Dota 2: Win Probability Analysis

Tianhong Ding and Chi Li
Department of Statistics
Columbia University
td2488@columbia.edu, cl3406@columbia.edu

Abstract — **Defense of the Ancients 2 (Dota2) is one of the most popular multiplayer video games around the world. In this paper, we use a dataset containing records of 97230 matches to solve two problems: 1. What's the best model to predict winning side; 2. What can we do to increase winning probability. We use TensorFlow neural network library to learn information from both sides and then decides which side will win. We also use Logistic Regression model and Multinomial Naïve Bayes model to do prediction. In order to increase win probability, we use recommendation algorithm to discover the popular heroes chosen by winners.**

Keywords- Dota 2; TensorFlow; Neural network; Machine learning; Recommendation algorithm

I. INTRODUCTION

In recent years, the E-sports environment around online digital games has gained immense momentum. NewZoo [1] reported that in 2016, the total E-sports audience in the US would reach 41.7 million, with 20 million E-sports Enthusiasts. The vast majority of these are males (72%) and millennial make up 48% of the enthusiast audience and over two thirds of E-sports Enthusiasts in the US are active in a team sport versus 26% of the online population. On the company side, considerable resources are being allocated to support the e-sports environment from the main companies in the domain such as Riot Games, Wargaming, Valve, Ubisoft and Turbine [2]. In 2016, prize money for Dota 2's main tournament of the year, The International, awarded the biggest prize pool in E-sports history at over $20 million, surpassing the record set at the previous International in 2015. The event, seen by millions of viewers globally, was considered to be one of the greatest E-sports tournaments of all time [3]. In short, Dota2 are today among the most played games in the world [4].

Dota2 was developed by Valve in 2003 and has been continuously gaining speed and strength since then in the bewildering Internet game field. It possesses enormous player basis as well as game data, therefore are of interest from the perspective of sports analysis [5] and game analytics [6] in general as a major e-sport, backed by a growing industry.

In this paper, we present several different analyses of player behavior in the MOBA Dota 2, including Logistic Regression, Multinomial Naïve Bayes and Neural Network.

II. RELATED WORKS

In recent years, there has been a wealth of publications that analyze player behavior in Dota2 in different aspects. Focus will here be on work directly on analysis of winning prediction.

At present, there exist two heroes picking guiding system available to players, namely, Dota2cp and Dota Picker. Dota2cp similarly applies Logistic Regression, and a correct rate of more than 50% has been reported, while Dota Picker gives hero pick guidance according historical win-and-lose results.

As for machine learning algorithms, Atish and Michael did a project on Dota2 which only considered team composition in 2014 [7]. They predicted winning side by looking at how both sides pick their heroes including interactions between heroes. However, they found out that after incorporating hero interaction, the prediction rate is lower than the model with out hero interaction. Their insight was that either the model needed to be fined or the data they collected were not good enough. In 2013, there was another project [8] presented by Kevin and Daniel where they did an survey on the machine learning algorithm that had been applied to Dota 2 prediction, and then made a recommendation engine that helps the player pick a hero to maximize winning rate.

III. SYSTEM OVERVIEW

Dota2 matches feature matches between two teams, Radiant and Dire, both of which consist of five members and aim to defeat the opponent by destroying its Ancient, a well- protected core camp. In the game, all players are represented by different identifications-known as Heroes-picked up by themselves as soon as they start the matches. Heroes are identified by multidimensional characteristics, the levels of which demonstrated their capabilities in different aspects.

Before the game, each player can pick a hero as the character from a pool of 113 playable heroes, with each hero having its own "Strength", "Agility" and "Intelligence". Each hero can only picked once in each game. Each hero has a set of features that define his role in the team and play style. Among these features there are his basic attribute

(Strength, Agility or Intelligence) and unique set of 4 (or for some heroes even more) skills. These features allow each hero to fill several roles in the team, such as “damage dealer" (hero, whose role is to attack the enemies in the fight), “healer" (hero, who mostly heals and otherwise helps his teammates), “caster" (hero, who mostly relies on his spells) etc. Besides attributes each hero can buy items to increase their characteristics or give additional abilities. A set of items purchased by player can significantly affect the success of his individual gameplay and the result of a match.

The training set consists of matches, for which all of the in game events (like kills, item purchase etc.) as well as match outcome are know. You are given only the first 5 minutes of each match and you need to predict the likelihood of Radiant victory.

Players can get gold and experience for killing other people's characters or other units. Experience affects the character level, which in turn makes it possible to improve the capacity. For the accumulation of gold players buy items to improve the characteristics of the characters or give them new abilities.

After the death of the hero is sent to the "tavern" and revived only after some time, so the team loses the player for a while, but the player can early redeem the character of the tavern for a certain amount of gold.

During the game, the team develops its heroes, defending his part of the field and attack the enemy.

The game ends when one team crushes a certain number of “towers” of the enemy and destroys the throne.

In this paper, the dataset is based on the archive of public games by OpenDota.com (formerly YASP). The dataset was made on the basis of discharge YASP 3.5 Million Data Dump replays games Dota 2 site yasp.co. During unloading thank Albert Cui and Howard Chung and Nicholas Hanson - Holtry. The license for the discharge of: CC BY-SA 4.0.

IV. Algorithm

Basically, We want to solve two problems: 1. Which model can give highest prediction precision? 2. As a player, how to win the game?

In order to solve the first problem, we need to take as much information as possible into account. One of the most important information from the records is Hero Selection. However, as the Hero information is neither numerical nor categorical, we cannot just put it into the prediction model directly. Thus we want to use a new method to deal with the Hero information. The new tool is TensorFlow, a new neural net library based on Python.

And for the second problem, we want to focus on Hero Selection, because in the previous part, we have known that Gold and Experience are crucial parts of winning the game. This is intuitive. But if proper combinations of heroes are selected, the change of winning can still increase.

We assume that winners have similarities in hero selection for two reasons. First, for the experienced players, usually they will not randomly select one hero from a pool of 133 heroes. It is more reasonable that he or she will select one specific hero in the game. Second, it’s time-consuming to know every aspect of a hero. If a player wants to use lots of different heroes in the games, he or she will need much more time in practicing, and may not be as good as others who only use one hero, in making use of the hero.

Thus, we will use item-based recommendation algorithm and recommendation system to discover the similarities of hero selection among all winner. After the system is built, we can have some recommended heroes.

A. *Tool :Tensorflow Library*

TensorFlow is Google’s second generation machine learning system, which is released as open sourced software on November 9,2015. We used it as a neural networks library in Python.

In order to feed the model, we need to use an input which can extract as much information from the hero selection matrix. We found that a binary table of the characters chosen by each team satisfies our requirement. This method is first used by Conley, K., & Perry, D. [8] with an aim to recommend heroes to players.

1) *Binary table construction process*

There are 113 heroes in the Dota 2, and they are recorded as ID numbers ranging from 1 to 113. So we use the following functions to construct input table X and output vector Y.

$$x_i = \begin{cases} 0, if\ a\ radiant\ player\ choose\ a\ hero\ with\ ID\ i \\ 1, \qquad otherwise \end{cases}$$

$$x_{113+i} = \begin{cases} 0, if\ a\ dire\ player\ choose\ a\ hero\ with\ ID\ i \\ 1, \qquad otherwise \end{cases}$$

$$y = \begin{cases} 0, if\ the\ radiant\ team\ won \\ 1, \qquad otherwise \end{cases}$$

2) *TensorFlow structure*

We use a network from Mark Dunne as a base model [10] and then we propose a more complex one, which will be compared with the base model using accuracy rate.

Figure 1 shows the architecture of network of base model.

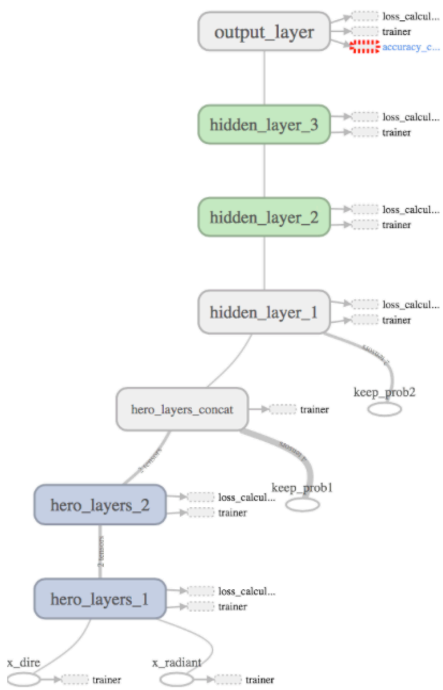

Figure 1. Architecture of base model

At the bottom of Figure 1, the selection information from Radiant side and Dire side are read into the network. Then Hero layer 1 and hero layer 2 will train these data, before they are concatenated by the concatenation layer. And then three hidden layers will reduce the dimension of the data, and finally the output layer will give the decisions of which side wins.

Figure 2 shows a more complex model. More information are added into the model, so we add two new nodes at the bottom of the figure. These two nodes absorb the information from the variables other than hero selection, such as Gold and Experience. Then they are processed by the characteristics layer, before they are combined by the concatenation layer. And then the hidden layers will train the data. Finally, the output layer gives out prediction results.

It is expected that the new model may perform better than the base model, because new model is fed with more information.

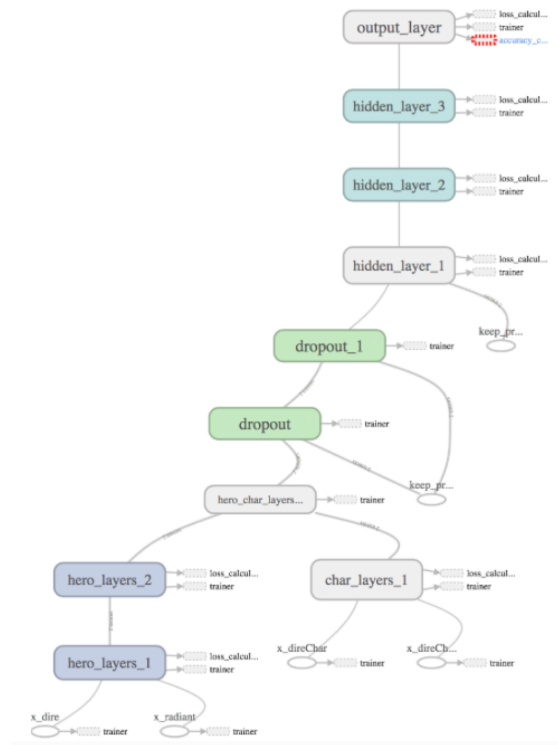

Figure 2. Architecture of new model

B. Algorithm : Item-based recommendation system

We use item-based recommendation algorithm to help players select heroes. Before we describe the algorithm we need to show how to construct the item table.

1) Item Table

We need to find out the similarities of hero selection among the winners. So we can see the winners as ”customers” and the heroes as “products”. If a winner choose hero with ID No. 1, then it’s like the winner “buys” the hero. So the rating of this hero will change from 0 to 1.

For example, in the first match, the Radiant side wins the game with hero_1 to hero_5. Then this means customer 1 buys item hero_ to hero_5. Customer 1 will give rating of 1 to hero_1 to hero_5, and rating 0 to other heroes. The same thing happens to the winning side of the rest of matches. In this way, we can build a n Item table.

2) Recommendation algorithm

Linden, G., Smith, B., & York, J. [11] and Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. [12] show many useful item-based recommendation system algorithms. One of the algorithms is as follows:

Item-based Algorithm

For every item I that u has no preference for yet
For every item j that u has a preference for
Compute a similarity s between i and j
Add u's preference for j, weighted by s, to a running average
Return the top items, ranked by weighted average

To implement recommendation algorithm, we use Mahout's built-in package to deal with data.

V. SOFTWARE PACKAGE DESCRIPTION

After running TensorFlow, we get an output file that contains the records of change of accuracy rate during iteration periods. It also included data of the architecture of neural networks. We can use TensorBoard to look at them.

The following are some screen shots of TensorBoard. We just need to click on the tabs to see each section.

In Figure 3 and Figure 4, we can trace the training process by looking at real time data in the black box.

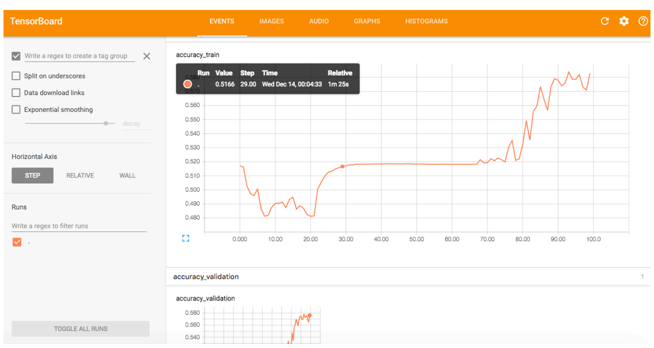

Figure 3. Screen shot of TensorBoard (1)

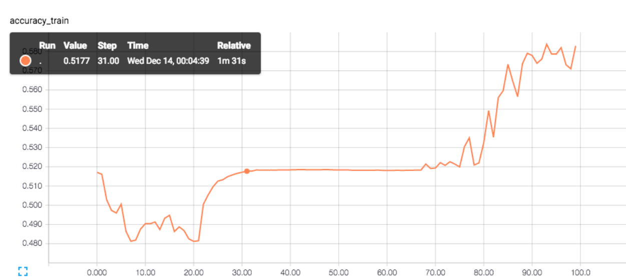

Figure 4. Screen shot of TensorBoard (2)

We can see the whole structure of neural network (Figure 5).

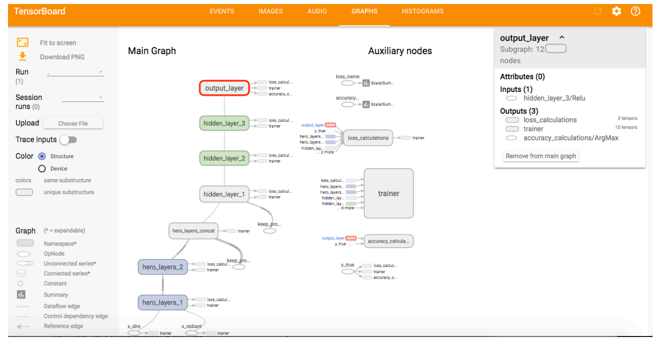

Figure 5. Screen shot of TensorBoard (3)

By clicking each part, we can go deeper into each node to learn more about how it processes the input data (Figure 6, Figure 7 and Figure 8).

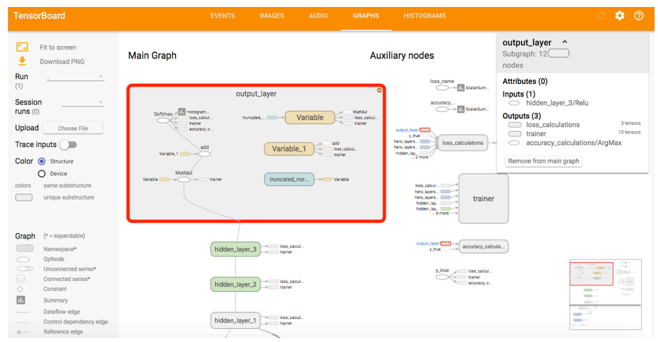

Figure 6. Screen shot of TensorBoard (4)

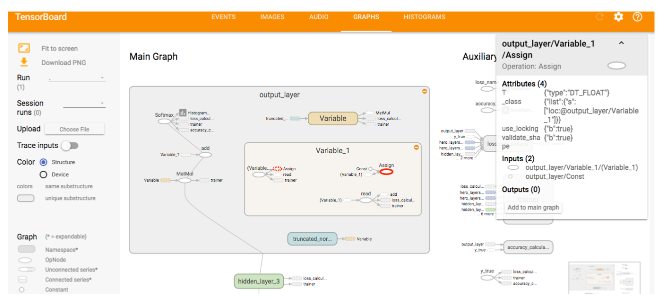

Figure 7. Screen shot of TensorBoard (5)

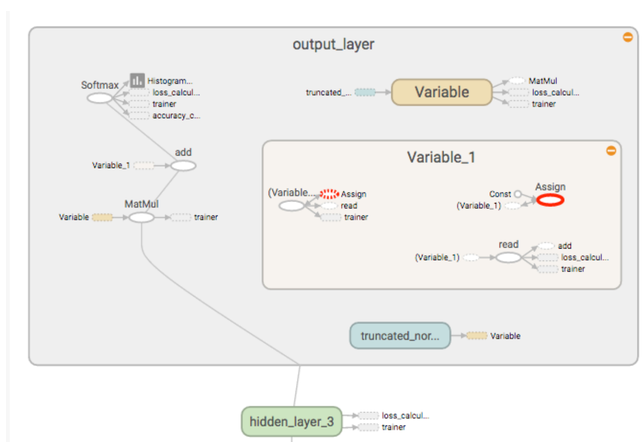

Figure 8. Screen shot of TensorBoard (6)

VI. Experiment Results

A. *Prediction model results*

We use Logistics regression model, multinomial Naïve Bayes model and neural networks to predict win probability. We found that logistics regression has highest accuracy rate, 68%. Multinomial Naïve Bayes model and neural networks have accuracy rates of 65% and 58%, respectively.

1) *Neural netwokd results*

Figure 9 shows the accuracy curves for 100 epochs in TensorFlow for the base model. The accuracy rate is 58%. The new model has lower accuracy rate of 52%, which implies that the newly added information may interference with the hero selection information, or the new model is not appropriate enough.

The low accuracy rate may imply that it's hard to give accurate prediction with only hero information. So we need to include more factors in the prediction model.

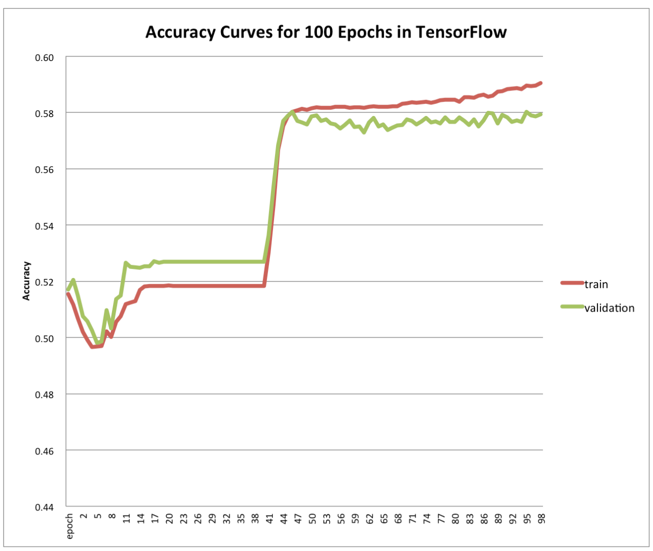

Figure 9. Accuracy curves for base model in TensorFlow

2) *Other models results*

By using hero selection binary matrix and other available variables, we get accuracy rate of 68%. Figure 10 is the plot of ROC curve from logistic regression model. Compared to guessing, which means 50% of accuracy, logistic regression model performs much better.

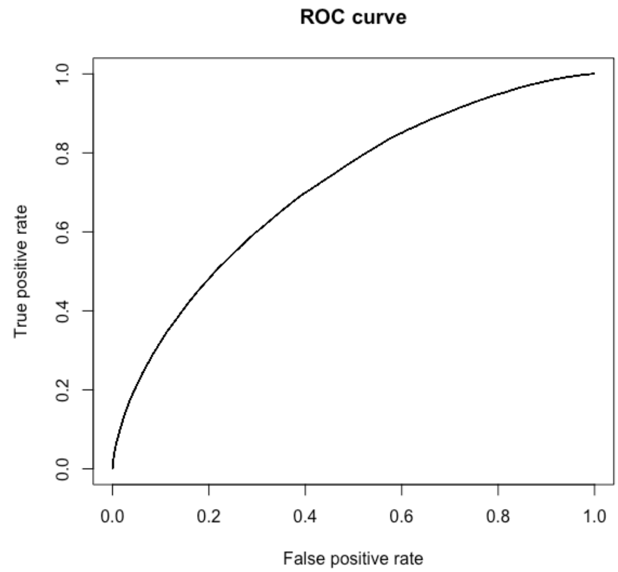

Figure 10. ROC curve for Logistic regression

We also use Multinomial Naïve Bayes model to predict winning side, and we get a lower accuracy rate of 65%.

B. *Recommendation system results*

After putting the item matrix into Mahout's recommendation engine, we get 5 recommended heroes for each match. We want to know who are the most popular heroes among the winners, so we draw a histogram of the recommendation results (Figure 11).

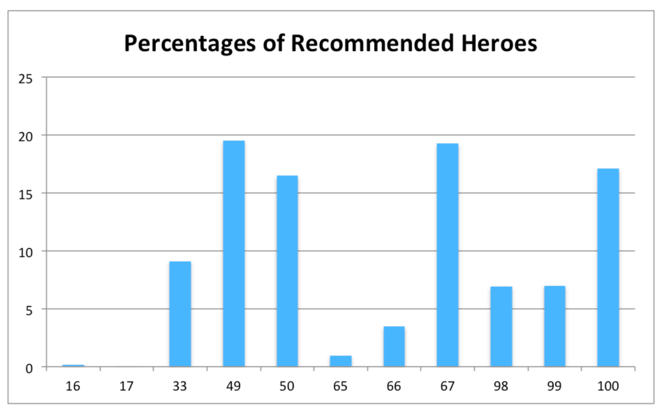

Figure 11. Histogram of recommendation results

From Figure 11, we find that heroes with ID 49, 67, 50, 100, 33, are the top five most popular ones. This means these heroes may be the favorite ones among all 113 heroes.

VII. Conclusion

A. Current Contribution

It is difficult to extract hero selection information from the match records, because they are recorded as IDs in a poll from 1 to 113. In this paper, we make use of the hero information by transforming them into binary matrix and then use neural networks to learn it. Although the result is not so satisfying, we find that with only hero information, we cannot get accurate prediction of the winning side.

Besides, recommendation algorithm is used to deal with hero information. And we find that some heroes are quite popular among winners of Dota 2.

We use other models to predict win probability, and we find that Logistic regression is the most useful one with accuracy rate of 68%.

B. Future Work

To improve the performance of neural networks, we will adjust the structure of neural nets. We may also change the weights and bias of the learning function.

For the recommendation system, we will use more appropriate similarity methods to calculate similarities between objects, such as categorical similarity measures.

Acknowledgment

We would like to thank Professor Ching-Yung Lin for his excellent lectures. We would like to also thank TA Eric F. Johnson for providing setup guides of Mahout and Hadoop.

References

[1] NewZoo, “the us esports audience 2016 ”, insights, [Online]. Available: https://newzoo.com/insights/infographics/the-us-esports-audience-2016/

[2] Skill-based differences in spatio-temporal team behaviour in defence of the Ancients 2 (DotA 2), A Drachen, M Yancey, J Maguire, D Chu, IY Wang- Games media entertainment (GEM), 2014 IEEE, 2014

[3] Thursten, Chris. "The International 2016 ends with a spectacular grand final". PC Gamer. Retrieved 15 August 2016.

[4] J. Miller and J. Crowcroft, “Group Movement in World of Warcraft Battlegrounds,” in Int. J. of Advanced Media and Communication, vol. 4, no. 4, pp. 387–404, 2010.

[5] F. Rioult, J.-P. Metivier, B. Helleu, N. Scelles and C. Durand. “Mining Tracks of Competitive Video Games,” in Proc. AASRI Conference on Sports Engineering and Computer Science, 2014.

[6] P. Yang, B. Harrison and D. L. Roberts. “Identifying Patterns in Combat that are Predictive of Success in MOBA Games,” in Proc. Foundations of Digital Games, 2014.

[7] Semenov, A., Romov, P., Neklyudov, K., Yashkov, D. and Kireev, D., Applications of Machine Learning in Dota 2: Literature Review and Practical Knowledge Sharing.

[8] Conley, Kevin, and Daniel Perry. "How does he saw me? A Recommendation Engine for Picking Heroes in Dota 2." *Np, nd Web* 7 (2013).

[9] Dunne, Mark. "Tensorflow Dota Predictor¶." Tensorflow Dota Model. Mark Dunne, 07 Feb. 2016. Web. 22 Dec. 2016. <http://markdunne.github.io/2016/02/07/TensorFlow-Dota-Model/>.

[10] Sarwar, Badrul, et al. "Item-based collaborative filtering recommendation algorithms." Proceedings of the 10th international conference on World Wide Web. ACM, 2001.

[11] Linden, Greg, Brent Smith, and Jeremy York. "Amazon. com recommendations: Item-to-item collaborative filtering." IEEE Internet computing 7.1 (2003): 76-80.