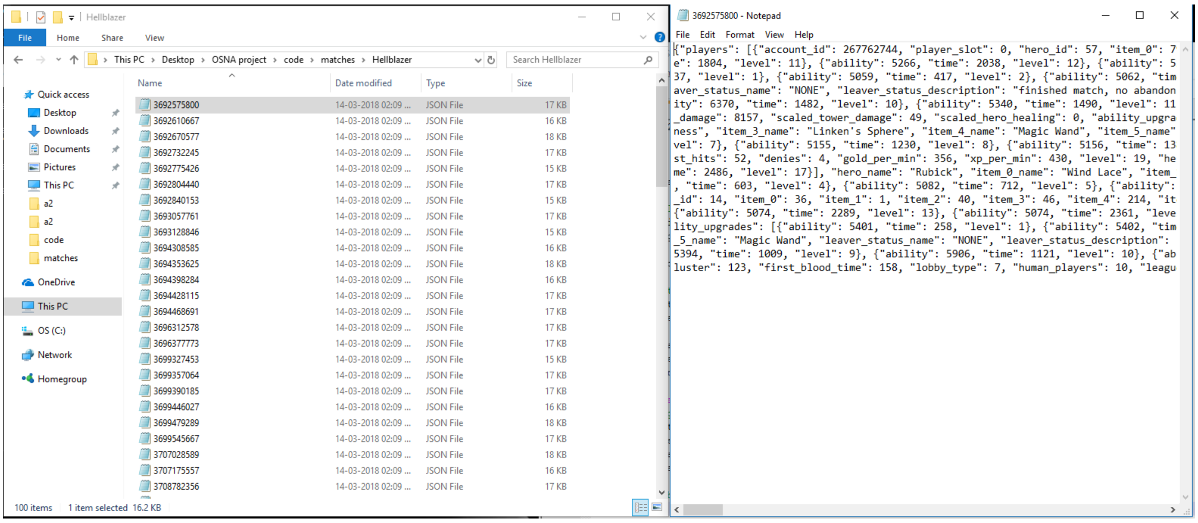
   The edge weights for each edge gets incremented if the same end-points (heroes) co-occur in the same game and the game resulted in a win.

Results:

Gathering data of recent matches and storing them in JSON files. (These files do not contain in depth data but just the general details)



Gathering match details for one single player. (100 games)



Gathered the hero details of all 120 heroes present in the game. Each hero is represented with a unique id to provide better representation in graphs.

HERO LIST
{1: 'Anti-Mage', 2: 'Axe', 3: 'Bane', 4: 'Bloodseeker', 5: 'Crystal Maiden', 6: 'Drow Ranger', 7: 'Earthshaker', 8: 'Juggernaut', 9: 'Mirana', 11: 'Shadow Fiend', 10: 'Morphling', 12: 'Phantom Lancer', 13: 'Puck', 14: 'Pudge', 15: 'Razor', 16: 'Sand King', 17: 'Storm Spirit', 18: 'Sven', 19: 'Tiny', 20: 'Vengeful Spirit', 21: 'Windranger', 22: 'Zeus', 23: 'Kunkka', 25: 'Lina', 31: 'Lich', 26: 'Lion', 27: 'Shadow Shaman', 28: 'Slardar', 29: 'Tidehunter', 30: 'Witch Doctor', 32: 'Riki', 33: 'Enigma', 34: 'Tinker', 35: 'Sniper', 36: 'Necrophos', 37: 'Warlock', 38: 'Beastmaster', 39: 'Queen of Pain', 40: 'Venomancer', 41: 'Faceless Void', 42: 'Wraith King', 43: 'Death Prophet', 44: 'Phantom Assassin', 45: 'Pugna', 46: 'Templar Assassin', 47: 'Viper', 48: 'Luna', 49: 'Dragon Knight', 50: 'Dazzle', 51: 'Clockwerk', 52: 'Leshrac', 53: "Nature's Prophet", 54: 'Lifestealer', 55: 'Dark Seer', 56: 'Clinkz', 57: 'Omniknight', 58: 'Enchantress', 59: 'Huskar', 60: 'Night Stalker', 61: 'Broodmother', 62: 'Bounty Hunter', 63: 'Weaver', 64: 'Jakiro', 65: 'Batrider', 66: 'Chen', 67: 'Spectre', 69: 'Doom', 68: 'Ancient Apparition', 70: 'Ursa', 71: 'Spirit Breaker', 72: 'Gyrocopter', 73: 'Alchemist', 74: 'Invoker', 75: 'Silencer', 76: 'Outworld Devourer', 77: 'Lycan', 78: 'Brewmaster', 79: 'Shadow Demon', 80: 'Lone Druid', 81: 'Chaos Knight', 82: 'Meepo', 83: 'Treant Protector', 84: 'Ogre Magi', 85: 'Undying', 86: 'Rubick', 87: 'Disruptor', 88: 'Nyx Assassin', 89: 'Naga Siren', 90: 'Keeper of the Light', 91: 'Io', 92: 'Visage', 93: 'Slark', 94: 'Medusa', 95: 'Troll Warlord', 96: 'Centaur Warrunner', 97: 'Magnus', 98: 'Timbersaw', 99: 'Bristleback', 100: 'Tusk', 101: 'Skywrath Mage', 102: 'Abaddon', 103: 'Elder Titan', 104: 'Legion Commander', 106: 'Ember Spirit', 107: 'Earth Spirit', 109: 'Terrorblade', 110: 'Phoenix', 111: 'Oracle', 105: 'Techies', 112: 'Winter Wyvern', 113: 'Arc Warden', 108: 'Underlord', 114: 'Monkey King', 120: 'Pangolier', 119: 'Dark Willow'}

For an individual player the number of times each hero was picked in the 100 games is calculated and displayed below.

```
HERO COUNTS
Anti-Mage 15
Axe 8
Bane 4
Bloodseeker 10
Crystal Maiden 7
Drow Ranger 5
Earthshaker 13
Juggernaut 26
Mirana 12
Shadow Fiend 39
Morphling 8
Phantom Lancer 9
Puck 6
Pudge 45
```

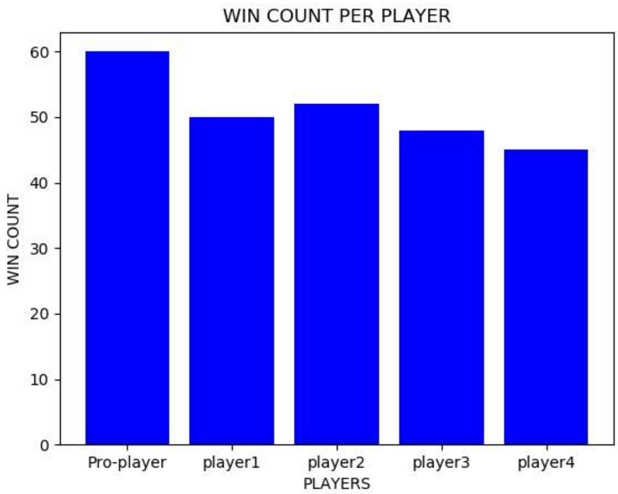
```
Razor 4
Sand King 15
Storm Spirit 13
Sven 9
Tiny 7
Vengeful Spirit 5
Windranger 5
Zeus 9
Kunkka 18
Lina 7
Lich 3
Lion 8
Shadow Shaman 15
Slardar 7
Tidehunter 4
```

```
Witch Doctor 4
Riki 5
Enigma 6
Tinker 25
Sniper 15
Necrophos 11
Warlock 0
Beastmaster 2
Queen of Pain 10
Venomancer 5
Faceless Void 23
Wraith King 6
Death Prophet 3
Phantom Assassin 11
Pugna 7
```

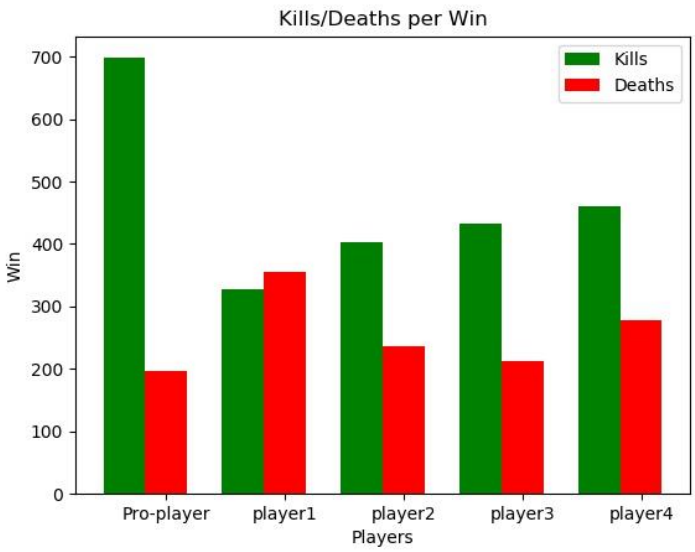
Analysis:

Win Ratio:

The win count for 5 different players (1 pro player and 4 regular players) is analysed in the graph. We can clearly see that the win rate for the pro player is the highest among the 5 players. So we will be using this as a basis for understanding how the four regular players can achieve the same win rate.



Kills/Death counts per win:



The number of kills and deaths achieved by the pro player in the games which resulted in a win is analysed in the graph.

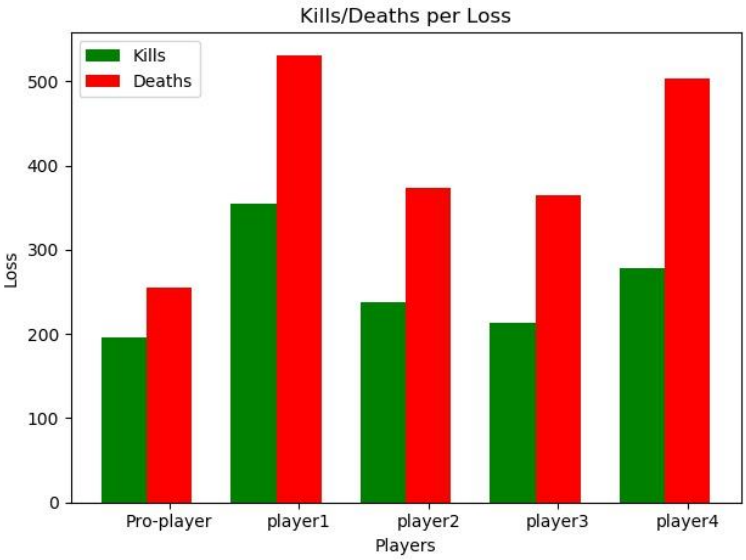
The number of kills for the pro player is very high compared to any of the other regular players. Hence u can infer that higher kill counts in a game would result in higher win rate.

Kill count α Win rate

Similarly the number of deaths of the pro player is the least among the players, so it we can infer that lower death count in a game results in higher win rate.

Death count α 1/Win rate

Kills/Death counts per loss:



The number of kills achieved by the pro player is lesser than most of the regular players but we can also see that the death count is the least among all the players. This further strengthens the inference provided by the kills/deaths per win graph.

Gold and Experience:

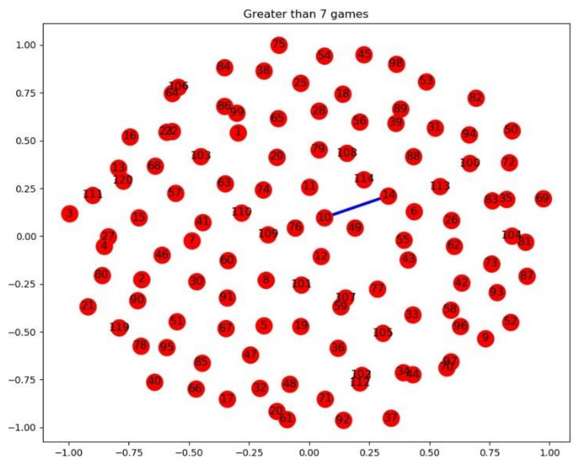
Here we analyse the gold and experience achieved per minute in the games. We can see that the pro player has the highest gold and experience gained among all the players. This allows us to infer that the gold and experience is directly related to win rate.

Gold per minute α Win rate

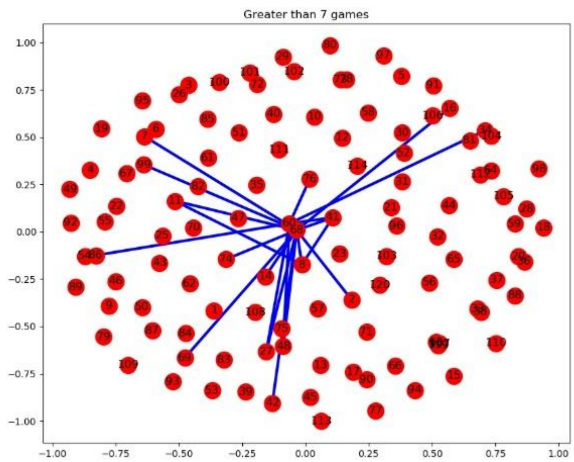
Experience gained per minute α Win rate

Hero Pair Co-occurrence:

In this section we analyse the hero pair co-occurrence. The nodes represent the hero ids, there exists an edge between two nodes if the hero pairs are present in more than N games.



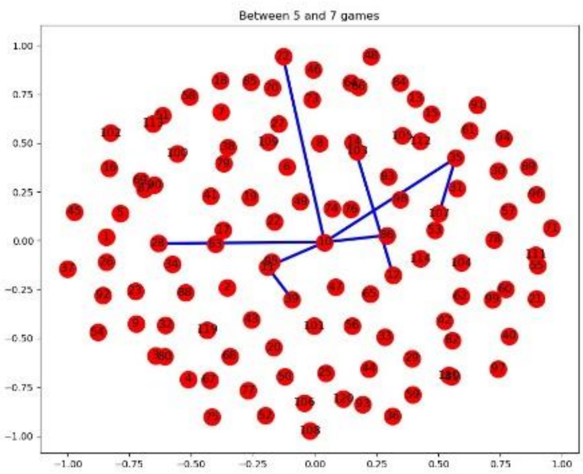
Pro



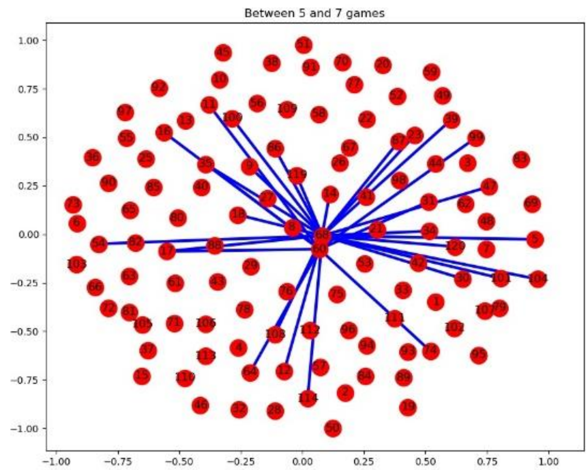
Regular

For the games played by the pro player we can clearly see that lesser edges exists compared to the numerous edges of the regular player.

We can infer from this that for the games played by the pro player a wide variety of heroes were picked. As the hero pair co-occurrence is less compared to the regular player.



Pro



Regular

Conclusion:

After analysis of various factors which affect a Dota 2 we can state that for a player to win a game he must:

- Increase the number of kills per game
- Decrease the number of deaths per game
- Improve the gold acquired per minute in every game
- Improve the experience gained per minute in every game
- Play a wide variety of heroes to avoid being predictable and keep the opponents guessing

Improvements:

Girvan-Newman partition algorithm can be implemented to split the heroes into communities based on hero pair co-occurrence (This is already implemented in the herocom.py file and commented out. This code was unable to be tested due to lack of match data, the result of 100 games were single node communities. Gathering match data upwards of 10000+ games would result in visible communities being formed.)

Spatio temporal evaluation of team movement can be performed by analysing the replay files and tracking movement patterns of teams.

A predictive model can be designed which takes in match data and predicts the outcome of a game by usage of the metrics provided.

Related Work:

https://cseweb.ucsd.edu/~jmcauley/cse255/reports/wi15/Kaushik_Kalyanaraman.pdf -This paper is used as a base paper for the project. Analysis is done by understanding and applying the ideas provided by the paper.

http://www.sloansportsconference.com/wp-content/uploads/2016/02/1458.pdf - This paper provides an in-depth method of analysing Dota 2 and other MOBA games based on encounters from replay files.

https://arxiv.org/ftp/arxiv/papers/1603/1603.07738.pdf - In this paper different metrics are used to evaluate the games based on spatio-temporal team behaviour.

https://people.engr.ncsu.edu/dlrober4/papers/fdg14-combat.pdf - This paper has a different methodology to analyse the game metrics but does not focus on metrics required to win a dota 2 game. Here the data is gathered via online communities.

http://ieeexplore.ieee.org.ezproxy.gl.iit.edu/xpls/icp.jsp?arnumber=6642858 – This paper using similar analyses techniques but measures different metrics to analyse the game generally but not what metrics are required to win.

References:

1. https://cseweb.ucsd.edu/~jmcauley/cse255/reports/wi15/Kaushik_Kalyanaraman.pdf

2. http://www.sloansportsconference.com/wp-content/uploads/2016/02/1458.pdf