# Predicting outcomes of professional DotA 2 matches

Petra Grutzik Joe Higgins Long Tran

December 16, 2017

## Abstract

We create a model to predict the outcomes of professional DotA 2 (*Defense of the Ancients 2*) matches in this project. DotA 2 is an esport where 2 teams of 5 human players each play against each other. By looking at the teams playing as well as each player's in-game characters and historical performance, we should be able to predict the outcomes of matches better than chance. The result is that our best model's performance is on par with human-level performance of this prediction task.

## 1 Introduction

*Defense of the Ancients 2* is a competitive esport played on the computer. Its popularity is large and still growing quickly: the largest tournaments boast prizes larger than the PGA Tour and Wimbledon, and viewership is on par with the MLB World Series and NBA Finals.

We provide a brief overview of DotA 2 and its gameplay for context and understanding of our data and feature sets. A DotA 2 match is played by 2 teams of 5 human players each (similar to basketball). Each team can play on one of two sides ("Radiant" or "Dire") and their objective in the game is to destroy a building in the opposing team's base - the "ancient." Each match takes place over two distinct phases: the draft phase and gameplay phase.

### 1.1 The Draft

During a game, each human player controls 1 in-game character, or "hero". At the time of writing this paper, there are 114 heroes each with unique abilities and characteristics (more heroes are added to the game in regular updates). For example, some heroes are good at destroying buildings quickly, while other heroes are good at scouting the map and gaining valuable information for the rest of their team. As a result, some of the heroes synergize with each other or counter common strategies. The hero each player controls is determined during the draft.

In the draft, each team's captain will take turns banning and picking heroes from the pool of 114. Banning a hero removes it from the pool so neither team can have a human control it, and picking it selects a hero for one of their human players to control. As each pick or ban is made, team captains will react and adjust their upcoming draft choices to best suit their own strategy while simultaneously trying to counter the strategy their opponents appear to be crafting. In total, the draft process takes about 10 minutes, the result of which is both teams now have their 5 hero line ups with which to execute their strategy of destroying the opposing ancient.

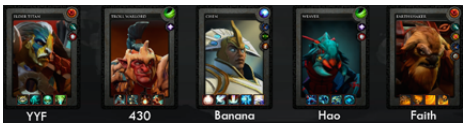


Figure 1: Example Radiant team


Figure 2: Example Dire team

### 1.2 Gameplay

After the draft, players control their heroes over a map. All heroes begin the game weak and gain power throughout the 30-45 minutes of the gameplay phase. Typically, the team that increases their heroes' power faster than the other team overpowers their opponent and is able to destroy their ancient. There are two ways for a hero to increase its power: accumulating gold to purchase items, and accumulating experience to gain access to powerful spells and abilities.

Thus, it is critical in predicting match outcomes to engineer features on a game-by-game basis that represent which teams are playing, the draft chosen, and how good human players have historically been in accumulating gold and experience during gameplay.


Figure 3: Map of DotA 2 gameplay field, locations of ancients marked by large red and blue stars

## 2 Previous work

Previous attempts of predicting the outcome of two team competitions have been fairly successful. From the perspective of sports analytics, it is easy to see a parallel: Sabermetrics and rigorous statistical analysis first transformed baseball management during the 1930s-1940s, which then led to a revolution in statistical applications in other sports and opened new fields of research in statistics itself. Progress has been made on predicting the outcomes of two player games (Lin). Games with multiple teammates on each team prove more difficult to predict. The intricacies of teammate interaction and individual player performance complicate the problem dramatically. The most similar sport to DotA 2 is basketball. Previous researchers have attempted to predict basketball matches using neural nets (Torres, Lin). We refer to these models when experimenting with configurations of our neural net. A CS 229 team previously attempted to predict DotA 2 matches using logistic regression (Song).

## 3 Human performance on the task

An important diagnostics test for many machine learning applications is comparing algorithmic performance with human performance on the task of interest. This diagnostic is especially useful for tasks on which humans perform well, such as facial and speech recognition. For DotA 2 prediction, we use the odds found on esports betting sites as a baseline. These odds represent the collective opinion of many bettors on match outcomes and are likely to be at least as accurate as, if not more than, the prediction of most individual bettors, provided that the betting markets are large enough. Moreover, a comparison with betting sites is extremely relevant, because it is directly related with a potential use case of our models, which is to inform bettors decisions.

We gathered betting data from the site Gosugamers.net, which provides betting services for DotA 2 games and records betting data of 16,800+ matches from May 2013 to November 2017. Here a match refers to a set of DotA 2 games between two teams (equivalent to match and set in tennis), and the format of a match can vary. Some examples of possible match formats are best-of-three (in which the first team that wins two games wins the overall series), best-of-one, best-of-two, etc. We note two things: first, bettors only have information before a match (i.e., they do not have information on hero selection, in-game actions, or results of any game within that match). Second, while it is impossible for two teams to draw in a game, it is possible to have a match draw (e.g., in a best-of-two match).

To estimate human performance on Gosugamers.net, for each match we selected the outcome (either team 1 wins, team 2 wins, or two teams draw) with the largest betting amount as the collective prediction of bettors. We then compared this prediction to the actual results, and found that the bettors collective prediction has a 62.8% accuracy rate. This result makes sense, as it is better than random prediction but is not very close to perfect accuracy, reflecting the competitive nature of DotA 2 games.

## 4 Dataset and Feature sets

### 4.1 Dataset

Our data is all professional DotA 2 games played from November 2011 to October 2017, or 47,440 games. Professional matches are defined as matches played by professional teams in tournaments that are officially sanctioned by Valve Corporation, the software company responsible for DotA 2's development. We partitioned our data into three groups that we will call train, dev and test. Since our model will be most useful if it can predict games yet-to-be-played in the future, we partitioned our test set to be the 4,862 games played after May 1, 2017. Of the remaining 42,578 games played before May

1, 2017, the train set is a randomly selected 90% of those games and the dev set is the other 10%.

## 4.2 Feature sets

Critical to our model's success is the engineering of features that contain structure with predictive power of winning a game. We constructed 3 features sets to test each of our hypotheses of what data would provide the most predictive structure.

### 4.2.1 Team indicator variables

Which teams are playing against each other should matter. Some teams are simply good teams that will win often, and some teams are not good teams that will not win often. A team may even be another team's "kryptonite", and have a well-developed strategy that consistently counters another team. To enable our models to understand the relationship between teams and winning, we created 8587 binary indicator variables that take on the values of 0 or 1.

$$I_{\text{team}} = \begin{cases} 1, & \text{if team is playing in the match} \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

The sum of all these indicators for any given match will be 2, since only 2 teams can play in a given match.

### 4.2.2 Draft indicator variables

Which heroes are playing in a given match should matter. Which heroes are being played together as a lineup of 5 should matter, and which lineup of 5 opposing heroes they are going against should matter. To give our models the proper flexibility to understand the relationships between the 5 hero lineups that are being controlled by each team, we created 228 binary indicator variables, one for each hero and each team that hero can be played on ($114 \times 2$):

$$I_{\text{hero, team}} = \begin{cases} 1, & \text{if team is controlling hero this game} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

The sum of all these indicators for any given match will be 10, since both teams control 5 heroes.

### 4.2.3 Player historical performance

The human players in a given match should matter. Some players are so mechanically skilled that their team will play strategies based entirely on getting that player's hero to be so powerful he can "carry" the rest of the team to victory (often called the "hard carry").[1] For this reason, we created features that are rolling averages of box-score metrics that are highly predictive of victory in a given game. Since we can't use metrics from the game we are trying to predict the outcome of, we must create features as rolling averages of these metrics from prior games.

Our final iteration of these metrics use a rolling average of the 10 immediately preceding games in order to strike a good balance between having a robust view of that player's historical performance and allowing it to move up or down as the player adjusts to new strategies or updates to the game. The metrics that we tracked for each human player are:

1. **Gold per minute:** Gold allows heroes to purchase powerful items that help them overpower other heroes. In nearly every game, the team with more total gold net worth wins.

2. **Experience per minute:** Gaining experience unlocks powerful spells and abilities for a hero throughout the game. In nearly every game, the team with more total experience wins.

3. **Kills per minute:** Killing enemy heroes not only gains your team gold, but also subtracts gold from the enemies and removes them from play for a period of time. Teams with more kills typically win.

4. **Lane efficiency:** This measures the percent of gold collected by a hero that's available to be collected in the first 10 minutes of the game, known as the "laning phase." Good performance during the laning phase often sets a team up for victory.

5. **Solo competitive rank:** This is a number that tracks the player's individual ranking. It fluctuates up and down as the player wins and loses, and is similar to an Elo rating system[2]. This number is created and maintained by the developer of DotA 2, Valve.

### 4.2.4 Bringing it all together

Since our player historical performance feature set consists of continuous variables on a different scale than each other and our binary indicators, when we run experiments with total feature sets that incorporate player historical performance we standardize input fields using

[1] https://dota2.gamepedia.com/Role#Hard_Carry
[2] https://en.wikipedia.org/wiki/Elo_rating_system

$z$-score:

$$z_{ij} = \frac{x_{ij} - \mu_i}{\sigma_i} \qquad \forall i, j \tag{3}$$

$i$: index of the set of features
$j$: index of the set of matches
$x_{ij}$: the value of feature $i$ in match $j$
$\mu_i$: mean of feature $i$
$\sigma_i$: standard deviation of feature $i$

# 5 Methods

We implemented two models over a variety of combinations of our feature sets. Our two models are a Support Vector Machine (SVM) and a Neural Network. We chose to use a SVM because it has been shown to be effective at classification in high dimensional feature spaces (which ours is) and because of the power of experimenting in even higher dimensional spaces with kernels. We also experimented with neural net for this task because of existing literature where neural networks have had success for predicting basketball outcomes[3].

## 5.1 Support Vector Machine approach

We experimented with different kernels and values of the regularization term. In the end, our most successful Support Vector Machine used a linear kernel with a penalty on the error term of 2 (success defined as maximizing dev set accuracy). In the following 'Experiments' section of the paper, it is this configuration that we refer to as the SVM.

## 5.2 Neural Network approach

We experimented with different configurations of neural nets to find the best performing one on the dev set. For our task, the most successful neural net used three hidden layers: 200 nodes in the first two layers with ReLU $f(x) = \max(0, x)$ as the activation function and one output node with a sigmoid activation function $f(x) = 1/(1 + e^{-t})$ as the final hidden layer. Again, we define success as maximizing dev set accuracy. ReLU has been commonly used as an activation function because it performed better than sigmoidal activation functions in deep NNs (Glorot, Bordes, and Bengio, 2011), and helped to obtain best results on several benchmark problems across multiple domains (e.g., Dahl, Sainath, and Hinton, 2013; Krizhevsky et al., 2012).

[3] https://homepages.cae.wisc.edu/ ece539/fall13/project/AmorimTorres_rpt.pdf

We used the sigmoid activation function in the final layer to classify the match from the perspective of one team as either a win or a loss.

# 6 Experiments

## 6.1 Support Vector Machine

| Feature Set | Train Acc. | Dev Acc. |
|---|---|---|
| Draft Features Only | 58% | 57% |
| Team Features Only | 72% | 59% |
| Team and Draft Features | 72% | 61% |
| Recent Player Stats and Draft Features | 55% | 55% |

Figure 4: SVM Accuracies

First we will discuss the model output where our feature set did not include player statistics (i.e., only Team and Draft indicator variables). We notice that draft features only were less predictive than team features only, which in turn were less predictive than both combined. This leads us to believe that the humans playing the game have a stronger effect on the outcome of a game than the heroes they control.

Additionally we notice that training accuracy gets much higher when we introduce team features, but the dev accuracy only goes up a little. This leads us to believe that our SVM is overfitting when it has team features because there are a large number of features (8587 from team indicators) relative to how many training examples we have.

Next, when we include recent player statistics, accuracy goes down. Our hypothesis here is that these features actually do not provide any additional predictive power to our model. See figure 5 of XP/min from trailing 10 games versus XP/min in current game. While XP/min has a lot of predictive power for victory in a current game, past XP/min does not predict XP/min in a current game very well for professional players. This lack of correlation holds for other rolling average windows, not only 10.

## 6.2 Neural Network

We noticed that the train accuracy is much greater than our accuracy on the development set. We suspect this is caused by over fitting. To compensate for over fitting we conducted multiple experiments using regularization,

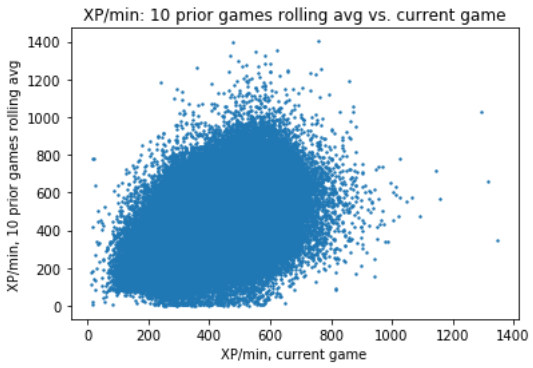


Figure 5: Rolling XP/min vs. current XP/min

| Feature Set | Train Acc. | Dev Acc. |
|---|---|---|
| Draft Features Only | 68% | 54% |
| Team Features Only | 67% | 58% |
| Team and Draft Features | 69% | 60% |
| Recent Player Stats and Draft Features | 53% | 53% |

Figure 6: Neural Net Accuracies

early-stopping, and dropout. The results of our experiments are listed in Figure 7.

We found dropout and L2 regularization did correct for the over fitting on the train set. However, this also decreased accuracy the development set.

## 7 Test Performance

We tested the SVM and neural net with our reserved test data using the Team and Draft feature set (figure 8). We used the best performing models on our dev set to run on our test set. Our hypothesis was that test accuracy would decline over time as strategies evolved over time and game updates change the dynamics. However, there does not appear to be a trend in accuracy over time.

## 8 Conclusion/Future improvements

We are encouraged that these early results approach human performance on DotA 2 match prediction, and are optimistic that further tweaks can cross that threshold. To recap, our SVM with linear kernel had the best dev and test accuracy, but we hope to improve upon our methods in the future. A few ideas for next steps include:

- Reduce the dimensionality of our team indicator variables, making our models less prone to over fitting

| Team and Draft Features | Train Acc. | Dev Acc. |
|---|---|---|
| Base Model | 68.6% | 59.6% |
| L2 Regularization (factor = 0.001 | 68.5% | 59.7% |
| Early Stopping | 68.6% | 59.6% |
| Early Stopping and L2 Regularization (factor = 0.001) | 68.3% | 59.8% |
| Early Stopping and L2 Regularization (factor = 0.001), 20% hidden layer dropout | 68.6% | 58.8% |
| Early Stopping and L2 Regularization (factor = 0.001), 25% hidden layer dropout | 68.6% | 58.7% |

Figure 7: Regularization and Dropout Experiment Accuracies

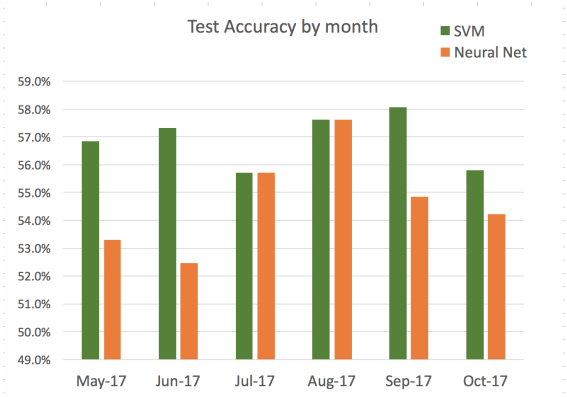


Figure 8: Test Accuracy of SVM (Green) and Neural Net (Orange)

ting

- Experiment with different box score metrics and different rolling average window lengths
- Weigh the influence on a prediction more by games close together in time
- Expand data set by creating two records for each game, the first record considering one team the allied team and the second record considering the other team the allied team (similar to image recognition algorithms that flip images to expand their data sets)

# 9 Contributions

**Joe Higgins:**

- Wrote module for programmatically collecting data from www.opendota.com web API
- Wrote feature engineering for all team, draft, and player metrics
- Wrote module for creating model-ready input data
- Lead for running SVM experiments
- Co-lead for final paper

**Petra Grutzik:**

- Lead for running Neural Network experiments
- Lead for poster
- Co-lead for final paper

**Long Tran:**

- Determined metric for human performance baseline
- Collected and analyzed data for human performance baseline
- Neural Network regularization

# 10 Bibliography

G.E. Dahl, T.N. Sainath, G.E. Hinton Improving deep neural networks for LVCSR using rectified linear units and dropout IEEE International conference on acoustics, speech and signal processing, IEEE (2013), pp. 8609-8613

Glorot, X., Bordes, A., and Bengio, Y. (2011). Deep sparse rectifier networks. In AISTATS, vol. 15 (pp. 315323).

Lin, Jasper, Logan Short, and Vishnu Sundaresan. "Predicting National Basketball Association Winners." (2014): n. pag. Web.

Song, Kuangyan, Tianyi Zhang, Chao Ma "Predicting the winning side of DotA2". CS229 2015.

Torres, Renato Amorim. "Prediction of NBA games based on Machine Learning Methods". University of Wisconsin Madison. December, 2013.