Kyle Pomerleau

8/12/2015

Using Machine Learning to Predict the Outcome of Dota 2 Matches

Introduction

Dota 2, or "Defense of the Ancients" 2 is a Multiplayer Online Battle Arena (MOBA) PC game. Although Dota 2 has amassed an impressive following throughout the world, the game is very difficult to learn and to play. The object of my project is to use Machine Learning techniques to help players learn one important aspect of the game: hero selection. My question of interest, specifically is: Is it possible to predict the outcome of a Dota 2 match using Machine Learning?

Using a logistic regression trained on over 19,000 games, I was able to predict the outcome of a match with 61.7 percent accuracy based on hero selection alone, with an AUC of .68. Controlling for the performance of each player in a game, I was able to predict the outcome of a match with a 98 percent accuracy. My results imply that 20 percent of the outcome of a match is determined by hero selection and the remaining 80 percent is a result of player performance.

Part 1: What is Dota 2?

Dota 2, or "Defense of the Ancients" 2 is a Multiplayer Online Battle Arena (MOBA) PC game. It is the sequel to the popular Warcraft 3 mod Dota which was developed back in 2003. Dota 2 was developed by Value Software and was released on July 9th 2013. Since its release it has become very popular throughout the world with large followings in the North and South America, Europe, and South East Asia. Currently about 560,000 people play Dota 2 every day.

Dota 2 also has a significant profession scene throughout the world. For the last 5 years, Valve has hosted what is called "The International" tournament. In this annual tournament the best teams in the world compete for a large cash prize pool. The 2015 International began on Sunday August 25th. The 2015 prize pool is $17.5 million, the largest it has ever been. Valve produced and released a documentary in 2014 about professional Dota 2 players and its International tournament.

Game Play

As stated previously, Dota 2 is a Multiplayer Online Battle Arena (MOBA). Each game is played with ten players, five on each team. One team is called the "Radiant" and the other the "Dire." The teams are based in each corner of the map [Figure 1]. The Radiant is based in the bottom left and the Dire in the Top right. The object of the game is to destroy the other team's "Ancient." The ancients are located in the bases of each team on the top right and bottom left.

Figure 1.

Drafting

At the beginning of each match, players can choose from 1 of 110 "heroes." When one hero is chosen, no other player can select that hero [Figure 2]. Each team needs to balance itself based on the abilities of each hero. As is the case with soccer or football, there are different roles or positions in a team that players need to fill. Certain heroes are better for certain roles than other heroes.

Furthermore, certain heroes work well with other heroes. These are called synergies. One hero's abilities may work well with another. Conversely, many heroes can counter others. One hero's abilities may be able to neutralize others' abilities.
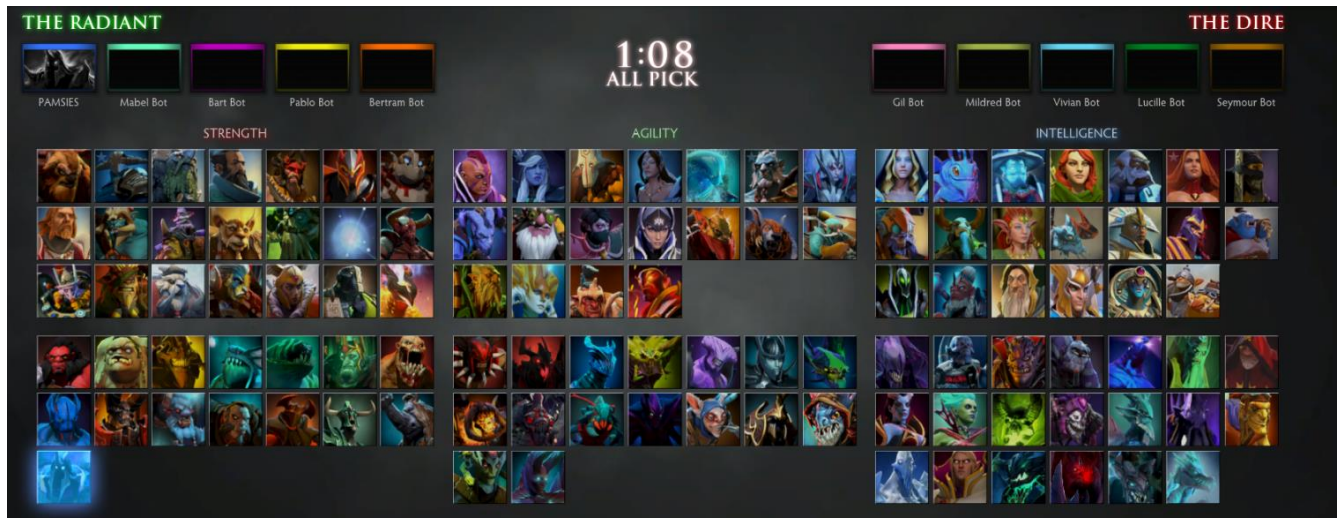

Figure 2.

Throughout the game each hero becomes more powerful by collecting gold to purchase items and gaining experience to level their abilities. You gain both experience and gold by killing enemy heroes or non-player units called "creeps" [Figure 3]. The figure below shows Dota 2's Heads-up display (HUD). It indicates the hero's level (bottom left), the hero's stats, its abilities (middle), and items (right). This displays for the entirety of the game.

Figure 3.

The object of the game is to work with your four other teammates to destroy the other team's "Ancient."

Part 2: What is my Project Question?

Although Dota 2 has amassed an impressive following throughout the world, the game is very difficult to learn and to play. At its release, the game was given very good reviews. However, the one downside is

that the game has a very steep learning curve. And this frustrating aspect of the game is only compounded by an unforgiving player base. Players are basically stuck between learning the game without playing it, or learning while playing, but risking a bad experience.

The object of my project is to use machine learning techniques to help players learn one important aspect of the game: hero selection. As stated above, players can choose heroes at the beginning of a match and it is necessary to choose heroes based on what your teammates have chosen or what your opponents have chosen. Machine learning may be able to indicate what hero is the best to choose given what your teammates and opponents have chosen.

My project question is: *is it possible to predict the outcome of a Dota 2 match based on hero selection using data on the outcome of past Dota 2 matches?*

Part 3: Data Collection

Dota 2 is hosted on Steam, which is Valve Software's game platform. The Steam API contains a significant amount of data on many of its games. For the purposes of this project, I will being using methods pertaining to Dota 2.

For Dota 2, the API contains information on every single public match that has been played (private matches are not accessible). Each match in the API is identified by both a unique "match id" and a unique sequence number. You can access Dota 2 match details through three different API calls [Table 1].

Table 1. Steam API Calls to Get Dota 2 Match Details	
GetMatchDetails	Information about a particular match.
GetMatchHistory	A list of match ids, filterable by various parameters.
GetMatchHistoryBySequenceNum	A list of matches and their details by their sequence number.

Available Match Details

Each API call by either match id or sequence number returns a dictionary filled with extensive match details. The dictionary includes the length and outcome (who won) of the match, the players (by account id), heroes chosen, the item each player chose for their hero, when they chose those items, and what order each hero ability was selected.

API limitations

Currently there are a few API limitations that make data collection less than ideal for the purposes of this project. The most ideal way to collect the data would be to use "GetMatchHistory." Specifically, this API call returns a list of matches that can be filtered in a number of ways. Most importantly, matches can be filtered by skill level. This is useful in that I would only be able to see the matches in which players are playing and using heroes optimally. As of 2014, Valve has limited the number of matches you can call by this method to 500, which is far less than necessary for this project.

As stated previously, I used GetMatchHistorybySequenceNumber. This API call does not have the 500 match limit. You could conceivably get data on every single public Dota 2 match played in the order they were played. However, this API call does not have a filter method. Your API gets all matches, regardless of skill level. And as indicated above, there is not indicator in the match data itself for match skill level, so post filtering is not possible either.

Part 4: Data Overview, Descriptive Statistics

I collected one data set for this project. Each data set contains 25717 games. Each data set is set up with 221 features: one feature for each hero of the 110 heroes on the Radiant or Dire team and a final feature that indicates with team 1 (Radiant) won. For example, a given game has ten heroes: 1 through 10. Observation number one would indicate 1 for each of the 5 heroes on one team and 1 for each of the 5 heroes on the opposing team. Then if team 1 wins, a 1 is indicated. Another way to think of this is that each row in a data set has a sum of either 10 or 11, depending on whether team 1 wins or loses.

Table 2. Dataset used for Analysis
221 features, dummy variables indicating hero selection on each team. One variables indicates whether the Radiant team won the match.

Descriptive Statistics

Of the 25717 games, the Radiant won 13,727 games or 53.3 percent of all the games in the sample. The Dire team won 11,990, or 46.7 percent of the games in the sample. This slight imbalance is expected given that the Radiant team is known to have a slight advantage.

Table 3. Wins by Team in Sample		
Team	Wins	Win %
Radiant	13727	53.30%
Dire	11990	46.70%
Total	25717	100%

All 110 of the Heroes are represented in the sample of 25,717 games. The most popular hero played in the games sample was "Bloodseeker" (hero id 4). He appeared in 8,456 games or 32 percent of all games on either the Radiant or Dire side. Chen (hero id 66) appeared in only 220 games in the 25666 sample. On average, each hero appeared 2333 times. All 110 heroes also appeared on both the Radiant and Dire teams with relatively good balance [Table 4, appendix].

Part 5: Models Used

Logistic Regression

The first model I used is a logit regression model that regresses heroes chosen (indicated by a 1 or a 0) on the radiant team and the dire team on the likelihood of victory for radiant.

Random Forest

The second model I used is a random forest model. A Random forest is a classification method that uses a series of random decision trees that uses a random sample of features for each tree. The advantage of a random forest model over a logistic regression is that it learns feature interactions automatically. For the purposes of this project, feature interactions are important. Certain heroes have synergies with others and, conversely, heroes may be countered by other heroes on the opposing team. This model will pick up those secondary effects. The downside is that this model takes a much longer time to run.

Results

In order to evaluate both models, I split the model into a training set of 19287 games and a testing set of 6430 games.[1] The first model evaluation metric I used was accuracy. Accuracy shows the percent of the games in the testing set the model was able to predict correctly (win or loss). The null model, a model that simply predicts the average outcome (Radiant always wins) has an accuracy of 52.3 percent. Adding features for each hero chosen increases the model's accuracy to 61 percent. The random forest model had a maximum accuracy of 60% at around 500 max estimators.

In addition to accuracy, I calculated each model's "Area under the curve" or AUC. This model evaluation technique is impacted less by sample balance than accuracy. The null model (guessing Radiant always wins) has an AUC of .5. The logistic regression model has an AUC of .64. The random forest model had an AUC of 0.58.

Model Evaluation		
	Accuracy	**AUC**
Model 1	61%	0.64
Model 2	60%	0.58
Null	52.30%	0.5

Conclusion

Using a logistic regression model I was able to predict the outcome of a Dota 2 public match with 61 percent accuracy based solely on hero selection. Using a random forest, which captured hero interactions, I was able to predict the outcome of a Dota 2 match with 60 percent accuracy. Both of these outcomes imply that about 20 percent of the outcome of a match is determined by hero selection.

Although this model improves upon random guessing, there are known weaknesses with the analysis that likely harm the overall accuracy of the model. First, the inability to filter matches by skill level likely introduces randomness into the model. Low-skilled players may play poorly regardless of team composition, which makes predicting their match based on hero selection problematic. Having a dataset that only uses high-skilled players would likely improve the result.

Second, the use of the Random Forest model captures many of the hero interactions that are a fundamental part of the game. However, the model actually performs slightly worse. One reason that this may be the case is that hero selection is skewed. Certain heroes are selected a lot, while some only

---

[1] In addition to splitting the data, I used cross validation with 40 folds. The results were identical, so I saved time by just using train, test, split.

appear in the dataset 200 times. The Random Forest model may have trouble when some hero combinations may not exist in the dataset.

Future work that may improve this model would be to drastically increase the sample size to attempt to capture more hero combinations. In addition, there may be more advanced ways to call the API in order to filter matches by skill level.

Appendix

Table 4. Table Hero Appearance, Total and by Team				
Hero ID	Hero Name	N Total	N Radiant	N Dire
1	Anti-Mage	3530	1779	1751
2	Axe	3581	1801	1780
3	Bane	616	305	311
4	Bloodseeker	8456	4239	4217
5	Crystal Maiden	3436	1736	1700
6	Drow Ranger	1899	971	928
7	Earthshaker	7632	3864	3768
8	Juggernaut	5847	2878	2969
9	Mirana	3618	1807	1811
10	Morphling	801	380	421
11	Shadow Fiend	5744	2905	2839
12	Phantom Lancer	3400	1659	1741
13	Puck	1142	575	567
14	Pudge	6659	3326	3333
15	Razor	977	481	496
16	Sand King	2155	1049	1106
17	Storm Spirit	6154	3081	3073
18	Sven	2315	1161	1154
19	Tiny	2367	1163	1204
20	Vengeful Spirit	1653	835	818
21	Windranger	7895	3932	3963
22	Zeus	4781	2405	2376
23	Kunkka	1495	723	772

25	Lina	5107	2517	2590
26	Lion	4121	2083	2038
27	Shadow Shaman	1615	812	803
28	Slardar	2451	1290	1161
29	Tidehunter	1973	1003	970
30	Witch Doctor	4010	2021	1989
31	Lich	1204	599	605
32	Riki	2685	1379	1306
33	Enigma	1258	660	598
34	Tinker	1868	984	884
35	Sniper	3840	1857	1983
36	Necrophos	3992	2028	1964
37	Warlock	1030	499	531
38	Beastmaster	494	265	229
39	Queen of Pain	4487	2309	2178
40	Venomancer	1019	497	522
41	Faceless Void	2947	1509	1438
42	Wraith King	3058	1530	1528
43	Death Prophet	968	500	468
44	Phantom Assassin	4682	2308	2374
45	Pugna	1277	630	647
46	Templar Assassin	2261	1109	1152
47	Viper	2065	1076	989
48	Luna	1168	565	603
49	Dragon Knight	1386	670	716
50	Dazzle	1770	855	915
51	Clockwerk	2451	1156	1295
52	Leshrac	4927	2394	2533
53	Nature's Prophet	1950	1002	948
54	Lifestealer	1770	880	890
55	Dark Seer	831	411	420

56	Clinkz	1538	779	759
57	Omniknight	2367	1162	1205
58	Enchantress	474	234	240
59	Huskar	2544	1275	1269
60	Night Stalker	881	430	451
61	Broodmother	593	288	305
62	Bounty Hunter	3991	1905	2086
63	Weaver	1626	843	783
64	Jakiro	991	500	491
65	Batrider	428	217	211
66	Chen	220	113	107
67	Spectre	2571	1319	1252
68	Ancient Apparition	1343	662	681
69	Doom	1343	675	668
70	Ursa	2260	1178	1082
71	Spirit Breaker	4613	2257	2356
72	Gyrocopter	2925	1433	1492
73	Alchemist	1205	594	611
74	Invoker	4542	2256	2286
75	Silencer	4209	2116	2093
76	Outworld Devourer	828	426	402
77	Lycan	553	276	277
78	Brewmaster	461	236	225
79	Shadow Demon	501	255	246
80	Lone Druid	496	235	261
81	Chaos Knight	1281	651	630
82	Meepo	710	364	346
83	Treant Protector	741	370	371
84	Ogre Magi	2115	1027	1088
85	Undying	3466	1736	1730
86	Rubick	2786	1417	1369

87	Disruptor	1531	748	783
88	Nyx Assassin	1548	766	782
89	Naga Siren	568	295	273
90	Keeper of the Light	1071	532	539
91	Io	427	204	223
92	Visage	275	136	139
93	Slark	4957	2472	2485
94	Medusa	1801	914	887
95	Troll Warlord	1286	634	652
96	Centaur Warrunner	1298	631	667
97	Magnus	2437	1257	1180
98	Timbersaw	1145	578	567
99	Bristleback	3073	1496	1577
100	Tusk	3732	1888	1844
101	Skywrath Mage	1512	771	741
102	Abaddon	1758	855	903
103	Elder Titan	411	192	219
104	Legion Commander	3522	1731	1791
105	Techies	2722	1408	1314
106	Ember Spirit	1685	839	846
107	Earth Spirit	1067	587	480
109	Terrorblade	455	229	226
110	Phoenix	1070	563	507
111	Oracle	297	139	158
112	Winter Wyvern	1589	786	803