# Match Outcome Prediction in MOBAs with Deep-Learning-based Team Fight Models

*Note: Sub-titles are not captured in Xplore and should not be used

1st Given Name Surname
*dept. name of organization (of Aff.)*
*name of organization (of Aff.)*
City, Country
email address or ORCID

2nd Given Name Surname
*dept. name of organization (of Aff.)*
*name of organization (of Aff.)*
City, Country
email address or ORCID

3rd Given Name Surname
*dept. name of organization (of Aff.)*
*name of organization (of Aff.)*
City, Country
email address or ORCID

4th Given Name Surname
*dept. name of organization (of Aff.)*
*name of organization (of Aff.)*
City, Country
email address or ORCID

5th Given Name Surname
*dept. name of organization (of Aff.)*
*name of organization (of Aff.)*
City, Country
email address or ORCID

6th Given Name Surname
*dept. name of organization (of Aff.)*
*name of organization (of Aff.)*
City, Country
email address or ORCID

***Abstract*—Esports are complex computer games that are played competitively. *DOTA 2* is one of the most popular esports titles worldwide and field a rapidly evolving gameplay across a large virtual arena. Due to this complexity, commentators, audiences, and players can miss events happening during live matches. This makes it necessary to utilize esports analytics to bring attention to important events that can aide in match outcome prediction. One of the most important events are team fights, when players from opposing teams encounter and battle each other. Despite their importance across strategy, gameplay and audience experience, team fights remain relatively unexplored. Their role and potential to support match prediction models not well understood. This paper presents a definition of team fights in *DOTA 2* and proposes an algorithm to extract and quantify them. Deep learning models were then used to analyze the influence of team fights on overall match outcomes. Results from eight different types of Recurrent Neural Network (RNN) models revealed that given a minimum of 2 team fights within 5 minutes of game play, our models were able to correctly classify the outcome in over 50% of all games. This accuracy increased to over 70% if given 32 minutes of game time to train the model. These results highlight the use of team fights to inform and improve match prediction models. Furthermore, the work presented contributes to the development and automation of in-game camera movements to guide spectators, and provide commentators with real-time data-derived predictions.**

***Index Terms*—Esports Analytics, Deep Learning, Neural Network, DOTA2, Team Fight**

## I. Introduction

Esports is a term used to describe video games which are played competitively [1], [2]. Esports are varied in their form and gameplay, and today attract audiences and players in the hundreds of millions worldwide [3]. Uniquely for esports, as compared with traditional sports, there is a large degree of overlap between audiences and players. As a sub-sector of the games industry, esports has grown immensely in recent years, and today comprise a multi-billion dollar sector [4]. Due to the high degree of innovation and technology adoption in the sector, and the detailed data available from many titles, esports has become a test bed for research across many domains, not the least data science [1], [5]–[7].

In recent years, esports has also become a focus for research on how to enhance sports broadcasting in the future, and how to utilize data to enhance the viewing experience, or provide interactivity [2], [7]. One of the most popular genre of esports games, in terms of audience numbers and academic research, is the Multi-Player Online Battle Arena (MOBA). This genre include titles such as *Dota 2* and *League of Legends*, each with major tournaments sporting price pools in the dozens of millions, often driven fully or partly by the community [8].

Similar to the traditional sports, the tacit collaborations within each team are always the highlight of the game and at times can be the turning point for the match. For example, each team fight in MOBA game such as DOTA 2 is significant since it might determine the landscape for the rest of the match [9]. The commentators could easily tell whether a team fight happens, but it requires years of experiences to understand the impact of each team fight and convey it clearly to audiences [2]. Furthermore, although previous studies have focused on player encounter interactions [1], no detailed analysis exists that explores whether team fights can predict match outcome.

Previous research within esports analytics tends to focus on the whole game as a unit of analysis, as compared to investigating events occurring within matches [6]. The work here builds on previous analyses and extends it by focusing on the impact of team fights on match outcome prediction. With *DOTA 2* as the case study subject, a team fight detection algorithm is deployed, and match outcome predictions based

Identify applicable funding agency here. If none, delete this.

on the team fight features were conducted.

This paper analyzes the impact of individual team fights in a *DOTA 2* match to the match outcome. The purpose is not to build the most accurate real-time match predicting model, but to explore if team fights on their own provides a foundation for prediction models. These models will then contribute towards integrating machine learning and audience engagement, allowing for a context grounded approach to match outcome commentary and analysis.

It consists of two main components: a) The team fight detection algorithm would separate all individual team combats and aggregate the relevant team fight statistics; and b) the predictive model would then analyze the impact of each team fight and make the match outcome prediction.

### A. Contribution

This paper contributes to two current areas of research in esports analytics, 1) esports match outcome prediction and 2) esports audience engagement along with machine learning powered match analysis for commentators [2], [7]. It does through the following areas:

1) Build on DOTA 2 team fight detection algorithms developed in past research [1]. Refine past encounter algorithms to focus on team fights to speed up data preprocessing and extract the most important information from a match.
2) Utilizing team fight features to predict overall match outcome through deep learning models. This paper employed extensions of the Recurrent Neural Network model to predict the outcome of a DOTA 2 match.
3) Developing the basis for a machine learning powered match analysis tool for audience and commentator engagement. This paper experimented with various data masking variations to simulate the progress of a live match in order to identify models that are capable of providing match outcome predictions with limited data.

### B. Ethics Approval

Ethics approval was granted by the computer science department at the University of York. Data was collected through readily and freely available means and no personal or otherwise identifiable information was collected, stored or utilised at any stage beyond the publicly available replay game files. No identifiable information was extracted from those files and data was stored in an aggregate format to prevent deanonymization.

## II. Background: DOTA 2 Game-play

In DOTA 2, there are two opposing teams, named Radiant and Dire. These fight against each other in a virtual arena, seeking to destroy the opposing teams' base (called the "ancient"), and protect their own ancient at the same time. Each team consists of five players and each player control a different hero with unique abilities and skills to fight within this closed environment. The bases of both teams are distributed diagonally and the map is divided into different sections for each team (Figure 1). There are three major lanes across the map and are designated as top lane, middle lane, and bottom lane. Except for the three major lanes, there are jungle areas, including various properties (outposts, shops, effigy buildings), as well as neutral enemies, often referred as neutral creeps or neutral monsters. There are also multiple towers protecting each team's ancients and lanes. Each team has to take down some of the towers in order to eventually destroy the base of enemy team. At the home base of each team, there is a fountain where heroes would revive after being killed and waiting for the revive countdown. Another interesting feature of the DOTA 2 map is the fog of war. The fog blocks the sight of heroes so that each player would only have vision within a given range around the hero they control, friendly heroes/units, or buildings, and cannot observe other places on the map. However, players could obtain additional vision on the map by purchasing and placing items such as "wards".

At the time of writing, there are 121 heroes for players to choose. Each hero possesses distinct abilities and skills. Different combination of heroes would build a team with unique strengths and weaknesses. Each player would control their respective hero/units to engage in combat until the enemy ancient is exposed and destroyed. During the match, heroes would kill enemy creeps, heroes, or units to gain gold and experience. The gold enables players to buy more powerful items and the experience enables players to level-up or learn new abilities so that they could outperform the enemy team.

In a typical DOTA 2 match, each player would farm by collecting gold and gaining experience by killing enemy creeps in their respective lane in the early game. As the game progresses, players tend to gather together and fight against the enemy team in a group. We call this type of group fight a team fight. Team fights are always the highlights for both players and audiences since the effects of group spell casting are splendid and the winning team in the team fight would gain huge advantages, especially in the late game. This is one of the reasons we chose to focus this research on team fights. Most of time, team fights would have a tremendous impact to the game and the result of a team fight might change the landscape of the entire match [9].

## III. Related Work: Esports Analytics

The domain of esports analytics emerged over the past decade, and has expanded rapidly since. The literature contains a broad area of work and has seen an accelerating pace of publications in recent years [11]. Esports analytics was defined by Schubert et al. [1] 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." The definition of Schubert et al. [1] highlights a fundamental challenge in esports, namely making complex and fast-paced games comprehensible to players and audiences alike.

Thanks to the ready availability of data from esports games from public API systems provided by the game publishers,

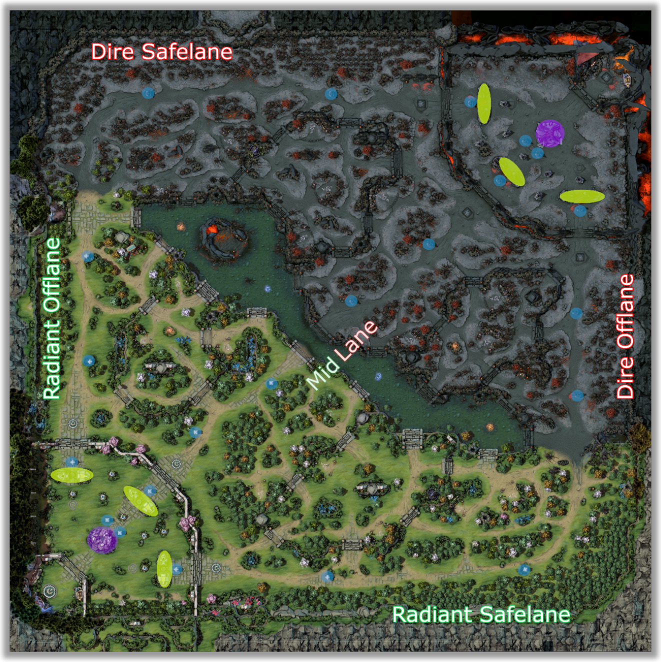



Fig. 1: The map of DOTA 2. Towers are highlighted in blue, Racks ("barracks") are highlighted in yellow, and the Ancients are highlighted in purple. The three lanes are marked with the names associated with their corresponding teams [10]

esports analytics has become a fertile ground for research in machine learning, AI, and sports, with high-dimensional and high-volume data across amateur to professional levels being utilized [6], [12]–[14].

Predicting the result of esports matches has emerged as a key topic in esports analytics. Not only does such predictions provide interesting material for broadcasting and audience engagement [2], [7], but are also of use to inform players and teams for the purpose of training.

Prior studies demonstrated the application of machine learning algorithms in DOTA 2 match analysis. Demedium et al. [12] utilized unsupervised machine learning algorithm to classify the role of players in DOTA 2 games, while Eggert et al. [9] used supervised learning algorithms instead to identify player roles in DOTA 2 game. Sifa et al. [14] detected outliers occurring during a game for improving the commentator-driven storytelling experience. Drachen et al. [15] investigated the relationship between team skill and spatio-temporal behaviour of the team using time series clustering. Katona et al. [11] utilized a feedforward neural network with shared weights to predict the probability of a player hero being killed within a five second window. Yang et al. [16] modeled DOTA 2 games using graphs and constructed Decision Trees using extracted patterns to predict the match outcome with 80% accuracy. Semenov et al. [13] experimented with the possibility of predicting DOTA 2 match outcome from draft picks using Factorization Machines (0.66 AUC) and XGBoost classifier (0.65 AUC).

More relevant to our research, Yang et al. [17] performed real-time match outcome prediction using individual players' match history and real-time features and achieved up to 93.73 percent accuracy when including up to 40 minutes of game data. Hodge et al. [6] also examined real-time game result prediction for DOTA 2 using standard machine learning models and achieved 85 percent accuracy after 5 minutes of gameplay.

Past literature, as summarized above, deal with different aspects of an esports match. Although various researches focus on match prediction and analysis [6], [13], none have dealt with the influence of team fights, which are important events that could drastically alter the outcome of an entire match [9]. Our work aims to bridge this gap in the existing literature and does so by focusing on real-time game outcome prediction for DOTA 2. However, different from prior researches, our prediction models are based on the concept of *team fight* adapted from *encounter* components defined by Schubert et al. [1]. The goal of our work is to provide an innovative way of retaining spectator engagement by providing match outcome predictions after each team fight. This framework would generate data-driven insights to assist commentators and augment the audience experience [7].

## IV. Data Collection and Preprocessing

### A. Data

In this study, a dataset comprising a total of 1,493 professional-level DOTA 2 matches, from the time the game was in patch 7.27, was gathered using the OpenDota API [18]. The data contains all behavioral actions of players during matches, as replay files need to be detailed enough for the game client to rerun games, providing highly granular data about DOTA 2 matches. The Clarity Analyzer Library was used to parse match replay files into JSON format [19]. Spatio-temporal information was extracted on a per-second level.

### B. Preprossessing for Team Fight Detection

We first parsed the raw JSON data using SQL queries into tabular format and removed games that were only partially recorded. The remaining data consisted of 1,456 games with 747 won by the Radiant and 709 won by the Dire. Each row consisted of a hero action and/or performance at a certain time within the game. This data was then fed into our team fight detection algorithm.

### C. Feature Engineering for Game Prediction

We detected all team fights in our data, using the team fight detection algorithm defined and explained in the next section (V-B Team Fight Definition), and created an output table. We then joined the processed data with the output table to label each row of the data based on the following rules: First, if the data entry is during a team fight, label it with a team fight number in the order the fight happened in that specific game. For example, if it is the first team fight that occurred in a game, label it as 1. Second, if the data entry is not during a team fight, the label will be Null. Once we have successfully labeled the entire data set with the according team fight number, we filtered out the rows that were labeled as Null because we only need team fight relevant data for our subsequent use.

Next, we aggregated the data set by team fight for each of the 1,456 unique DOTA 2 games and summarized team fight statistics. For the purpose of building predictive models that can predict the final winner of the game (the Radiant team or the Dire team), we performed another level of aggregation to summarize team fight statistics by faction. More specifically, for each team fight we calculated the number of hero kills, assists, deaths, total damage dealt during team fight, total gold obtained during team fight, and the number of players who participated in the team fight for both the Radiant and the Dire teams [10]. Besides these general statistics, we also generated additional features from the data, defined below, that can be helpful for our predictive models.

*1) Total Crowd Control Time:* In most modern role-playing games and MOBA games, crowd control time (or disable time) is defined as effects that cause affected players to partially or fully lose control of their heroes [20] [12]. The effects include but are not limited to stun, slow, silence, mute, break, hex, and disarm. Each of these effects can be caused by a variety of hero spells and items. Crowd control has important strategic influence in a DOTA 2 team fight, because preventing enemy heroes from moving, attacking, and casting spells can greatly increase the likelihood of killing enemy heroes or preventing teammates from dying, thus helping the team to gain advantages for the overall game. We generated this features by summing up the total disable time applied to a team and used that as the total crowd control time for the opposing team.

*2) Total Spell Damage:* Spell damage makes up the majority of damage in the early game team fights because players have not yet gained enough gold to purchase items that boost up their attack damage. Therefore, total spell damage dealt during team fights is a crucial feature for early game success. We calculated this feature by summing up all damages labeled as a spell damage event.

*3) Total Auto Attack Damage:* Auto attack damage refers to damage dealt using a hero's regular attack. Attack damage can be amplified by abilities or by purchasing items that directly increase a hero's attacking power or provide on-hit effects, such as critical strike [21]. Auto attack damage makes up the majority of damage in late game team fights where the carries of each team are equipped with valuable items. Therefore, total auto attack damage dealt by a team represents how effectively the carries can put pressure on the enemy heroes to stay alive during team fights. This is a crucial feature for late game success because dying in the late game in DOTA 2 has severe penalties. We calculated this feature by summing up all damages labeled as an auto-attack damage event.

*4) Total Item Damage:* The remaining component of damage dealt to the enemy team during team fights is item damage. We need this feature for our predictive model because some DOTA 2 heroes heavily rely on key items to deal damage, such as Dagon for Tinker. We calculated this feature by summing up all damages that were incurred by items during a team fight.

*5) Total Distance Traveled:* There can be a lot of moving during a team fight for many different purposes. Here we explain two different scenarios. One possibility is that one team is winning the team fight, so they are chasing the enemy heroes in order to reap more kills. Another possibility is that one team has more ranged heroes while the other team has more melee heroes, so the ranged team performed a lot of "kiting", which means they are constantly moving to prevent the melee heroes from landing attacks on them. We want to capture these types of information in our predictive models and therefore we generated this total distance traveled feature by summing up heroes' total displacement in the DOTA 2 arena for each team during a team fight.

*6) Number of Buildings Destroyed:* As we discussed in Section II Background: DOTA 2 Game-play, players have to destroy the ancients of the opposing team as well as the buildings protecting the ancients to win the game. One of the goals when engaging in a large scale team fight in a DOTA 2 match is to destroy one or more of the enemy buildings. Thus, the number of destroyed enemy buildings during a team fight has strategic influence for the final match outcome and therefore we want to include this feature in our predictive models. We generated this number of buildings destroyed feature by counting the number of buildings on the map that deplete to zero health during a team fight.

## V. Methodology and Results

The aim of this research is to create a model that is capable of predicting a DOTA 2 match outcome using only features within team fights. To achieve this goal, we first developed a team fight detection algorithm to identify team fights. We then utilized this algorithm to extract and aggregate features used in our supervised prediction modeling. This section describes our team fight detection algorithm and match outcome predictions.

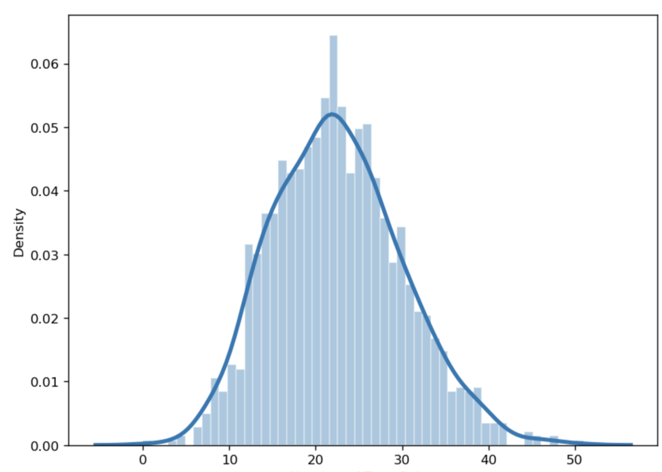



Fig. 2: Density plot of the number of team fights during matches. The majority of matches have 20 to 25 team fights

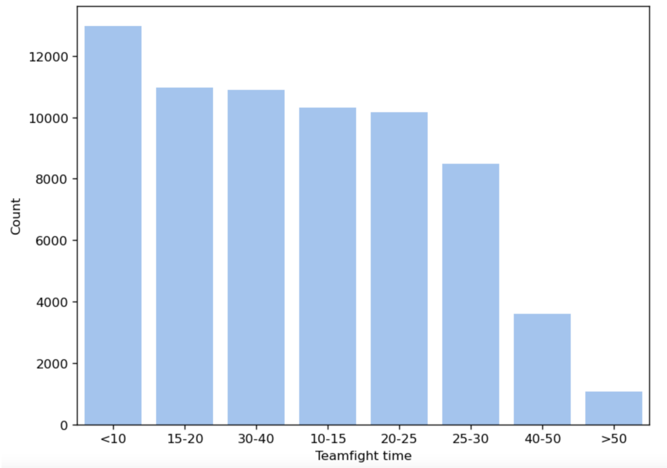



Fig. 3: Barchart showing the distribution of team fight start time. Most team fights are within the first 30 minutes of game time

### A. *Team Fights*

Before carrying on with our research, we should first understand what are team fights. Although the specific details vary across definitions, team fights occur when players from opposing teams meet within the arena of DOTA 2. Team fights are viewed as important to determining the outcome of matches [9] and also form central components of the narrative developed by commentators and casters [2]. However, while team fights have been utilized conceptually in multiple esports research publications [9], [22], a formal definition has not been widely agreed upon in the esports community [1]. In this section, we attempt to provide a flexible, broadly applicable definition and model of team fights which takes into account the spatio-temporal nature of DOTA 2 as highlighted by previous work, e.g. Schubert et al. [1] and Eggert et al. [9].

Past research utilized rules based algorithms to detect hero encounters within DOTA 2 [1]. In our research, we referenced this paper's definition of encounter as the basis for our team fight definition and constructed our own approach for identifying team fights by further enhancing the encounter detection algorithm as described in Section V-B Team Fight Definition.

### B. *Team Fight Definition*

Generally speaking, we define a team fight as an encounter of player units from both teams with one side of the encounter having at least two players from the same team, and at least one killing event happened during the encounter. This definition filters out 1-on-1 and 2-on-1 trades and trades, while focusing on fights that have a more significant impact on both teams.

We first define the two teams are $T_1$ and $T_2$, each with five player units, which are represented as $u_i$. We also define a function called $D(u_i, u_j)$ to calculate the distance of two player units. In addition, we define a player link $L(u_i, u_j)$ to describe the player units relationships. There are three kinds of player links we think are essential in defining a team fight, which are combat link, support link and kill link.

*1) Combat Link:* We define a combat link as a player units relationship where the two player units are from different teams and the distance between them are within the general attack range $\epsilon_a$ (700 units) of player units in DOTA 2. It is represented as a $L_c(u_i, u_j)$ where $u_i \in T_1$ and $u_j \in T_2$ and $D(u_i, u_j) <= \epsilon_a$.

*2) Support Link:* We define a support link as a player units relationship where the two player units are from the same team and the distance between them are within the general healing range $\epsilon_h$ (400 units) of player units in DOTA 2. It is represented as a $L_s(u_i, u_j)$ where $u_i \in T_1$ and $u_j \in T_1$ and $D(u_i, u_j) <= \epsilon_h$.

*3) Kill Link:* We define a kill link as a player units relationship where the two player units are from different teams and one player unit kills the other player unit. It is represented as a $L_k(u_i, u_j)$ where $u_i \in T_1$ and $u_j \in T_2$ and $u_i$ has killed $u_2$.

*4) Encounter Component:* We define an encounter component $EC_t$ as a subset of player units where each player unit is connected to all other units via a path which consists of combat and support links. For an encounter component, there should be at least one combat link and one support link, which indicates that there are at least two player units from the same team and at least two player units from different teams, shown in Figure 4. An Encounter Component depicts a kind of cross-team interaction of player units at a specific time tick $t$. We represent an encounter component as a graph called $G(U, E)$ where $U$ is a set of nodes or player units and $E$ is a set of edges or player links. For player units, there $\exists u_i \in U$ from $T_1$ and $\exists u_j \in U$ from $T_2$; for player links, there $\exists e_i \in E$ is $L_c$ and there $\exists e_j \in E$ is $L_s$.

*5) Successor:* We define a successor $EC_{t+\Delta t}$ as a subsequent encounter component to a sequence of encounter components whose last component is $EC_t$. The time difference between the successor and the last previous encounter component $\Delta t$ should not exceed a time threshold $\tau$. And an additional requirement is that there should $\exists u_i \in EC_t$ from $T_1$ such that $u_i \in EC_{t+\Delta t}$ and $\exists u_j \in EC_t$ from $T_2$ such that $u_j \in EC_{t+\Delta t}$.

*6) Encounter:* We define an encounter as a sequence of encounter components where each encounter component at time tick $t$ is a successor of a previous encounter component. An encounter is dynamic in terms of its components, since player units can join and leave during the entire time span of an encounter.

*7) Team Fight:* Finally, we define a team fight as an encounter which contains at least one kill link, or to say a team fight should be a special form of encounter which involves killing activity. The reason for making this definition is that team fights with kills are more consequential than non-kill team fights. If someone dies in a fight, there is a clear punishment to the team – gold and experience (XP) gain to the other team as the most direct consequence. While there can be many ”encounters”, we believe the ones that involve killing have a more tangible impact on the game and can provide us with useful information for making predictions on game result.

Fig. 4: Illustration of Combat Link (red) and Support Link (blue) during Encounter. When there exist at least one Combat Link and at least one Support Link, the algorithm detects it as one encounter Component [23].

### *C. Algorithm Design*

After defining team fights, we followed and implemented an algorithm outlined in the paper Esports Analytics Through Encounter Detection [1] to automatically detect encounter components from raw game data. We then added an extra constraint of requiring a kill event to happen during the encounter to classify it as a team fight.

The algorithm works by reading in a stream of player unit positions, and at each tick, we constantly updated the position and the distance, and identified the possible combat components. Then, we identified the possible predecessors of the combat components, and try to link components together as encounters based on specific conditions described above. Finally, we filtered out the encounters that contain one or more kill links and identified them as team fights. A list of team fight encounters is outputted by the algorithm.

### *D. Team Fight Detection Results*

The output of our team fight detection algorithm given a single DOTA 2 game is a list of team fight encounters as defined in Section V-C Algorithm Design. This list is a homogeneous list of Encounter objects, i.e. the team fight encounters that we detected from a given DOTA 2 game. We can convert the list of team fights into an output table as comma-separated values. The output table has N number of rows with respect to the total number of team fight we detected from the input DOTA 2 game. Each row has the following attributes: team fight number (first team fight of the game, second team fight of the game, etc.), team fight start time and end time, a list of players who participated in the team fight, and whether there is any death during the team fight. We can then use the generated output table for our predictive models.

### *E. Match Outcome Prediction*

The cleaned data was further aggregated by game, team fight number, and team faction. The resulting data containing the generated features discussed previously were then used to classify the overall outcome of the match.

### *F. Logistic Regression*

We first utilized a logistic regression model, a linear classification algorithm [24], on the entire data set to understand the association between different team fight features and the odds of a team winning or losing the entire match. Our results revealed that teams who achieved higher number of kills (Odds ratio, OR: 1.412) and assists (OR: 1.025) during team fights had a higher probability of winning the entire match. On the other hand, teams that destroyed more buildings during team fights (OR: 0.697) and participated in team fights with longer duration (0.978) were less likely to win the overall match. The logistic regression model managed to only achieve a training accuracy of 57.1%, despite using all available data, indicating the existence of significant non-linear associations between team fight features and the overall outcome of the match. Thus, we also employed random forest algorithms to verify any non-linear associations between team fight performance and match outcome.

### *G. Random Forest*

Next, we fitted two random forest models [25] with two different data treatment approaches:

1) Each record is representative of a team fight. We defined a binary response variable where 1 indicates Radiant wins and 0 indicates Dire wins. Each column is the difference between two teams for each variable.
2) Using a combination of two records (one for Radiant and one for Dire) to represent one team fight. Each record contains the exact values of team fight variables for each team.

The two approaches resulted in similar training accuracy, but quite different feature importance outcomes. Since the first approach returns a combination of the two teams’ performances, its model accuracy (Test Accuracy: 66%) is 2% higher than the second data treatment (Test Accuracy: 64%). As we can see from the feature importance plot, compared to the first model, the second model devalues the importance of Total Auto Attack and emphasizes the importance of Team Fight duration for both teams.

As shown in the data treatment 1 column in Table I, the first approach emphasized the importance of total auto attack damage, change in net worth, and total distance traveled as the three variables most influential in determining whether a match is classified as Radiant or Dire winning.

Data treatment 2’s feature importance can be seen in the data treatment 2 column in Table I. Although change in net worth and total distance traveled are still extremely important, total auto attack damage’s importance was eclipsed by team fight duration.

TABLE I: Top 6 important features for random forest models

| Features | Importance Score | |
|---|---|---|
| | Data Treatment 1 | Data Treatment 2 |
| Total Auto Attack | 0.118 | 0.099 |
| Change In Net Worth | 0.117 | 0.150 |
| Total Distance Traveled | 0.113 | 0.134 |
| Total Spell Damage | 0.106 | 0.112 |
| Total Damage During team fight | 0.104 | 0.098 |
| Team Fight Duration | 0.096 | 0.125 |

Though we've obtained higher accuracies from using Random Forest models, there is still room for model improvements considering that the random forest model is unable to treat the performance of each team fight sequentially. In the next section we employ deep learning algorithms to model the sequential nature of the team fights.

### H. Recurrent Neural Networks

In our logistic regression and random forest models, we treated each team's performance in a team fight as an isolated (a.k.a. independant) event. Although this allowed us to use less sophisticated and computationally faster models, it does not take into account the sequential and ordered nature of each team fight within a match. Recurrent neural networks (RNN) are a type of deep learning model that retains the memory of previous inputs within the network's internal state [26]–[29]. This construction allows past inputs or contextual information to influence the model's output. This makes RNNs some of the best deep learning algorithms to model sequential data [30].

However, RNN models suffer from the problem of vanishing gradients. The influence of an input would decay or explode exponentially as the RNN model trains. In order to address this issue, we have chosen to utilize two different algorithms that extends the simple RNN model [26], [27].

*1) Bidirectional RNNs:* Bidirectional RNNs are a type of RNN that allows the model to access both past and future context. The input data sequence is fed to two separate recurrent hidden layers that are connected to the same output layer [26]. In terms of DOTA 2, the use of a bidirectional construction allows the model to utilize team fight information in the past and future. Bidirectional model constructions also work with RNN extensions such as LSTM and GRU.

*2) Long Short-Term Memory:* Long Short-Term Memory (LSTM) [29] is a neural network that is a special RNN, that replaces summation units in the hidden layers with memory blocks, which are a type of recurrently connected subnets. Multiplicative gates within LSTM memory cells allow the algorithm to store and utilize information over long periods of time [26]. LSTMs are able to decide whether the content derived from an input should be overwritten at each time step. Thus, it is better able to retain important features over a long distance [27].

*3) Gated Recurrent Unit Networks:* A gated recurrent unit (GRU) is a recurrent unit that can adapt and capture dependencies from different time scales. GRUs also have gating units similar to LSTM, but they do not have separate memory cells. Thus, GRUs do not control the exposure of hidden memory content. Other units in the network can use the full content within the memory. GRUs are simpler in design compared to LSTM (i.e. containing a reduced number of parameters to be learned) without sacrificing model performance [27].

*4) RNN Model Results:* We applied four different RNN models to our data: LSTM, GRU, bidirectional LSTM, and bidirectional GRU. These four models were also tested using two different architecture variants with either one layer or two layers. All features were standardized to between 0 and 1 before modeling. The model consisted of an initial layer with 256 nodes. If the architecture tested had two layers, the output of the first layer was then fed into a second layer with 128 nodes. This was followed by a fully connected layer with softmax activation. Loss was calculated using categorical entropy with an Adam learning rate optimizer. Early stopping was applied if the model's validation accuracy did not improve in 20 epochs. All models were trained up to a maximum of 51 epochs using a batch size of 256. Ten percent of the entire data was used as the holdout test set. The remaining training data was further split into training and validation sets (90:10).

We first tested all models using the complete training and test data. Results can be seen in Table II. All eight models were re-trained 10 times and their performance on the hold out test set was calculated. It can be seen that the bidirectional GRU model with two layers out performed all other models with an average test set accuracy of 79.2%. However, this accuracy is achieved only with the complete training and test data available.

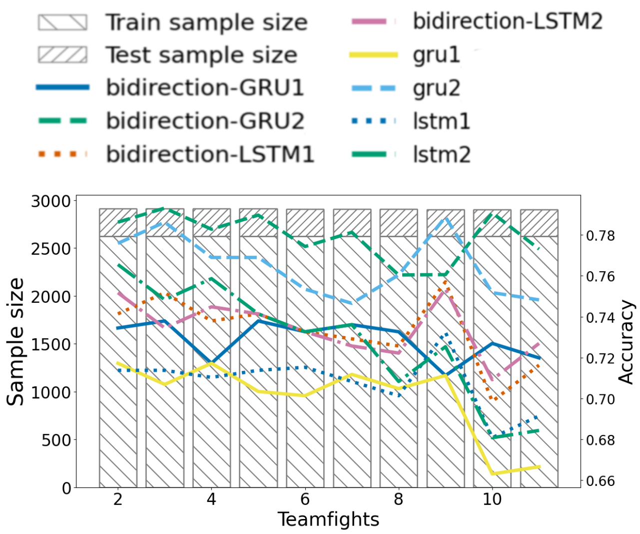
TABLE II: RNN model performance with all available team fight data in training and test set

| Model Type | GRU 1 | GRU 2 | LSTM 1 | LSTM 2 |
|---|---|---|---|---|
| Regular | 0.712 | 0.759 | 0.711 | 0.753 |
| Bidirectional | 0.734 | 0.792 | 0.738 | 0.742 |

*Note*: Training sample size: 2,622. Test sample size: 290. Value shown are the average test accuracies over 10 runs.

DOTA 2 games can vary widely with some games filled with frequent but inconsequential skirmishes, and others dominated by a few game changing team fights. In order to ensure that our models are not dominated by outliers, we also trained each model on either a filtered training set or the entire training set. All models were then compared using the holdout test set accuracy. The holdout test set was also filtered accordingly to simulate incomplete real time match data. In the first part of each analysis, we trained models using the complete training data set, but tested them using a hold out test set that has been filtered according to different criteria. We then trained the same model again, but this time also using incomplete training data filtered according to the same criteria as the test set. The goal of this evaluation is to identify weaknesses within the models if they were given incomplete match data to train with. This is especially important for predicting match outcomes immediately after patch changes to DOTA 2.

We start with the number of team fights as the filter/cut-off criteria. In Figure 5a test data sets were filtered according to the number of team fights. Only matches that have at least the corresponding number of team fights will be included in the model predictions. The eight lines within the figure are the prediction accuracy for the test set by various RNN models. The stacked bar chart in the background provide information on the size of the train and test sets. The y-axis for model accuracy are on the right and the y-axis for sample size are on the left. X-axis values indicate the cutoff for train and/or test datasets, using either minutes of game time or the number of team fights. Results revealed that by ensuring only games with two or more team fights are used in the holdout test, the accuracy for all models would be increased to 70% or higher. However, this benefit does not increase if we were to limit our predictions to only games with a high number of team fights.




(a) RNN model performance using number of team fights as cutoff for test data


(b) RNN model performance using number of team fights as cutoff for training and test data

Fig. 5: RNN model performance with restricted team fight data

In Figure 5b both training and test data sets were filtered according to the number of team fights. Only matches that have at least the corresponding number of team fights will be included in the model predictions. Results revealed that by ensuring only games with two or more team fights are used in the holdout test, the accuracy for all models would be increased to 80% or higher. However, this benefit did not resulted in large increases if we were to limit our predictions to only games with a high number of team fights.

Next, we evaluated each model using game time as the cut-off criteria. Performances of the eight models tested using team fights that started before a certain game time are shown in Figure 6a. Performances for all eight models increased drastically as the the number of minutes increased. As more team fight data is added into model training, accuracy changes from lower than 50% to over 70%. Most model performances were similar, although the two layer bidirectional GRU model had the highest performance when all team fights up to the 32 minute mark are included for the test data.

Performances of the eight models trained and tested using team fights that started before a certain game time are shown in Figure 6b. Similar to the models trained using the full data set, performances for all eight models increased drastically as the the number of minutes increased. As more team fight data is added into model training, accuracy changes from lower than 49% to slightly over 70%. Most model performances were also similar, but the two layer bidirectional GRU model had the highest overall performance when team fights up to the 20 minute mark are included for both the training and test data.

Results in Figures 5a and 6a indicated that having both a team fight number cutoff combined with a game time cutoff would result in a model that is best able to generalize to future matches. The results of all eight models with the test data filtered by game time and containing at least two team fights are shown in Figure 6c. The results indicate that by having two filters the number of test data is drastically reduced. This resulted in stronger initial performance compared to 6a, but similar performance afterwards until the 20 minute game time cutoff with the requirement of also having at least two team fights. Best model performances at 32 minutes of game time increased to over 72%.

The results of all eight models with both the training and test data filtered by game time and containing at least two team fights are shown in Figure 6d. The results indicate that by having two filters the number of training and test data is reduced. This resulted in poorer initial performance compared to 6a, but similar, if slightly higher, performance afterwards until the 32 minute game time cutoff with the requirement of also having at least two team fights. Model performances at 32 minutes of game time is slightly higher than 72%.

To verify the effect of having both filters, we also tested the same model configurations with a more stringent number of team fights. The results of all eight models with the test data filtered by game time and containing at least three team fights are shown in Figure 6e. Similar to what was observed with the requirement of two team fights, training the model on filtered data resulted in better initial performance, but no accuracy improvements afterwards. Best model performance at 32 minutes of game time is around 72%.

The results of all eight models with both the training and test data filtered by game time and containing at least three team fights are shown in Figure 6f. Similar to what was observed in Figure 6d, training the model on filtered data resulted in poorer initial performance, but a slight improvement after 20 minutes of game time. Model performance at 32 minutes of game time is slightly higher than 72%.

Overall, results in Figure 6 indicate that single layer models tend to achieve higher accuracy with less data. However, two layer models are able to provide better results as more training and/or test data becomes available.

## VI. Discussion

In this research we first expanded on the encounter detection algorithm developed by Schubert et al. [1] by defining team fight as encounters with at least one kill. Our approach allowed us to extract important team fight based attributes to use in our match outcome prediction models.

Match outcome prediction results revealed that RNN models were able to predict the outcome of an ongoing DOTA 2 match. Our results indicate that it is possible to utilize deep learning models in predicting the outcome of real-time ongoing matches. Accurate predictions also do not necessitate the use of all game related data, but only features related to team fights. By leveraging only team fight performance in the first 5 minutes of a match, our models were able to achieve over 50% accuracy in predicting final match outcome. If an additional filter requiring a specific number of team fights to be included were added, the model's accuracy would improve to over 50% using only the first 5 minutes of data.

The performance of all eight models were similar in terms of accuracy. No model performed best in all scenarios. GRU models, especially two layer bidirectional GRUs were able to achieve slightly higher performance when there were more data due to game time cutoffs. This is especially evident when both training and test data were filtered according to game time. However, all eight models had less than 1% difference in accuracy when only 5 minutes of data were included. This indicates that if ample computation resources are available, all eight models could be used to create a more accurate prediction. Different models should be deployed for different stages of the game in order to maximize the advantages of each given limited data. Based on our results, to achieve the highest possible prediction accuracy, both a one layer model and a two layer model should be employed. The one layer model would be used to predict match outcome if less than 10 minutes of data is available. Once the match has progressed beyond 10 minutes a two layer model, preferably a two layer bi-directional GRU model, should be used.

The implications of our results are twofold. First, we have established that it is possible to build a real-time prediction system for ongoing DOTA 2 matches using RNN models only trained on team fight data. The accuracy improves as the match unfolds and more team fights occur, similar to the result obtained by Hodge et al. [6]. Although our models did not achieve better accuracy compared to past research [6], these results do indicate that team fights serves as an important data point for predicting overall match outcomes.

The model presented could be repeatedly updated in real time to provide an esports audience and/or commentator with progressively more accurate predictions of the overall match outcome, similar to models proposed by others, e.g. Hodge et al. [6] and Schubert et al. [1].

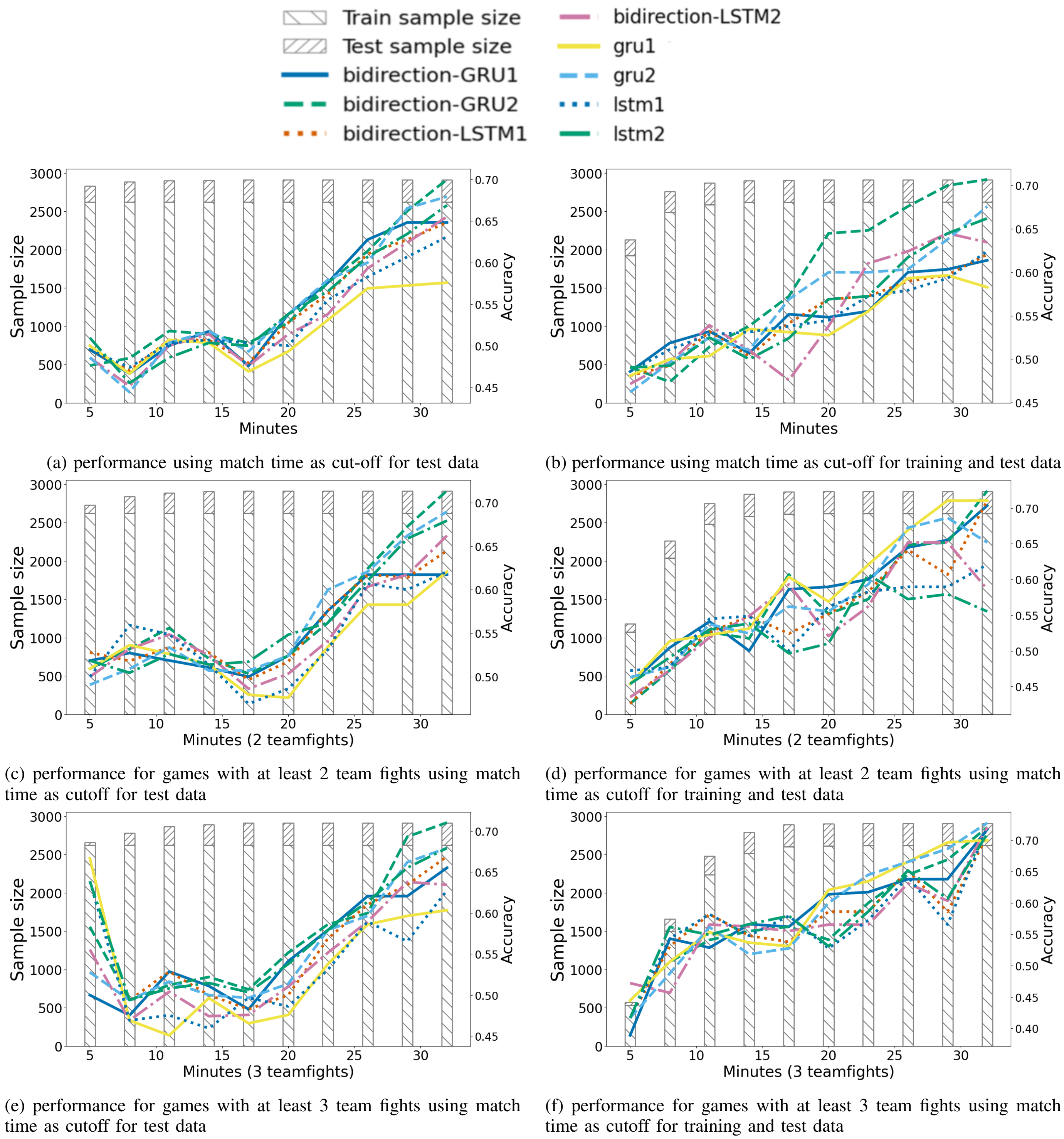
On a further note, due to the varied nature of DOTA 2 matches, restricting the model to only utilize games with up to a certain number of team fights would result in overfitting, due to the smaller sample size. Restricting the model to only using the features within a certain number of fights would also result in lower accuracy. This implies that it does not matter how many team fights a match contains, what matters are the features within the fights, the match time at which they are fought, and the order they are in.

A limitation of our research is the exclusive use of aggregated data. By aggregating all team fight performance data to the faction and team fight level, we were able to ensure that our models were trained efficiently. However, a more granular approach to the modeling, by focusing on player level performance could potentially increase the overall accuracy of our model [6], [10], [12], [13].

Previous prediction models for esports, to the best knowledge of the authors, did not integrate team fights as a factor. As shown here, team fights alone provide a signal for match prediction, and therefore appear to be a contender for inclusion in match prediction models as a novel feature. Therefore, to improve match prediction models in esports analytics, a potential venue for future exploration could be the integration of both in-game player team fight performance with traditional performance statistics [6]. A third line of data that could be explored to enhance prediction systems are player physiological characteristics [31]. Another area that could be expanded upon is the identification of players/heroes with exceptional contributions within team fights (Most Valuable Player, MVP). This would involve leveraging player role identification and individual player performance to enhance our existing models [12]. Of course, the influence of team hero combinations could also be added to enhance the performance of our models [13].

## VII. Conclusion and Future Work

In this research, we identified and defined the core concept of team fights in DOTA 2 esports. Team fights occur when players from opposing teams encounter each other on the playfield, and are important elements of an esports match that can be decisive for the match outcome. We utilized data from team fights in the esports game *Dota 2* to explore the potential use of team fight information in real-time match prediction models in such multi-player online battle arena games. This involved defining a team fight detection algorithm to filter and aggregate faction level (i.e. each team) features within each team fight. We then utilized the resulting data to train eight different types of RNN models to predict overall match outcome. Our models were able to achieve an accuracy of over 70% when including all team fight data up to 32 minutes into a match. Model performances were over 50% when trained on the first 5 minutes of each match and at least 2 team fights.




(a) performance using match time as cut-off for test data

(b) performance using match time as cut-off for training and test data

(c) performance for games with at least 2 team fights using match time as cutoff for test data

(d) performance for games with at least 2 team fights using match time as cutoff for training and test data

(e) performance for games with at least 3 team fights using match time as cutoff for test data

(f) performance for games with at least 3 team fights using match time as cutoff for training and test data

Fig. 6: RNN model performance with restricted data by game time

These results indicate that team fights alone contain a signal useful for predicting match winners, and indicates that team fight data could potentially be used as a new source of features to improve match prediction models in esports. Nevertheless, our results were not as strong as that obtained in previous research [6]. The performance of our models could potentially be improved by incorporating further signals, as suggested by adding player roles in Demediuk et al. [12].

The deep learning models presented here can be utilized in real-time, allowing commentators to note the impact of each team fight on the overall match outcome. The team fight detection algorithm could also be extended to automate the in-game camera to focus on detecting team fight events. Given the nature of *Dota 2* and similar games, ensuring the in-game camera is always focused on events that are of interest to the audience is a key challenge in esports, where the in-game action can take place in multiple places of the map at the same time [7]. Esports commentators can easily tell whether a team fight happens, but it requires years of experiences to understand the impact of each team fight and convey it clearly to audiences [2]. Furthermore, there is no quantitative way for measuring the influence of each team fight over the whole match. Our models, by solely leveraging team fight information, allows any prediction updates in live matches to be a direct reflection of the last team fight. Hence, the focus on only team fight information allows the RNN model predictions to be intimately tied to game context. This makes any changes to the predicted overall match outcome easily interpretable by commentators and audiences alike.

As would be expected, model performance in a team fight-based prediction model increases as more match time elapses, consistent with previous work (e.g. [6]. By adding team fight data to prediction models, it is possible their performance and accuracy could be enhanced. Future studies could thus extend previous work, and the work presented here, by integrating team fights data with player level performance features [6], [13], [31], hero role identification [12], spellcasting and ability use, (currently not explored in the esports analytics literature, but could be treated as text tokens that can be utilized with Transformer models [32]) and potentially even physiological characteristics [31].

## REFERENCES


[1] M. Schubert, A. Drachen, and T. Mahlmann, "Esports analytics through encounter detection," in *MIT Sloan Sports Analytics Conference*. MIT Sloan, 2016.

[2] F. Block, V. Hodge, S. Hobson, N. Sephton, S. Devlin, M. F. Ursu, A. Drachen, and P. I. Cowling, "Narrative bytes: Data-driven content production in esports," in *Proceedings of the 2018 ACM international conference on interactive experiences for TV and online video*, 2018, pp. 29–41.

[3] C. Gough, "esports audience size worldwide from 2019 to 2024," https://www.statista.com/statistics/1109956/global-esports-audience/, 2021, accessed: 2021-05-29.

[4] J. Ahn, W. Collis, and S. E. Jenny, "The one billion dollar myth: Methods for sizing the massively undervalued esports revenue landscape," *International Journal of Esports*, vol. 1, no. 1, 2020.

[5] Y. N. Ravari, P. Spronck, R. Sifa, and A. Drachen, "Predicting victory in a hybrid online competitive game: The case of destiny," in *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, vol. 13, no. 1, 2017.

[6] V. J. Hodge, S. M. Devlin, N. J. Sephton, F. O. Block, P. I. Cowling, and A. Drachen, "Win prediction in multi-player esports: Live professional match prediction," *IEEE Transactions on Games*, 2019.

[7] A. V. Kokkinakis, S. Demediuk, I. Nölle, O. Olarewaju, S. Patra, J. Robertson, P. York, A. P. Pedrassoli Chitayat, A. Coates, D. Slawson *et al.*, "Dax: Data-driven audience experiences in esports," in *ACM International Conference on Interactive Media Experiences*, 2020, pp. 94–105.

[8] Cyborgmatt, "Dota 2 prize pool tracker," https://dota2.prizetrac.kr/, 2021, accessed: 2021-05-30.

[9] C. Eggert, M. Herrlich, J. Smeddinck, and R. Malaka, "Classification of player roles in the team-based multi-player game dota 2," in *International Conference on Entertainment Computing*. Springer, 2015, pp. 112–125.

[10] S. Demediuk, A. Kokkinakis, M. S. Patra, J. Robertson, B. Kirman, A. Coates, A. Chitayat, J. Hook, I. Nolle, O. Olarewaju *et al.*, "Performance index: A new way to compare players," in *2021 MIT Sloan Sports Analytics Conference*. IEEE, 2021.

[11] A. Katona, R. Spick, V. J. Hodge, S. Demediuk, F. Block, A. Drachen, and J. A. Walker, "Time to die: Death prediction in dota 2 using deep learning," in *2019 IEEE Conference on Games (CoG)*. IEEE, 2019, pp. 1–8.

[12] S. Demediuk, P. York, A. Drachen, J. A. Walker, and F. Block, "Role identification for accurate analysis in dota 2," in *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, vol. 15, no. 1, 2019, pp. 130–138.

[13] A. Semenov, P. Romov, S. Korolev, D. Yashkov, and K. Neklyudov, "Performance of machine learning algorithms in predicting game outcome from drafts in dota 2," in *International Conference on Analysis of Images, Social Networks and Texts*. Springer, 2016, pp. 26–37.

[14] R. Sifa, A. Drachen, F. Block, S. Moon, A. Dubhashi, H. Xiao, Z. Li, D. Klabjan, and S. Demediuk, "Archetypal analysis based anomaly detection for improved storytelling in multiplayer online battle arena games," in *2021 Australasian Computer Science Week Multiconference*, 2021, pp. 1–8.

[15] A. Drachen, M. Yancey, J. Maguire, D. Chu, I. Y. Wang, T. Mahlmann, M. Schubert, and D. Klabajan, "Skill-based differences in spatio-temporal team behaviour in defence of the ancients 2 (dota 2)," in *2014 IEEE Games Media Entertainment*. IEEE, 2014, pp. 1–8.

[16] P. Yang, B. E. Harrison, and D. L. Roberts, "Identifying patterns in combat that are predictive of success in moba games." in *FDG*, 2014.

[17] Y. Yang, T. Qin, and Y.-H. Lei, "Real-time esports match result prediction," *arXiv preprint arXiv:1701.03162*, 2016.

[18] OpenDota, "Opendota api," *https://docs.opendota.com/#*, 2021, accessed: 2021-01-01.

[19] M. Schrodt, "Clarity analyzer," *https://github.com/spheenik/clarity-analyzer*, 2015, accessed: 2021-01-01.

[20] D. . Wiki, "Disable," *https://dota2.fandom.com/wiki/Disable*, 2021, accessed: 2021-06-01.

[21] ——, "Attack modifier," https://dota2.fandom.com/wiki/Attack_modifier, 2021, accessed: 2021-06-01.

[22] E. Kleinman, S. Ahmad, Z. Teng, A. Bryant, T.-H. D. Nguyen, C. Harteveld, and M. Seif El-Nasr, "" and then they died": Using action sequences for data driven, context aware gameplay analysis," in *International Conference on the Foundations of Digital Games*, 2020, pp. 1–12.

[23] M. Rao, "System requirements," https://gurugamer.com/pc-console/dota-2-system-requirements-8946, 2020, accessed: 2021-06-01.

[24] J. Friedman, T. Hastie, R. Tibshirani *et al.*, *The elements of statistical learning*. Springer series in statistics New York, 2001, vol. 1, no. 10.

[25] A. C. Tamhane, *Predictive Analytics: Parametric Models for Regression and Classification Using R*. John Wiley & Sons, 2020.

[26] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, ser. Studies in computational intelligence. Berlin: Springer, 2012. [Online]. Available: https://cds.cern.ch/record/1503877

[27] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[28] R. Sifa, "Predicting player churn with echo state networks," in *2021 IEEE Conference on Games (CoG)*. IEEE, 2021, pp. 1–5.

[29] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[30] T. Iqbal and S. Qureshi, "The survey: Text generation models in deep learning," *Journal of King Saud University - Computer and Information Sciences*, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1319157820303360

[31] A. Smerdov, E. Burnaev, and A. Somov, "Ai-enabled prediction of esports player performance using the data from heterogeneous sensors," *arXiv preprint arXiv:2012.03491*, 2020.

[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.