Win Prediction in Multi-Player Esports: Live Professional Match Prediction

Victoria J. Hodge, Sam Devlin, Nick Sephton, Florian Block, Peter I. Cowling *Member, IEEE*, Anders Drachen

Abstract—**Esports are competitive videogames watched by audiences. Most esports generate detailed data for each match that are publicly available. Esports analytics research is focused on predicting match outcomes. Previous research has emphasised pre-match prediction and used data from amateur games, which are more easily available than professional level. However, the commercial value of win prediction exists at the professional level. Furthermore, predicting real-time data is unexplored, as is its potential for informing audiences. Here we present the first comprehensive case study on live win prediction in a professional esport. We provide a literature review for win prediction in a multi-player online battle arena (MOBA) esport. The paper evaluates the first professional-level prediction models for live DotA 2 matches, one of the most popular MOBA games and trials it at a major international esports tournament. Using standard machine learning models, feature engineering and optimization, our model is up to 85% accurate after 5 minutes of gameplay. Our analyses highlight the need for algorithm evaluation and optimization. Finally, we present implications for the esports/game analytics domains, describe commercial opportunities, practical challenges, and propose a set of evaluation criteria for research on esports win prediction.**

Index Terms—**computer games; data analytics; esports; prediction; real-time analytics**

I. INTRODUCTION

Esports is the term used to describe video games that are played competitively and watched by, normally large, audiences [1]. Esports is an important research field across academia and industry just in terms of size [1], [2], [3], [4], [5]. Goldman Sachs [6] predicted a compound annual growth rate of 22% with the market worth $1.1 billion by 2019 and Superdata [7] estimated there will be 330 million spectators by 2019. The availability of detailed data from virtually every match played coupled with this huge expansion has introduced the field of esports analytics [8], [5]. Esports analytics is defined by [5] as: "*the process of using esports related data, [...], to find meaningful patterns and trends in said data, and the communication of these patterns using visualization techniques to assist with decision-making processes*". This definition highlights a fundamental challenge in esports: making the matches comprehensible to the audience. Many esports are

complex and fast-paced, making it hard to fully unpack the live action with the naked eye. MOBAs also provide a fertile testing ground for machine learning due to the availability of high-dimensional, high-volume data.

Esports analytics has focused on the Multi-player Online Battle Arena (MOBA) genre, which is arguably the most common esports format. MOBA titles such as *League of Legends*, *DotA 2* and *Heroes of Newerth* attract hundreds of millions of players [5], [7]. Within esports, but notably in MOBAs, **win prediction** has formed the focal point of analytics research across industry and academia, even if that research is somewhat fragmented [3]. However, previous work has several limitations, including the fact that it is mainly focused on pre-match predictions, which informs betting, rather than models that can integrate live data streams, and seek to inform and engage the audience. There is also a lack of research at the professional level, despite differences in player behavior as a function of skill being documented [8]. Furthermore, no previous prediction models have been adapted for and tested in actual esports tournaments.

We discuss a range of win prediction techniques in section III-B. However, this previous work has limitations. In many ways this is due to esports analytics being an emergent field of inquiry. We detail these limitations in sections III-C and VIII which include: under-prioritizing data from professional players (as also noted by [3]), building models from data across the entire skill set which lowers the accuracy for professional match win prediction, only predicting historical data rather than real-time (live) prediction, and using data generated over long time periods across significant game updates and changes. Unlike traditional sports, in which the game rules are mostly stable, in esports major updates can significantly alter the core characteristics of the game mechanics. These major updates could render previous data obsolete.

The focus of this paper is to use live game state (e.g. positions of players, performance metrics etc.) to predict the likely winner for the popular MOBA game **DotA 2**[1]. This paper builds on and significantly expands a preliminary feasibility report [9] which demonstrated prediction on a small data set and established some data features to use from an initial set of possibilities.

The goals for the work presented here is threefold:

- Building and expanding on previous research, investigate the possibility for developing models that can provide live (runtime) match prediction for professional-level MOBA matches, with the aim of providing a basis for informing

[1]http://blog.dota2.com/

players and audiences. In many esports, unlike traditional sports, there is no 'score'. Furthermore, these games are highly complex and can be hard to follow for novice audiences. Therefore, simple statistics that summarize the current game state, such as a match win percentage, can broaden the appeal of the games and make them more accessible for viewers.

- Evaluate the impact of using non-professional data on professional match prediction,
- Implement and test a solution at a major esports tournament.

Importantly, the goal of the work presented here is not to provide an algorithmic contribution towards optimizing prediction models, or previous work, as there is no previous match prediction system for live tournament broadcasts to optimize. Rather, we adopt models towards addressing the live match prediction problem at the professional player level, and then test the solution in a major international esports tournament.

The contribution presented here can be summarized as follows:

- The first structured literature review and analysis of the state-of-the-art of win prediction in DotA 2.
- We present extended methods, results and analyses for win prediction in professional games using training data across extremely high skill public and professional-level games.
- Our evaluations thoroughly analyze the prediction algorithms used in the literature and their respective configurations to identify the best performing algorithm-configuration on various features of MOBA data.
- Our system can predict professional MOBA games, and produces reliable prediction results even with limited, mostly professional-level training data. No previous academic work has implemented a real-time prediction system and deployed it in real tournament settings. This is done here with a discussion of the practical implications and issues of live prediction systems in esports.

The remainder of the paper is organized as follows. In the next section we provide an overview of DotA 2 gameplay and in section III we analyze related work in the literature for DotA 2 win prediction, we focus on the data and algorithms used, and identify limitations in the current work. In section IV, we describe our dataset from mixed professional and high-skilled DotA 2 games. In section V, we present a training approach using the mixed data to train machine learning algorithms for prediction. We then evaluate the learned models on benchmark and professional data in section V. Section VI describes the design and implementation of a fully functional prototype for real-time win prediction in DotA 2 and evaluates a real-time deployment. Section VII provides discussion and detailed analysis of our evaluations from sections V and VI. Finally, in the Conclusion in section VIII, we summarize the results and their implication for future research. We reflect on the wider space of real-time prediction in esports.

Fig. 1. DotA 2 map from (http://dota2.gamepedia.com/Lane). The Radiant base is bottom left and Dire is top right. The colored circles are towers with Radiant in green and Dire in red.

II. DotA 2: GAMEPLAY

A DotA 2 match has 10 players in two teams called 'Dire' and 'Radiant' (5 players per team). Each match is played on a map (see Figure 1) which is split into two sides by a river. Each side of the map 'belongs' to a team and the end point of the game is when one team destroys the opposition's base located on the opposite side of the map (top right and bottom left in Figure 1). Before each match starts, each player picks a unique game character (hero) from 113 possible heroes for this data set (older DotA versions had fewer heroes and a recent update has 115 heroes). Each hero has different characteristics and abilities so the combination of heroes on each team can significantly affect which team wins or loses. The more advanced players consider their hero combinations very carefully. Once the match commences, the heroes play different roles where they aim to generate resources via fights against the rival team to progress through hero levels and become more powerful. Winning a game requires coordination within the team and the ability to react to the opposition's tactics and behavior. The game is real-time with hidden information, and good positioning and strategies will beat speed of play. Figure 2 is a screenshot of a game. We analyze standard 5v5 DotA 2 which is complex and player actions affect a long-term time window. Actions that have little short-term impact can form an overall team plan which adapts and shifts as the game evolves. This makes analyzing standard DotA 2 matches, and professional matches in particular, much more difficult and complex.

III. RELATED WORK

In this section we provide a structured review of existing approaches for predicting the winner of DotA 2 matches. The review is structured along three themes: 1) Data used; 2) Algorithms employed and: 3) Limitations in current work. Valve recently (12 March 2018) introduced a DotA 2 subscription service (Dota Plus) that includes a win prediction graph for viewers watching matches across all ability ranges. This demonstrates the value of prediction to enhance the viewer experience and helps lay the foundation for further academic research and industry development. No implementation details are available with respect to the data and algorithms used.
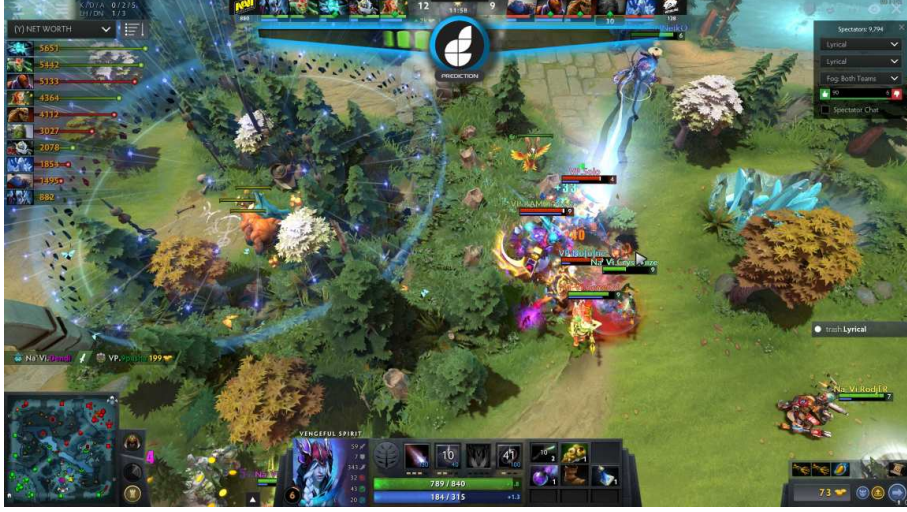
Fig. 2. Screenshot of a DotA 2 game. The on-screen display shows many statistics which can be daunting to novice viewers. Our predictor provides a simple overall statistic to illustrate which team is leading. It is displayed towards the top of the screen and surrounded by a blue gradient. In this screenshot "Radiant" is currently predicted to win.

However, Yu et al. [10] evaluated it using 72 professional tournament matches and found it had 68% win prediction accuracy at the half way point of the matches.

A. Data Used for Prediction

Previous work on win prediction in MOBAs has adopted a variety of data features representing different aspects of matches, which have been trained into a variety of machine learning algorithms (discussed in section III-B). The data for win prediction are sets of instances where each instance has a vector of features. The algorithms learn the association of data vectors with the winning team and then predict the winning team for new data vectors using the learned prediction model. We identified 11 data vectors used in the literature for DotA 2 win prediction, which can be categorized as follows:

Pre-game features: These features are generated before a match starts (in the hero selection phase). Heroes (player characters) in DotA 2 have different strengths and weaknesses so a good hero selection is important for team success.

1) **Hero vectors** are either 226-dimensional binary vectors (113 heroes in 2 teams)[11], [12], [13], [14], [15], [16], [17], or 113-dimensional tri-state vectors where x_i is 1 if hero i was in Radiant; x_i is -1, if hero i was in Dire and x_i is 0 otherwise [18], [14]. These vectors describe heroes, teams and hero-team combinations.

2) Authors have augmented these hero vectors with **specific hero combinations** such as 50 powerful two-hero combos selected using the authors' in-game knowledge [15] and represented as binary vectors.

3) Within a DotA 2 match, each hero plays a particular **role** in the game, for example, supporting other heroes. Yang et al. [19] analyzed 5 roles to label the nodes in their combat graphs and Makarov et al. [2] also used 5 roles and built separate models for each role using in-game features as the training data for each model. Makarov et al. [2] predict winners by combining the individual

models and weighting them to factor in the various roles' contribution to a win. Semenov et al. [14] generate a vector from the number of heroes playing each of 4 roles while [20] compared using 9 roles with just 3 roles as features and found 3 roles outperformed with respect to prediction accuracy with logistic regression.

In-game features: Many authors produce time-series vectors from in-game features that describe how the game develops and can assess game similarity through vector similarity. These feature vectors can also be trained into predictors and used for real-time prediction. Yang et al. [19] posit that in-game features contain the most useful information for prediction compared to pre-match or post-match features.

4) Many aspects of DotA 2 games can be extracted as **time-series vectors**. These vectors encompass features of heroes and other game entities such as non-hero characters, buildings, runes and spells, and Eggert et al. [20] incorporate player positions and details of hero fights. Most authors use a sliding window approach to generate fixed-width vectors [21], [10] or a time-series generated from the beginning of the game to the current time [17] for input to the various machine-learning algorithms. The features can be attributed to individual heroes, individual roles [2] or collectively to teams. Schubert et al. [5] evaluated a broad range of features. They identified that the rate and difference of the accumulation of rewards by teams, as well as their ability to kill the opposing team's heroes, are key in determining match outcomes. These features describe team encounters.

5) **Graph-based approaches**. Yang et al. [19] model hero interaction during in-game combats as a timestamped sequence of node graphs with the nodes representing the hero pairs dueling. Similarly, Kalyanaraman [12] analyzes the co-occurrence network of hero nodes for winning and losing teams where the weight of a hero-

pair (graph edge) is incremented when both heroes co-occur in the same team. They were able to identify communities in the graph representing hero sets that are frequently picked together on winning and losing teams.

6) Rioult et al. [4] analyze time-stamped **topological features** derived from the shape described by the positions of all players on a team as they move around the map (e.g., area, inertia, diameter and distance) which feed into a decision tree to predict the winning team. The spread of heroes in the team is important and Rioult et al. [4] found the distances of the heroes to the team's barycenter is most important for match prediction.

Post-game features: Other features are generated post-match and summarize the game, notably the game end state.

7) Kinkadze et al. [13] and Makarov et al. [2] use **post-match statistics** as features to train a prediction algorithm, such as team rewards, team kills and match duration [13] or gold and experience earned by each player [2]. Wang [16] notes that game duration is important as the win-rate of particular heroes and particular hero combinations varies according to game duration so Wang [16] subdivided games into 15 minute phases when predicting matches.

8) **Hero win rate** can be calculated either: as a coefficient using logistic regression of previous game statistics [22], using pairwise win rates (5 Radiant heroes x 5 Dire heroes = 25 hero combinations) [17] or as a team synergy calculated by summing the win rates of hero pairs in each team [13]. Kalyanaraman [12] uses a genetic algorithm to calculate success sets of heroes which contribute the most to victory from the co-occurrence network in their graph-based approach (see item 5).

9) The human **player's skill** is very important in determining match winners. It can be represented by their final score and current skill (skill rating percentile) [17] or their performance calculated using logistic regression on 17 different features [22].

10) The **Player-Hero skill** combines the player's skill with the hero success by calculating 8 features to describe the players previous play records using this hero [17].

11) **Social ties** inside the team (the degree of social friendship between team members represented by $max \# friends$) [22] are important factors in prediction.

B. Algorithms for MOBA Prediction

Machine learning (ML) is a field of computer science covering systems that learn "*when they change their behavior in a way that makes them perform better in the future*" [23]. These systems learn from data without being specifically programmed. Many ML algorithms (including regression) use supervised learning (or classification learning), where the algorithm learns a set of labeled (classified) example inputs, generates a model associating the input vectors with their respective classes (labels) and then classifies (or predicts) the class of unseen examples using the learned model. For DotA 2 win prediction, the algorithm effectively maps input vectors representing sets of game metrics to output labels (winning

team). The winning team can then be predicted for unseen vectors by applying the unseen vectors to the learned model and outputting a winning team prediction. A wide variety of machine learning algorithms have been used in the literature for supervised prediction of DotA 2 winners. The fundamental difference between these algorithms lies in how they build their models and how those models function internally.

Much of the previous win prediction work used logistic regression (LR) including: [18], [20], [13], [2], [22], [5], [15], [17]. LR had superior accuracy for win prediction compared to artificial neural networks [16], [17] and Random Forests (RFs) [12]. Kalyanaraman [12] found a tendency for RFs to over-fit the training data so they focused on LR and combined it with genetic algorithms (GAs) to extract sets of heroes with the highest winning rate. In contrast, Johansson et al. [21] showed that RFs had the highest prediction accuracy for their data vectors while Conley & Perry [11] found that k-nearest neighbor (kNN) outperformed LR as kNN can model the relationships inherent in the data better than LR. However, Johansson et al. [21] found that kNN (and support vector machines) were unsuitable due to the excessive training time (over 12 hours on 15,146 files). Rioult et al. [4] and Yang et al. [19] simply used decision trees (DTs) which are simple, easy to understand and allow rules to be extracted. Yu et al. [10] trained recurrent neural networks (RNNs) using 71,355 pro matches and predicted the winners in a small set of 72 professional matches. They achieved an accuracy of 71% at the half way stage of matches. We note that these matches may span multiple major game updates.

Authors have used combinations of methods. Semenov et al. [14] used both Factorization Machines (FMs) and XGBoost (XGB) (an enhancement of Random Forests that uses meta-learning (boosting rather than Random Forest's bagging) to derive the individual decision trees in the forest rather than random selections of trees). We analyze a similar algorithm in section V. In related work, Cleghern et al. [24] predicted hero health in DotA 2 using a combination of techniques: an ARMA model to predict small changes in health and non-homogeneous Poisson point process estimation (see [24]) to predict large changes in conjunction with logistic and linear regression to predict the sign and magnitude of the change. Our results in section V suggest that win prediction is difficult and no one technique excels so combining techniques into ensembles may well be necessary.

C. Summary and Limitations

Table I is an overview of the win prediction literature surveyed in this paper. It provides a simple comparison of the data and machine learning algorithms used by authors. The reporting of the data composition and details is inconsistent and how the authors process the data also varies. We include the accuracy achieved by the authors on their own data to show the spread of accuracies claimed. Hence, readers should be aware that the authors' data sets vary widely and direct comparison is not possible. Semenov et al. [14] speculated that the accuracy of the win prediction model depends on the skill level of the players. Hence, in table I we list the skill level

TABLE I
OVERVIEW OF THE FEATURES (NUMBERS IN SECTION III-A), ALGORITHMS, SIZE AND SKILL-LEVEL OF THE DATA, THE ACCURACY ACHIEVED AND THE TIME WHEN THE PREDICTION IS MADE FOR THE PAPERS IN THE LITERATURE. NOTE: PRE-GAME INDICATES AFTER THE HEROES ARE SELECTED AS ALL OF THESE PAPERS USE HERO DATA. NOTE, THESE DATASETS VARY IN SIZE, COMPOSITION AND DOTA 2 VERSION SO COMPARING ACCURACY NEEDS TO BE TREATED WITH CAUTION. ++[22] EVALUATED SINGLE FEATURES TO DETERMINE THEIR WORTH FOR WIN PREDICTION, HENCE ACCURACY IS LOWER. **[14] CALCULATED AUC RATHER THAN ACCURACY AND ##[10] USED ONLY 72 PRO GAMES FOR TESTING AFTER TRAINING WITH 71,355 PRO MATCHES. PLEASE SEE TEXT FOR ADDITIONAL COMMENTS.

Author	Year	1	2	3	4	5	6	7	8	9	10	11	Algorithm	Games	Skill	Accuracy	Time
Agarwala [18]	2014	*											LR	40,000	Low-High	62%	Pre-Game
Conley [11]	2013	*											kNN	18,000	Top 8%	70%	Pre-Game
Eggert [20]	2015												LR	-	-	-	not reported
Johansson [21]	2015	*			*				*				RF	59,208	V. High	82%	5 mins
Kalyanaraman [12]	2015	*		*	*	*		*	*				LR & GA	15,146	V. High	91%	Pre-Game
Kinkade [13]	2015							*					LR	18,500	V. High	73%	Pre-Game
Makarov [2]	2018			*	*		*		*	*			LR	-	Pro	72%	Halfway
Pobiedina [22]	2013											*	LR	87,204	Normal (60% win rate)	50%++	Pre-Game
Rioult [4]	2014												DT	1,120	-	90%	not reported
Schubert [5]	2016												LR	412	-	87%	after 10 fights
Semenov [14]	2017	*	*										FM & XGB	5,071,858	Normal-V. High	70%**	Pre-Game
Song [16]	2015	*		*	*	*		*					LR	6,000	-	68%	Pre-Game
Wang [16]	2016	*		*					*	*	*		LR	5,071,858	Normal-V. High	60%	Pre-Game
Yang [19]	2014												DT	407	Pro	80%	not reported
Yang [17]	2016	*			*								LR	20,631	V. High	72%	5 mins
Yu [10]	2018				*								RNN	72##	Pro	71%	10-90%

of the players' data collected by the various authors. The skill levels are not specified for some datasets. All DotA 2 players have a match-making rating (MMR) score quantifying their skill level (the higher the score the more skilled the player). It allows players of equal skill to be matched together in games. The average is 2,250 and the 99th percentile MMR is 4,100 (http://dota2.gamepedia.com/Matchmaking_Rating). Many of the datasets are described as "very high" skill but the authors do not quote score ranges for their data. None of the datasets used in the literature contain professional game data except [2], [19], [10] who each use a small number of professional games. Additionally, only two authors [21], [17] provide a prediction accuracy after 5 minutes 82% and 72% respectively and 20 minutes 99% and 81% respectively all using v. high skill data. Yu et al. [10] and Makarov et al. [2] who predict professional games both measure time as a percentage of the total game time. This is only known after the game and varies from 10 minutes up to 2 hours with an average game time of 40 minutes. Percentage of game time cannot be used for live prediction as the game length is not known until the end.

We detail how we address these in the following sections.

IV. DATASET AND PRE-PROCESSING

Previously, we ran a short feasibility study on 1,933 replays (**1.9K data**) [9]. Using the knowledge gained, we now analyze a much larger data set comprising 5,744 replays (**5.7K data**) and 186 professional tournament replays (**TI 2017**). Replays are binary files containing low-level game events that occurred when the match was played and are used by DotA 2 engines to recreate entire matches for re-watching and analysis. OpenDota (www.opendota.com) provide an API for accessing DotA 2 replay URLs that allows the end-user to request professional or public matches separately. We use this URL to download the file from Valve's servers. Our 5.7K data contains 23.97% professional matches (1,377 matches) and 4,367 public matches with extremely high MMR (>5000 which represents the 96th percentile (https://dota.rgp.io/mmr/), played between 27th March 2017 and 14th July 2017. We use this in a real tournament setting in section VI.

Valve do not provide a parser to extract information from replays, so the DotA 2 community has developed a range of mainly open-source parsers in a variety of programming languages. Among them is a fast, open-source Java-based parser, **Clarity**[2], by Martin Schrodt. We used Clarity to convert each replay's binary data into a CSV file of data vectors representing the game-state at each minute plus the winning team. These vectors form the inputs to our prediction models.

Another key feature of these data is the mix of pro and high skill non-pro games. There are only a limited number of professional matches for training the models and relying solely on professional training data limits the data size too much for many algorithms. The mechanics and 'meta-game' change significantly when new patches are released and we need data to cover these changes. A new patch may mean that previous data is redundant and has to be discarded if the heroes, mechanics and meta have changed significantly.

[2]https://github.com/skadistats/clarity

Our aim is to successfully predict professional matches so in our evaluation, we establish whether high skill public matches may be used as a proxy for professional match data to ensure sufficient training data for the prediction models. During our data collection period, there were no significant changes to the core game mechanics.

As outlined in Section III-A, a popular data feature for win prediction is time-series vectors of various in-game metrics (see table I). Thus, to evaluate professional win prediction and to find the best performing prediction algorithm, we use time-series features to represent our data sets.

In addition to selecting features used for static win prediction in previous work in esports analytics, such as kills, net worth and XP gained across teams, we discussed DotA game analytics with DotA 2 experts (commentators, professional coaches, high ranking players and long-term players). They were able to pinpoint key facets of the game and the set of most important features for analysis, for example tower damage and last hits (table II). An important constraint is that the live data stream in section VI only provides a subset of the features available in replay files. This constrains the features to those practically accessible during live game play so, in our analyses, we only used the features that were available live AND were picked by experts. Note that this limitation would not have applied in the prior work in the literature review, which conducted both training and evaluation with downloaded replay files. These authors could select from a larger set of features rather than the smaller live match feature set available to us.

We split the dataset into training and testing data. To evaluate win prediction using professional data versus mixed skill data, we use two data splits: 1) all data split into train and test which forms our baseline accuracy; and 2) mixed data for training with professional tournament data for testing. When analyzing all 5.7K data, we split the data 66% for training and 34% for testing as per Weka's train/test split ratio with the data sorted in chronological order. This ensures we never use future data to predict past data which could not happen in reality and is important in esports where data evolves over time (days, weeks, months etc.). To predict tournament data, we use the 5.7K data as the training set and 186 matches from 'The International 2017' DotA 2 tournament which took place (August 2-12, 2017)(http://wiki.teamliquid. net/dota2/The_International/2017) as a test set. These were the 186 tournament matches that had an associated replay file and lasted 20 minutes or more. We refer to this data set as **TI 2017**.

We determine the best parameters for the three algorithms under analysis by comparing the results on the training data set. In all evaluations, we ensured that we compared an equivalent number of algorithm, parameter and feature selections at all stages to ensure no bias.

A. In-Game Data

Our two in-game datasets comprise time-series data from a sliding window of 5-minutes. DotA 2 is fast moving and changes rapidly so a 5 minute window encompasses sufficient

game data for prediction without containing out-of-date gameplay data. For the evaluation in section V, we use one 5-minute sliding window at the 20 minute (halfway) game time; the average DotA 2 game lasts approximately 40 minutes[3]). The halfway point provides a suitable time-point for prediction evaluation. It encompasses the initial strategy but is before the all important late-game play, Yang et al. [17] noted that the later stages of matches are more important for determining the winners than the earlier stages. We refer to the 5.7K mixed dataset as **Mixed-InGame** and the TI 2017 tournament dataset as **Pro-InGame**. In a 5-minute window, there are 30 features each convoluted in the time domain plus the 5 time-stamps and the class label (either 'DireWin' or 'RadiantWin'). We generate feature vectors X_{rt} to represent the current game state for replay r at time t. Each feature is recalculated for each time stamp t. For each feature, we calculate the value for team Dire D, the value for team Radiant R, the difference between Radiant and Dire $R - D$ and the change (gradient) since the last timestamp for Dire dD and Radiant dR respectively. Table II lists the features.

To analyze a full game and generate a running prediction as in section VI, we train a separate win predictor for each minute through the game starting when we have collected sufficient data to form a vector, 5 minutes in for 5-minute sliding window. The learned model M_t at time t, is trained with a vector Xr_t representing the game state for replay file r at time t where: $Xr_t = xi_{t-4}, xi_{t-3}, ..., xi_t$ for all features i, and there is one model M_t for each minute interval between 5 and n where n is the maximum game length in minutes. Thus, the 5-minute sliding window for the 20-minute mark contains $\{xi_{16}, xi_{17}, xi_{18}, xi_{19}, xi_{20}\}$ for all features i.

V. EVALUATION

Our evaluation analyses predicting **professional** data using **mixed** data. There are insufficient professional data available for accurate model building as the training data would not cover the data space sufficiently. We split the mixed data into train and test sets to provide a prediction accuracy benchmark for comparison with predicting professional data from a mixed data model. This evaluation will therefore establish whether the mixed data can be used as proxy data for professional data in prediction model building for a live system. In [9] we used hero combinations for prediction (described in section III-A) to allow us to predict before the game play data starts but results were poor, achieving prediction accuracy of 55.8% on professional data while authors have achieved up to 70% accuracy on lower skill data (see table I). The professionals consider their picks very carefully and pick hero combinations that counter the opposition so hero combinations are not effective win predictors in professional data. This further serves to illustrate the increased complexities of predicting professional data compared to predicting non-professional data.

A. Algorithms

As shown in section III-B and table I, LR and RF [25] are popular algorithms in the literature for predicting winners in

[3]https://dota.rgp.io

TABLE II
THE IN-GAME FEATURES USED TO PRODUCE THE VECTORS TO TRAIN OUR MACHINE LEARNING PREDICTORS. THERE IS ONE VECTOR FOR EACH TIMESTAMP. EACH VECTOR IS TRAINED INTO A SEPARATE MACHINE LEARNING MODEL FOR THAT TIMESTAMP, E.G., A MODEL FOR 5 MINUTES, A MODEL FOR 6 MINUTES ETC.

Feature/Metric	Description
Team Damage Dealt	This represents the amount of damage each player dealt to enemy entities since the game began. We sum the individual totals each minute to get the team totals for that minute.
Team Kills	We use the team kills metric in Clarity which counts the number of enemy heroes killed since the game began.
Team Last Hits	At each minute timestamp, we use the Clarity team last hits metric (who hit last when an enemy entity died) to count each teams' last hits since the game began.
Team Net Worth	To calculate team net worth, we sum the net worth of the individual team members at each time stamp (minute). Net worth is the sum of the gold in the bank, the gold value of a player's items in the courier and of those in their inventory (purchase value, not sale value).
Team Tower Damage	This represents the sum of the total damage all players in the team have dealt to enemy towers since the game began. We extract the team total from Clarity each minute.
Team XP Gained	We calculate team experience by summing each team members' experience at each minute. XP is earned by being within a specific radius of a dying enemy unit. It is used to level up individual heroes in the game.

DotA 2 matches. LightGBM has outperformed other gradient boosting algorithms in classification and prediction tasks [26] and Semenov et al. [14] used GBM for win prediction in DotA 2. Results for neural networks were not compelling (underperforming the algorithms in section III-B) and the newer deep learning methods require much larger training data sets than are available here. Thus, we use both LR and RF along with LightGBM to analyze our hypothesis that combining professional game data with high skill public data can be used to accurately predict the winners of professional games.

For classification, **LR** produces a linear model. It uses a logistic function of the data features (known as explanatory variables) to estimate the probabilities for each class:

$$P(win) = \sigma(w_0 + \sum_{i=1}^{n} w_i x_i),$$

where $\sigma(a) = (1 + exp(-a))^{-1}$ is an activation function, w_i is the weight (coefficient) applied to feature x_i and X has n features. LR does not consider sets of features or dependencies among the features. It only estimates the importance of the individual features with respect to the prediction.

RFs are ensembles of decision trees generated using bagging. They use averaging to improve the prediction accuracy and prevent over-fitting. Each tree in the forest is independent and learns a different version of the dataset; equal in size to the training set. This versioned dataset is generated from the original training data using random sampling with replacement. The versioned dataset will therefore contain some duplicates. RF builds the set of trees by randomly choosing a subset of features at each split and then selecting the feature within this subset that optimally splits the set of classes. To allow the RF to predict, it uses majority voting on the prediction of all trees in the forest. Unlike LR, RFs do consider combinations of features as they are essentially rule-based algorithms where the rules are determined by the tree branches.

Microsoft's **LightGBM** gradient boosting framework is based on decision tree algorithms. It generates an ensemble of decision trees and splits the trees leaf-wise using the greedy best-fit expansion [27]. Continuous-valued features are discretized into bins using histogram based algorithms [28]. LightGBM then uses a gradient descent procedure to generate trees and minimize the loss by expanding the leaf with the maximum delta loss. In our evaluations, we minimize the log-loss function. Expanding trees leaf-wise can reduce loss more than a level-wise expansion [27]. However, the leaf-wise algorithm may cause over-fitting particularly when the data set is small. LightGBM uses an additional parameter, *max-depth*, to limit the depth of the trees and avoid over-fitting - the trees can still grow leaf-wise. As with RF, LightGBM considers feature combinations and dependencies.

B. Algorithm Configurations for In-Game Data

For comparing the prediction accuracy, we trained a Weka LR algorithm, a Weka RF algorithm and the Microsoft LightGBM algorithm with the **Mixed-InGame** and **Pro-InGame** data. To analyze the accuracy across configurations, we varied the parameters for the three algorithms. For LR, we varied the ridge in the log-likelihood, for RF we varied the number of trees (iterations in Weka) and for LightGBM we varied the iterations in conjunction with the number of bins and leaves. Additionally, we used the Weka feature selector CfsSubsetEval with BestFirstSearch [29] to compare the algorithm configurations' accuracies.

Eggert et al. [20] used Weka to evaluate feature selection [23]. Their results showed that a 'wrapper' [30] selector produced the highest accuracy with their data set. It uses the algorithm itself to evaluate and select features. We compared its results to Correlation-based Feature Subset Selection [31], a 'filter' [30] method that examines greedily selected feature subsets, independently of the algorithm. It favors subsets containing features that are highly correlated to the class but uncorrelated to each other to minimize feature redundancy. CfsSubsetEval had higher accuracy on various datasets when we have evaluated it in the past [32] and, in particular, for DotA 2 in [9] so we use that here.

C. Predicting using In-Game Data

Table III shows the win prediction accuracies of the various algorithm configurations. All configurations perform significantly better than random guess which forms a naïve baseline. The highest accuracy is achieved using either all features or the features selected by CfsSubsetEval. The two ensemble decision tree algorithms have higher accuracy when the model

is built from all features whereas LR has higher accuracy using the features selected by CfsSubsetEval.

For the **Mixed-InGame** data, the highest accuracy is 77.51%, using a RF algorithm and all features. However, all accuracies are very similar ranging from 77.24% to 77.51% for all 3 algorithms and their configurations. For the **Pro-InGame** data, accuracies ranged from 70.81% to 74.59%. The highest accuracy is 74.59% for both the RF algorithm with all features and LR using the features selected by CfsSubsetEval. There is more of a variation in accuracy for the professional data compared to the mixed skill data.

VI. Real-time system implementation

Having developed and evaluated our system using mixed-skill training data to predict professional games in this paper and in [9], we produced a working prototype and evaluated it during a live esports tournament. Figure 3 shows a system diagram. The training module (left) uses Opendota's API to periodically retrieve the URLs of high-skill and professional matches (1). Using these URLs, the training module then retrieves and downloads the full replay files (2 / 3). The downloaded replay files are processed by an adapted version of the Clarity parser, which generates the required features for training the model (4).

The biggest challenge for a live prediction system is accessing data describing the state of a live game. DotA 2 has a real-time interface called *Game State Integration (GSI)*. However, it is poorly documented by the publisher, and only a handful of unofficial resources detail its workings[4,5]. On a conceptual level, GSI works by placing a JSON-formatted configuration file in a special sub-directory of the local game client (see Figure 3, label 5). Once configured, the game generates real-time updates about the game's state, as soon to the game client is observing a game in spectator mode.

There are two ways of watching a DotA 2 game. 1), we can tune in to any live game via a function called DotA TV. This, in essence, is a live data stream of the match delivered to a watching client. While this feature is available for professional games and its game state can be accessed, DotA TV usually has a 2 minute broadcast delay, rendering this mode unsuitable for real-time prediction. 2) the only way to watch in real-time is to add observer clients to what is referred to as the 'Lobby'. A lobby is a virtual room that is used, among other things, to stage professional matches. Prior to each match, the tournament organiser creates a lobby, inviting the 10 players, as well as a series of 'spectators'. These 'spectators' are not audience members, but members of the production that need to access the game in real-time (e.g. virtual camera operators). If gamestate is configured for an observing client in the lobby, it produces actual real-time snapshots of the game in configurable intervals. Those gamestate snapshots are formatted as JSON objects, and sent as an HTTP request to the configured address and port. To receive those updates, we have to create a HTTP web service listening at the specified port,

[4]https://developer.valvesoftware.com/wiki/Counter-Strike:_Global_Offensive_Game_State_Integration
[5]https://github.com/antonpup/Dota2GSI

and parse the received game-state JSON (see Figure 3, label 6). A custom-written live parser written in C# extracts the required features from the live match data, and sends the feature to the prediction model. The model then produces a prediction of the winning team, and the confidence in its prediction (the number of trees in the majority class / the total number of trees).

It is important to note the intricacies regarding the data provided by GSI and their implication on prediction algorithms. The data provided by the GSI is only provided in configurable intervals, and its timing is not accurate. Each frame of the JSON snapshot does contains the game time it represents, however, only accurate to the second so exact timings of frames need to be guessed by measuring the time elapsed between receiving the last frame. Consequently, the features generated at the exact minute marks, as required by the models, are estimates. This may lead to slightly inaccurate values for the live-features that are passed into the model, which may decrease accuracy. By comparing accurate features from parsing the replay files with features produced by the GSI we could conclude that those deviations are minimal and, as the following evaluation shows, accuracy of the systems was satisfactory. The third issue with accessing real-time data is that the software needs to be run in a Live Lobby, requiring active support by the tournament host. Alternatively, live prediction can be run on the DotA TV stream of a live match, however, this adds a significant delay in the data acquisition pipeline, and thus affects the timeliness of the prediction.

A. Evaluation

We tested the described system at ESL One Hamburg 2017 (Oct 26-29), one of the largest international DotA 2 tournaments. In [33] we analyse observational ethnographic data on how our tool impacted commentary and content production. We conclude that even simple graphical overlays of data-driven insights, can have measurable effects on the commentary and quality of coverage. With support from ESL, the tournament organiser, we could join the Live Lobbies and generate real-time predictions for all 28 games over the course of a four day schedule. The knockout stage took place in an arena with 20,000 fans, and was watched by over 25 million people. Our system ran continuously during the tournament, and was monitored by a human operator, who took qualitative notes about the prediction results during the tournament. Starting at five minutes into the game, the system generates minute-wise prediction results of a winning team and a confidence. For each match, the prediction results were saved as time-series data, along with the raw vectors used for the prediction. When a game concluded, the winner was added to the log file. Based on this data we calculated prediction accuracy at each timestamp, see Figure 4. Due to the low sample size (N=28) the accuracy varies between 70% and 90%. Between 5 and 20 minutes, prediction accuracy moves within the 70% - 80% range, while between 20 and 30 minutes, accuracy moves between 80% and 90%. Notably, at the 5 minute mark (first prediction of each game), the system reached an 85% accuracy.

We plotted a time-series chart of prediction results for each of the 28 games to analyze the consistency of prediction within

TABLE III

PREDICTION ACCURACY OF THE VARIOUS CONFIGURATIONS OF ALGORITHMS ON THE 'MIXED' AND THE 'PROFESSIONAL' 5.7K DATA. THE HIGHEST %
IS SHOWN IN BOLD FOR EACH DATASET. THE TABLE COMPARES THE RESULTS FOR LR, RF AND LighGBM WITH A SINGLE TIME-SERIES FEATURE
($Networth_{R-D}$), ALL FEATURES AND FEATURES SELECTED BY CfsSubsetEval.

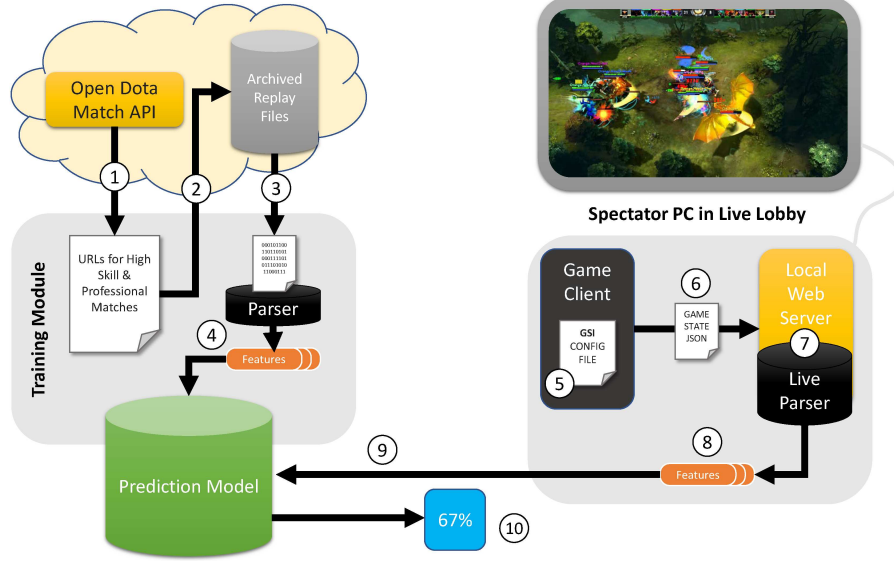|	Mixed-InGame			Pro-InGame		
Predictor	1-Attr	All	CFS Select	1-Attr	All	CFS Select
LR	74.06	77.35	77.41	71.89	72.97	**74.59**
RF	76.36	**77.51**	77.24	70.81	**74.59**	70.81
GBM	76.41	77.46	77.25	72.43	73.51	71.89
Random Guess	50.00	50.00	50.00	50.00	50.00	50.00

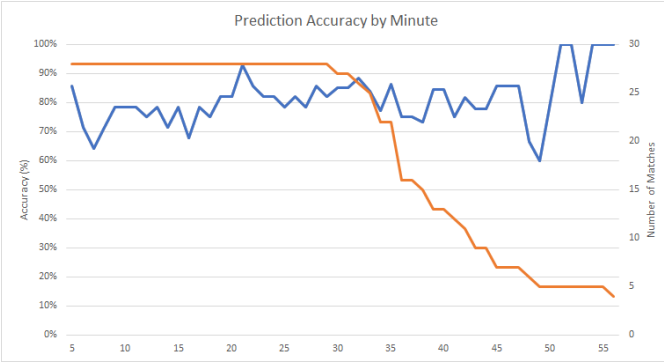Fig. 3. System diagram of real-time prediction software.

Fig. 4. Live prediction accuracy across 28 games at ESL One Hamburg 2017 showing the number of games that lasted at least that length of time (right y-axis, orange line) and the average prediction accuracy across those games (left y-axis, blue line) with confidence intervals excluded for clearer presentation.

each match. Two exemplars are shown in figure 5. We plot prediction confidence (blue) so that a positive value is a prediction result in favour of the winning team (correct prediction), while a negative value is a wrong prediction. The algorithm returns a winning team and a confidence between 0.5 and 1.0. For the losing team, we multiply the confidence by -1 and plot it on the negative sector of the y-axis (see the rightmost plot) to clearly emphasize any confidence swings. Note we only generated prediction models from 5 to 57 minutes as there

were insufficient training data for longer games (fewer than 100 examples). While the two charts are just examples, they do represent the two archetypes of outcomes. Of the 28 games, 11 games had no swing in prediction result, while 17 games had 2 or more swings. Figure 6 (left) shows a distribution of the number of swings observed across all 28 matches.

VII. DISCUSSION AND ANALYSIS

This paper represents a case study for real-time professional win prediction to generate a simple in-game statistic towards informing the audience. Semenov et al. [14] posited that the accuracy of win prediction varies across skill levels and that higher skill games are harder to predict. Semenov et al. did not evaluate professional games and we would expect these to be even harder to predict given the complex and evolving nature of the game an the strategies adopted by professional teams [8], [20]. We established a baseline framework by predicting the winners of professional matches using models trained with mixed data. The results of our analyses in section V suggest slightly lower accuracy when a model trained with mixed-skill data predicts winners in professional data than when a model trained with mixed data predicts winners in a mixed data test set. However, with careful algorithm selection and parameter optimization, the results for predicting professional data are only slightly worse with accuracies up to 74.59% achieved by RF with all features and LR with CfsSubsetEval features.
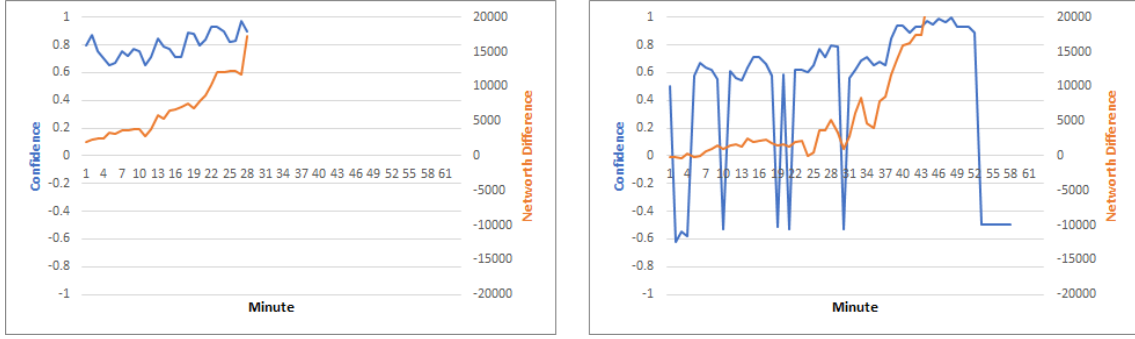
Fig. 5. Time series for prediction confidence (blue) and net worth difference (orange) between teams. For both charts, a positive value indicates a value in favor of the winning team, i.e. the winning team is shown above the x-axis. Note x-axis label 1 in both charts is the 5 minute mark (the 1st prediction). In the rightmost chart, the prediction stops at x-axis label 52 due to the lack of training data for games that long.
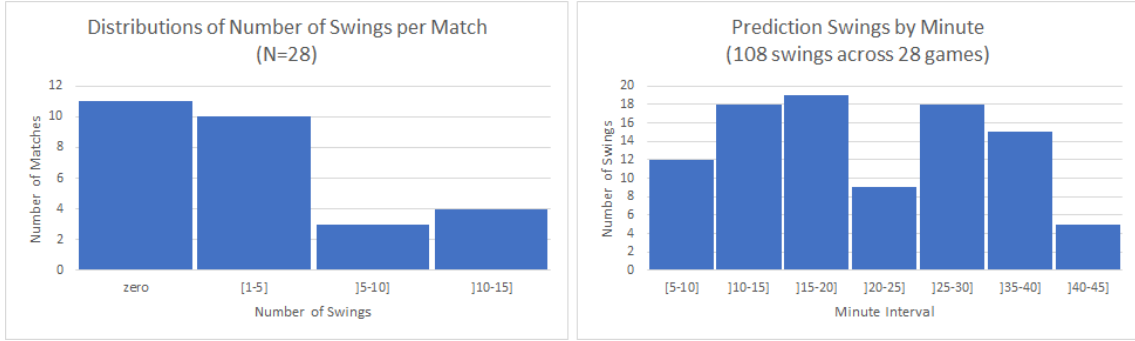

Fig. 6. Distribution of number of swings in prediction results.

The hypothesis that professionals play differently and generate different data than non-professionals, is supported by skill statistics from c400,000 players[3], analyses by [8], and by the chart in Figure 8. Pro players are the top 1% by skill and >5K MMR data used here is the top 4% by skill. Figure 8 shows that the duration of games in the mixed data sets has increased slightly between April 2017 and August 2017, 13.3% of the mixed games in [9] lasted 50 minutes or longer compared to 17.1% of the 5.7K mixed games. However, the duration of the professional data has fallen much more. 26.5% of the [9] professional games lasted 50 minutes or longer but only 11.1% of the TI2017 games lasted 50 minutes or longer. This contrast reinforces that we need to carefully consider professional data and ensure that we optimize our algorithms by testing multiple configurations. We also need to constantly update any machine learning model used to predict winners in DotA 2 as professional games are constantly changing as teams update or invent new strategies, and the underlying game is changed and adjusted through live operations.

In our previous 1.9K dataset [9], the most important data feature selected by both CfsSubsetEval score and frequency of use in LightGBM trees was $Kills_{R-D}$. CfsSubsetEval selects features independently of any algorithm and is an objective measure to support our LightGBM tree findings. The most important feature selected by both CfsSubsetEval score and frequency of use in LightGBM trees in the 5.7K data is $Networth_{R-D}$. This identifies that the data has evolved over time and the key features for win prediction have changed according to CfsSubsetEval. In-game features represent the
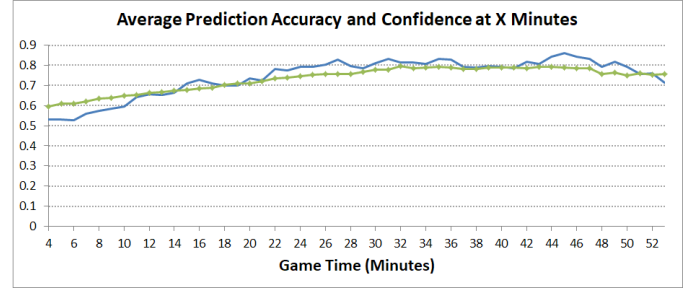

Fig. 7. Chart showing the RF prediction accuracy (in blue) and the prediction confidence (in green with markers) at X minutes into the games. The RF is trained with 5.7K mixed data and tested using the TI2017 pro data. It uses majority voting so the confidence is the number of trees in the majority class / the total number of trees.

current game state. These features effectively represent who is currently leading at each timestamp. We analyzed prediction at the 20-minute stage which is half-way through an average length match. The further the game progresses, the more accurate the in-game predictor should become. However, Yang et al. [17] noted that the later stages of matches are more important for determining the winners than the earlier stages with late game actions generally more important than early game actions. This further complicates our ability to predict the winners. Additionally, Yang et al. and Johansson et al. [17], [21] both found that the longer a match lasts then the lower the prediction accuracy is at X minutes into the game. Longer matches are generally more unpredictable throughout.
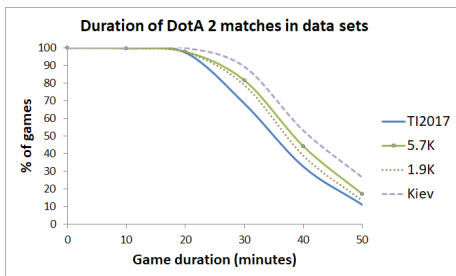
Fig. 8. Chart showing the game duration (minutes) as a % of total games in the 5.7K and TI2017 datasets with the % for the 1.9K and Kiev Tournament (professional) data (analyzed in [9]) shown as a dotted and dashed line respectively for comparison.

Figure 7 shows that the prediction accuracy fluctuates over time. There is a general upward trend in accuracy until the 35 minute mark when the accuracy drops. It rises again around 40 minutes and then falls from 45 minutes. This supports the hypothesis that the longer the match lasts then the lower the on-going prediction accuracy [17], [21].

For many esports viewers, the number and meaning of the statistics displayed can be confusing. It is often difficult to tell who is leading as the statistics can be contradictory. Figure 2 shows a typical match screen. Win prediction provides an overarching game statistic that assists the audience with judging the current balance of the game analogous to the score in many traditional sports such as football. In section VI, we detailed how we have successfully implemented and evaluated our win prediction models on live data at an esports tournament. An interesting paradox with win prediction, which is not considered in the esports literature, is that if the prediction accuracy is too low then the audience will not find the predictions believable. Conversely, if we could predict with 100% accuracy which team will win at 5 minutes then there would be little point continuing watching or playing and the game would not be enjoyable. Emphasizing this point, the DotA Plus tool provided by DotA 2 developer Valve, is according to Yu et al. [10] not great compared to their model, however, this has not prevented the player community from adopting the tool. Esports, as with all sports, need to maintain an element of doubt to be enjoyable. We have ensured a sufficiently high accuracy. We now need to ensure audience enjoyment and can perform A/B testing with the win prediction at different accuracy levels to assess enjoyment.

VIII. CONCLUSION AND FUTURE WORK

We identified that there has been very limited analysis of professional player data in DotA 2 mainly due to the sparsity of this data and the as yet emergent nature of esports analytics [5]. The commercial relevance and value of match prediction relies on algorithms that can analyze professional matches as this is where the audience interest is placed. The number of spectators watching professional esports is rising and esports viewing is becoming a popular social activity [7]. However, these professional games are fast-paced and change rapidly making them difficult to understand. Many esports, including DotA 2, display an array of statistics on screen but there is

no single 'score' so viewers need help to comprehend the on-screen action. Even casual players need professional games explained [5]. The win condition in DotA 2 is to destroy the enemy base. The likelihood of a team being able to destroy the opposition base is predicated on the economic advantage that team holds. Calculating and even understanding the economics is complex. To make esports more understandable, sociable and to broaden its appeal, broadcasters can provide in-game statistics to improve the spectator experience (see figure 2). Predicting the likely winners of games as they progress provides a simple, easily understood in-game statistic for the audience analogous to traditional sports scores.

Section III-A and table I provide the first comprehensive survey of academic research into DotA2 win prediction. This research analyses a range of skill levels, but there is no prior work on predicting professional games at scale. By evaluating this research, we have identified a number of limitations:

1) Professional game data: Previously reported work has not evaluated win prediction in professional games other than a small analysis of in-game combats [19] and evaluations on small datasets by Makarov et al. [2] and Yu et al. [10]. The most popular games among spectators are professional games so this is where the commercial value lies due to the number of viewers [20], [14], [5]. However, professional data is scarce (noted by [21]) and live tournament data provides fewer data features than the archived replay files so methods presented in prior work may not be applicable to live professional data or may not have sufficient accuracy to be usable.

2) Skill comparison: Previous work does not evaluate data from both professional and non-professional games together. We established that data from non-professional games can mitigate the lack of professional game data.

3) Meta-game changes: Previous research on DotA 2 win prediction has collected match data over time periods that crossed significant changes to the game (when new game patches were released). These patches significantly alter the 'meta-game' (the high level strategies adopted by players and teams beyond the standard rule of the game). Altering this meta-game introduce variations into the collected data. As noted by [14], the data being analyzed needs to be comparable for verifiable prediction.

4) Live and real-time prediction No previous work has implemented a working real-time prediction system for professional data and deployed it in real tournament settings. We were able to deploy our system at a major international tournament. We discussed the practical application of a live prediction system in section VI.

The aim of this work is to explain professional esports matches to the audience as the matches progress by accurately predicting the winner throughout the game. As there is only a limited number of professional matches for training our models, we aimed to supplement professional data with extremely high skill non-professional data to make sure that there are sufficient data for training. We found that the win prediction accuracy of professional matches using mixed professional and non-professional training data is only slightly lower than our benchmark accuracy when predicting by splitting the mixed data into training and test sets. We demonstrated

that evaluating multiple prediction algorithms coupled with algorithm optimization such as feature selection and parameter optimization is vital and a broad range of configurations need to be evaluated to ensure maximum accuracy.

We have performed a feasibility analysis at an international tournament on live data as described in section VI and overcame a series of pitfalls and issues with live data. We even shaped our feature set to ensure that the live and historical data are consistent with fewer features available in live data.

Our approach described here provides a baseline for future development. We can augment this approach with more data as matches become available. We can incorporate new data features such as those discussed in section III-A to provide a richer training set. We can add meta-learning with multiple predictors as recommended by [24] to analyze the data from multiple viewpoints.

In further work, we will analyze the prediction paradox discussed in section VII where inaccurate predictions will disappoint the audience and too accurate predictions will decrease enjoyment as the game would not be exciting if the outcome is known early game. We can find the ideal trade-off between prediction credibility and the enjoyment of esports games for all viewers. People across all levels of understanding will then be able to watch the games together. We will then explore the potential of applying our win prediction methods to digital games more broadly to maximize player and audience engagement. Other esports games with publicly accessible data include Team Fortress 2 by Valve Corporation or Counter Strike: Global Offensive (CS:GO) by Hidden Path Entertainment and Valve Corporation.

In the future, when similar high-frequency and detailed datasets are available from domains such as the Internet of Things (IoT) [34], we can start to apply our live prediction to human behavioral data in the real world.

IX. Data Access Statement

The data set used in this evaluation comprises matches between 27th March and 14th July 2017. New replays are created daily so new data are available. In the paper, we have provided details regarding how to scrape the data and are happy to help others with obtaining such data for themselves.

References

[1] J. Hamari and M. Sjöblom, "What is esports and why do people watch it?" *Internet research*, vol. 27, no. 2, pp. 211–232, 2017.
[2] I. Makarov et al., "Predicting winning team and probabilistic ratings in dota 2 and counter-strike: Global offensive video games," in *Procs 6th International Conference on Analysis of Images, Social Networks and Texts, Moscow, Russia, July 27–29, 2017.*, 2018, pp. 183–196.
[3] M. Mora-Cantallops and M.-. Sicilia, "Moba games: A literature review," *Entertainment Computing*, vol. 26, pp. 128 – 138, 2018.
[4] F. Rioult et al., "Mining tracks of competitive video games," *AASRI Procedia*, vol. 8, pp. 82–87, 2014.
[5] M. Schubert, A. Drachen, and T. Mahlmann, "Esports analytics through encounter detection," in *Procs of the MIT Sloan Sports Analytics Conference*, 2016.
[6] L. Yang, "Goldman sachs research: via youtube," 2017. [Online]. Available: http://www.goldmansachs.com/our-thinking/pages/buzzwords/multimedia/video-12-esports.html
[7] Superdata Research, "European esports conference brief," 2017. [Online]. Available: http://strivesponsorship.com/wp-content/uploads/2017/04/Superdata-2017-esports-market-brief.pdf
[8] A. Drachen, et al., "Skill-based differences in spatio-temporal team behavior in defence of the ancients 2 (dota 2)," in *Proceedings of the IEEE Consumer Electronics Society Games, Entertainment, Media Conference.* IEEE, 2014.
[9] V. Hodge et al., "Win prediction in esports: Mixed-rank match prediction in multi-player online battle arena games," *ArXiv e-prints (CS:AI)*, Nov. 2017. [Online]. Available: https://arxiv.org/abs/1711.06498
[10] L. Yu et al., "Moba-slice: A time slice based evaluation framework of relative advantage between teams in moba games," *arXiv preprint arXiv:1807.08360*, 2018.
[11] K. Conley and D. Perry, "How does he saw me? a recommendation engine for picking heroes in dota 2," tech. rep., Stanford University, Tech. Rep., 2013.
[12] K. Kalyanaraman, "To win or not to win? a prediction model to determine the outcome of a dota2 match," 2015.
[13] N. Kinkade, L. Jolla, and K. Lim, "Dota 2 win prediction," tech. rep., University of California San Diego, Tech. Rep., 2015.
[14] A. Semenov et al., "Performance of machine learning algorithms in predicting game outcome from drafts in dota 2," in *Procs 5th International Conference on Analysis of Images, Social Networks and Texts, Yekaterinburg, Russia, April 7-9, 2016.*, 2017, pp. 26–37.
[15] K. Song, T. Zhang, and C. Ma, "Predicting the winning side of dota2," tech. rep., Stanford University, Tech. Rep., 2015.
[16] W. Wang, "Predicting multiplayer online battle arena (moba) game outcome based on hero draft data," Master's thesis, Masters thesis, Dublin, National College of Ireland, 2016.
[17] Y. Yang, T. Qin, and Y.-H. Lei, "Real-time eSports Match Result Prediction," *ArXiv e-prints*, Dec. 2016.
[18] A. Agarwala and M. Pearce, "Learning dota 2 team compositions," tech. rep., Stanford University, Tech. Rep., 2014.
[19] P. Yang, B. E. Harrison, and D. L. Roberts, "Identifying patterns in combat that are predictive of success in moba games." in *FDG*, 2014.
[20] C. Eggert et al., "Classification of player roles in the team-based multiplayer game dota 2," in *International Conference on Entertainment Computing.* Springer, 2015, pp. 112–125.
[21] F. Johansson and J. Wikström, "Result prediction by mining replays in dota 2," Master's thesis, Blekinge Institute of Technology, Karlskrona, Sweden, 2015.
[22] N. Pobiedina et al., "Ranking factors of team success," in *Procs of the 22nd International Conference on World Wide Web.* ACM, 2013, pp. 1185–1194.
[23] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations.* San Francisco: Morgan Kaufmann, 2000.
[24] Z. Cleghern et al., "Predicting future states in dota 2 using value-split models of time series attribute data," in *Procs of the 12th International Conference on the Foundations of Digital Games.* ACM, 2017.
[25] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.
[26] C. Lucchese et al., "Rankeval: An evaluation and analysis framework for learning-to-rank solutions," in *Procs 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2017, pp. 1281–1284.
[27] H. Shi, "Best-first decision tree learning," Ph.D. dissertation, The University of Waikato, 2007.
[28] J. Shafer, R. Agrawal, and M. Mehta, "Sprint: A scalable parallel classi er for data mining," in *Procs 1996 International Conference on Very Large DataBases*, 1996, pp. 544–555.
[29] E. Frank, M. A. Hall, and I. H. Witten, *The WEKA Workbench. Online Appendix for "Data Mining: Practical machine learning tools and techniques.* Morgan Kaufmann, 2016.
[30] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial intelligence*, vol. 97, no. 1-2, pp. 273–324, 1997.
[31] M. A. Hall, "Correlation-based feature selection for machine learning," Ph.D. dissertation, The University of Waikato, 1999.
[32] V. Hodge, T. Jackson, and J. Austin, "A binary neural network framework for attribute selection and prediction," in *Procs of 4th Int'l Conf. on Neural Computation Theory and Applications*, 2012, pp. 510–515.
[33] F. Block et al., "Narrative bytes: Data-driven content production in esports," in *Procs of the ACM Int'l Conference on Interactive Experiences for TV and Online Video (TVX 2018).* ACM, 2018, pp. 29–41.
[34] Y. Sun, et al., "Internet of things and big data analytics for smart and connected communities," *IEEE Access*, vol. 4, pp. 766–773, 2016.