Bachelor Project

Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Cybernetics

Predicting Counter-Strike Game Outcomes with Machine Learning

Ondřej Švec

Supervisor: Ing. Gustav Šír, Ph. D.
Field of study: Cybernetics and Robotics
January 2022

BACHELORȏS THESIS ASSIGNMENT

I. Personal and study details

Student's name:	**Ġvec Ondŝej**	Personal ID number:	**483800**
Faculty / Institute:	**Faculty of Electrical Engineering**		
Department / Institute:	**Department of Cybernetics**		
Study program:	**Cybernetics and Robotics**		

II. Bachelorȏs thesis details

Bachelorȏs thesis title in English:

Predicting Counter-Strike Game Outcomes with Machine Learning

Bachelorȏs thesis title in Czech:

Predikce vĨsledkŢ ve hŝe Counter-Strike s pomoc² strojov®ho uļen²

Guidelines:

The e-sports industry experiences an unprecedented growth in popularity and impact, however, predictive analytics studies in the domain are still considerably rare. Moreover, in contrast to classic sports matches, there are also rich data records collected from the virutal game enviroment, allowing for a range of machine learning
methods to be tested. The subject of this thesis is such an exploratory machine learning study into the most popular e-sport - CS:GO.

1) Review the state of predictive analytics in CS:GO.
2) Collect statistically significant amount of match data.
3) Perform exploratory data analysis and preprocessing.
4) Experiment with a range of sample representations and models.
5) Analyse and compare your results against selected baselines.

Bibliography / sources:

[1] Bjºrklund, Arvid, et al. Predicting the outcome of CS: GO games using machine learning. BS thesis. Chalmers University of Technology, 2018.
[2] Makarov, Ilya, et al. "Predicting winning team and probabilistic ratings in ñDota 2ò and ñCounter-Strike: Global Offensiveò video games." International Conference on Analysis of Images, Social Networks and Texts. Springer, Cham, 2017.
[3] Minka, Tom, Ryan Cleven, and Yordan Zaykov. "Trueskill 2: An improved bayesian skill rating system." Tech. Rep. (2018).
[4] Zhou, Jie, et al. "Graph neural networks: A review of methods and applications." AI Open 1 (2020): 57-81.

Name and workplace of bachelorȏs thesis supervisor:

Ing. Gustav Ġ²r, Ph.D., Department of Computer Science, FEE

Name and workplace of second bachelorȏs thesis supervisor or consultant:

Date of bachelorȏs thesis assignment: **20.05.2021** Deadline for bachelor thesis submission: **04.01.2022**

Assignment valid until: **19.02.2023**

Ing. Gustav Ġ²r, Ph.D. Supervisorȏs signature	prof. Ing. Tom§ġ Svoboda, Ph.D. Head of departmentȏs signature	prof. Mgr. Petr P§ta, Ph.D. Deanȏs signature

III. Assignment receipt

The student acknowledges that the bachelor's thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the bachelor's thesis, the author must state the names of consultants and include a list of references.

.

Date of assignment receipt

Student's signature

Acknowledgements

I would like to thank my supervisor Ing. Gustav Šír, Ph. D. greatly for leading me throughout the year.

Computational resources were supplied by the project "e-Infrastruktura CZ" (e-INFRA CZ LM2018140) supported by the Ministry of Education, Youth and Sports of the Czech Republic.

Declaration

I declare that the presented work was developed independently and that I have listed all sources of information used within it in accordance with the methodical instructions for observing the ethical principles in the preparation of university theses.

Prague, date 4. January 2022

Abstract

The surge of the esports industry, alongside the rise in popularity of machine learning, provides unique opportunities for improving currently used methods of match outcome predictions.

This thesis aims to give a new perspective to predicting outcomes of Counter-Strike matches. In order to achieve that, first, high volume dataset consisting of over 40,000 matches was obtained and analyzed. Three sample representations, two types of data preprocessing, six machine learning models and a model utilizing Elo rating were proposed, cross-validated and compared to a selected baseline. Model utilizing Elo rating proved to be consistently the best one, with test set accuracy of 64 %, closely followed by Random Forest model with test set accuracy of 63 %.

Keywords: machine learning, neural networks, Counter-Strike, esports, predicting outcomes

Supervisor: Ing. Gustav Šír, Ph. D.

Abstrakt

Vzestup odvětví esportů spolu s nárůstem popularity strojového učení poskytuje jedinečné příležitosti pro zlepšení aktuálně používaných metod předpovědí výsledků zápasů.

Tato práce si klade za cíl poskytnout nový pohled na predikci výsledků zápasů Counter-Strike. Aby toho bylo dosaženo, byl nejprve získán a analyzován velký objem dat skládající se z více než 40 000 zápasů. Byly navrženy tři reprezentace dat, dva typy preprocessingu dat, šest modelů strojového učení a model využívající hodnocení Elo, které byly křížově ověřeny a porovnány s vybraným výchozím stavem. Model využívající hodnocení Elo se prokázal být nejlepším, s přesností na testovací sadě 64 %, těsně následovaný modelem Random Forest s přesností na testovací sadě 63 %.

Klíčová slova: strojové učení, neuronové sítě, Counter-Strike, esports, předpověď výsledků

Překlad názvu: Predikce výsledků ve hře Counter-Strike s pomocí strojového učení

Contents

Figures

Tables

Chapter 1

Introduction

Sports have been around us for as long as we can remember, with the documented history going back thousands of years. What may have started as one of the ways to prepare for an incoming war or perhaps a hunt, became an important part of today's cultural society. Over the years, sports have developed from throwing spears and rocks in a numerous different ways for people to compare their skills [1]. Some of these sparking up a discussion about what exactly sport is, what it embodies and what it does not. For some, sporting is an activity involving physical exertion and competition, for others the latter suffices [2].

With the invention of the computer and the internet, possibly the latest addition to sports arose - esports. Often talked in the context of the fastest growing sports industry (placed as the best sport rated by potential to grow revenues globally by PwC in 2018 [3], 2019 [4] and 2020 [5]), esports' worldwide market revenues estimates rose from 130 million U.S. dollars to nearly 1.1 billion U.S. dollars, making the predictions for 2024 reach over 1.6 billion U.S. dollars [6]. [1] The grow is has caught interest of traditional sports' organizations looking to diverge. In the last couple of years, sports organisations' level of engagement with esports grew rapidly, evolving from scepticism to embracing [5].

Counter-Strike: Global Offensive (CS:GO for short) has been launched in 2012 and has since become the highest ranking game on Steam in terms of average players each month with approximately 549 thousand players [11].

---

[1] According to [7], the esports audience grew 9.5 % from 397.8 to 435.9 million of occasional viewers with 220.5 million of esports enthusiasts.

One of its biggest advantages is the simplicity of core game mechanics, unlike in many other cases of other esports titles, making it very easy to pick up both playing and watching the game. To this day, CS:GO holds a standard of how a competitive FPS game is supposed to be like.

Sports and gambling have always gone hand in hand. Ancient Greek artefacts indicate that there was betting already on the Olympic Games [8]. However, with the invention of online betting, sports gambling is currently experiencing a huge boom. What used to be a small market, now is a billion-dollar industry. Sports betting is recently more and more, in terms of revenue, being compared to traditional sports [9].

With computational power being more accessible every year, machine learning became very popular in nearly every specialization in our lives. Thanks to machine learning, areas, such as speak recognition or image recognition, were greatly accelerated. Every other cellphone has nowadays got a speech recognition function powered by a machine learning model, and each year a steps towards fully autonomous vehicles are taken. In relation to sports predicting and betting, other methods are still preferred to machine learning. However, some preliminary research has already been done in the area, with some positive results.

1.1 Problem and goals

Machine learning models showed promising results in recent studies, improving predictions of outcomes, both, in traditonal sports and esports. This thesis tries to come up with different sample representations and prediction models to predict outcomes of CS:GO matches.

In this study, we have chosen to tackle the problem directly as classification task, i.e. predicting the winner of the game as one of the competing teams. Proposed sample representations then consist of player representation consisting of player statistics calculated to round average, team representation consisting of roster's ability to win percentages of classified rounds, and representation combining the two representations.

The models selected to approach this task then consist of various neural networks, standard machine learning models and an Elo rating-based model.

Chapter 2

Background

In this chapter we provide the necessary background to the game of CS:GO and the various machine learning concepts that will be utilized later in the thesis.

2.1 CS:GO and betting

Intersection of esports and betting is not one to take lightly. Since its launch in July 2012 to July 2013, CS:GO had averaged up to 20 thousand players on a month-to-month basis [11]. In August 2013, the Arms Deal update came out, allowing for trading and, especially, betting of in-game items, which, in turn, skyrocketed the interest in CS:GO. A full year later, in August 2014, CS:GO averaged 133 thousand players. Adding one more year, the count rose up even higher, to 357 thousand average players [11]. CS:GO is not the only esports title to heavily rely on gambling. Nearly 59 % of the 2021 esports revenue streams, valued at 641 million $, was made of sponsorship [7]. Conventional industries are getting themselves involved in esports as sponsors from all around, though, one of the biggest supporting forces in the industry is the gambling industry [12]. Interestingly, not only the teams take up gambling sponsorships, the tournament organizers do, too [13].

2.2 State of predictive analytics in CS:GO

With machine learning's rise in popularity, attempts to implement it one way or another in connection with CS:GO have been made, however, not many in terms of predicting match outcomes. Some people try teaching bots play via unsupervised learning like human players, to either gain an advantage, or to substitute human players when missing [16].

In 2017, [17] tried predicting outcomes of matches by using features calculated from so-called after-plant situations. After-plant situations occur when the attacking side (T side) plants a bomb and the timer before explosion starts to countdown. Features are made of data from each second of the after-plant. The study harvests data from 162 map demos (demo contains all actions happening in the game) played from February 2016 to March 2017. It achieves a 62 % accuracy. In this thesis, rather than observing a small dataset and trying to predict based on very specific features, a huge dataset is obtained and more of a general approach, in terms of features chosen, is taken.

CS:GO allows to record statistics from any match played in the game. This means anyone can gather data by playing the game and saving demo files. And many third party sites also allow viewing detailed statistics of any player. In 2018, [18] analyzed 6,000 matches of 1,000 top players from matching site FaceIt.com[1]. The study's goal was to collect in-game statistics, group players into clusters and predict a winning team by the clusters present in each match's teams. It achieved accuracy of 65.11 %, which showed an improvement compared to predictions made solely by observing FaceIt's ranks. However, this method favors from the fact that FaceIt matching doesn't take clustering of players in consideration at all, meaning it's perfectly possible for a team to have players preferring the same role in the game. This seems to be hardly transferable to professional match predicting, as teams are not made up by an algorithm, but by players teaming up, themselves.

Interestingly enough, as opposed to the lack of studies linking CS:GO and machine learning predictions of outcomes, number of studies for MOBA[2] games exist.

---

[1] FaceIt is third-party service, where players queue up and get matched based on rank FaceIt calculates for each player based on their perfomance in previous matches.

[2] Multiplayer Online Battle Arena , a popular genre of esports games

2.3 Rules of CS:GO

Counter-Strike is a team-based multiplayer first person shooter. Two teams of five players each compete in multiple rounds with the goal of winning enough rounds to win a map. Most matches are played in either a best-of-1, best-of-3, or best-of-5 (often shortened to just BO1, BO3 and BO5) setting, meaning, that winning majority of a given count of maps results in a win.

Each map takes place in one of seven in-game levels[3] called Active Duty Map Pool. The order of maps in a match is set via a vote-ban system where teams first decide maps that they do not want to play by banning them and follow up by picking maps that are then played. Every level is substantially different from all the others.

A map starts by deciding which team plays which side first. The usual way to decide the starting sides is performing a single knife round in which both teams play with only knives as weapons. The winner then chooses the preferred starting side. Maps are in most cases played as a best-of-30. First fifteen rounds are played, then sides are changed and rest of the rounds is played out until one of the teams reaches a majority (sixteen) of rounds or the map comes to a tie (15-15 score[4])

To win a round, a team has to fulfill one of a few conditions depending on the side the team is playing in the round. Attacking side (often called Terrorist side or T side for short) wins a round by successfully planting and exploding a bomb in one of designated areas in a level. Defending side's (often called Counter-Terrorist side or CT side for short) goal is to prevent the bomb from exploding by either defusing an already planted bomb or delaying the attacking team enough for the round to close as time runs out. Both sides also win the round by eliminating all players of the other team before reaching their goal.

Economy plays a crucial role in Counter-Strike. A round starts with both teams locked in buy zones for the duration of buy time. Teams buy armor, utility and weapons in order to maximize the chances of winning the round. At the end of each round, both teams receive bonus money for next round. While the losing team receives a higher bonus, multiple member of the winning team usually survive the round, which results in keeping the bought equipment for the next round without having to rebuy it.

---

[3] In this thesis level and map is used interchangeably, as well, as map and match, depending on the context.

[4] In which case, an overtime is usually played.

2.4 Player ranking - Elo

Elo is a rating system developed by Arpad Elo and used since 1970. It was developed, and to this day used, for rating chess players [19]. Nowadays, Elo rating and its derivatives are used in countless of sports and competitions, though, usually tweaked.

At the beginning, all players start with Elo rating of 1500. After the match ends, resulting in one of the sides taking a win, Elo ratings are recalculated using following formula [20]:

$$E_A = \frac{1}{1 + 10^{(R_B - R_A)/400}};$$
$$R'_A = R_A + K \cdot (S_A - E_A);$$

where R_A stands for rating of side A before the match, R_B for rating of side B before the match, R_A for rating of side A after the match, S_A the actual outcome of the match from side A's perspective (1 for a win, 0 for a loss, 0.5 for a tie), E_A is the probability of side A winning the match and K a chosen scaling factor. In this study, R_A, R_B are calculated as means of theirs respective players. To update the Elo values, difference $R'_A - R_A$ is added to each of the players' ratings.

In this thesis, Elo rating serves a role of non-machine learning model, for comparison purposes.

2.5 Machine Learning

In this section, background is laid out to the used machine learning tools. First sub-section explains non-neural network models used in this thesis, second describes elemental layers of neural network models, that have been used, with the last two subsections explaining loss function and optimizers.

2.5.1 Non-neural network models

With the ever-growing expansion of machine learning, countless of very advanced models exist. For the lack of studies in CS:GO's outcome prediction

Figure 2.1: The logistic sigmoid function [22].

using machine learning, we apply Occam's razor, or, as often called, the principle of parsimony [21]. What is meant by that, is the fact, that rather than trying high-end machine learning models, simple machine learning are tested to see how they perform.

Logistic regression

One of the oldest classification models, Logistic regression, is named after the use of a sigmoid (logistic) function

$$\sigma(x; w) = \frac{1}{1 + e^{-x^T w}}, \tag{2.1}$$

where x stands for vector of input features and w stands for vector of weights. If we look at the image of function σ (figure 2.1, we can see that values are in range $(0, 1)$. This means that the probabilty of predicted class in a binary classification task can be defined as follows

$$P(y = 0 | x) = \sigma(x; w), \tag{2.2}$$

$$P(y = 1 | x) = 1 - \sigma(x; w), \tag{2.3}$$

where $y \in \{0, 1\}$ is predicted label.

Optimal weights w are found minimizing a selected loss function (more in section 2.5.3).

Figure 2.2: Binary decision tree functionality illustrated [26]

Random forest

A binary decision tree is a structure based on a subsequent decision process made by asking a series of questions. Starting from the root of the tree, a feature is evaluated and, depending on the outcome, one of the two branches is selected to proceed further. This procedure is repeated until a leaf of the tree is reached [23]. This leads to the binary decision tree being one of the most intuitive, as, instead of having to observe calculated weights, one can simply look at the questions asked.

To acquire such questions, first, a dataset distribution has to be measured and suitably divided. This can be achieved using Gini index, which is a summary statistic measuring the equability of dataset distribution. Considering the dataset is ideally split, the split receives a Gini score of 0. Otherwise, values range from 0 to 0.5, where 0.5 stands for the worst split[5]. After splitting the dataset in numerous ways and evaluating all splits by a chosen criterion function (in this case, Gini index), the best performing split is chosen as the node in the tree [25]. Starting from the root, all nodes are calculated until maximmum tree depth is achieved. Maximum tree depth is a hyperparameter stating number of nodes from the root node of the tree to the leaves. The higher value of tree depth, the higher chance of overfitting [25].

To obtain a higher performance, binary trees are used in ensembles, meaning, many trees are used and their predictions are averaged for the final output value, helping the overall stability of the classification. However, this can only be achieved, if each tree is created based on a different dataset, done by dividing the whole training set into smaller sets [26]. One of the algorithms for dividing training sets is called Bootstrap aggregating. It, first, generates m uniform training subsets with replacements, meaning, a fraction of the dataset consists of repeating values. This helps random forests tremendously,

---

[5] The calculating process of the Gini index can be seen at [24].

Figure 2.3: k-NN classifier illustration. For $k = 3$, l_p is red triangle, for $k = 5$, l_p is blue square [28].

as even a small change in the dataset can result in a very different decision tree [27].

k-Nearest Neighbors algorithm

Another model known for its simplicity and intuitivity. Suppose n rows with m continuous features exist. First, all of the rows are placed in $\mathbb{R}^m$ space. Let

$$C_i = (c_{i1}, \dots, c_{im}) \in \mathbb{R}^m, \; i \in \{1, \dots, n\} \tag{2.4}$$

be coordinates of i-th row in the training set. Then c_{ij}, $j \in \{1, \dots, m\}$ stands for j-th feature of the i-th row. Also, let

$$l_i \in \{0, 1\}, i \in \{1, \dots, n\} \tag{2.5}$$

be the label value for i-th element in the set.

Predicting label l_p of an input with feature values resulting in coordinates C_p starts with ordering all C_i coordinates by a selected norm:

$$||C_p - C_1|| \leq \dots \leq ||C_p - C_m||, \tag{2.6}$$

resulting in $(C_1, \dots, C_k)$ being $k \in \mathbb{N}$ nearest neighbors. Label value with the highest amount of appearances in $(l_1, \dots, l_k)$ becomes the predicted label l_p. The process is also illustrated in figure 2.3.

2.5.2 Neural network models

With the increase of computational power, neural networks recently gained a lot of popularity. Two types are especially important, as others are often built, at least partly, from them - fully connected neural nets and convolutional neural nets [29].

Fully connected layers

Fully connected layers are made of layers of neurons, often described in comparison to the neurons of a human brain. Each neuron in a layer n takes each value passed from previous layer $n\ \ 1$, $x_{n\ \ 1;i}$, multiplies it with the neurons learned weight w_i and adds bias b [29]:

$$y_{n;j}(x_{n\ \ 1}) = f(w_j\ \ x_{n\ \ 1} + b); \tag{2.7}$$

where $y_{n;j}$ stands for output of j-th neuron in n-th layer and $f(x)$ stands for non-linear activation layer. Often times, we write the equation 2.7 as matrix multiplication:

$$y_n(x_{n\ \ 1}) = f(W\ \ x_{n\ \ 1} + b);\ \ W = (w_1; : : : ; w_J)^T\ 2\ R^{I\ \ J}; \tag{2.8}$$

where I stands for number of neurons in layer $n\ \ 1$ and J for number of neurons in layer n. The situation is illustrated in figure 2.4.

Convolutional layers

Convolutional layers introduce sliding dot product kernel of weights. Now, only of partion of neurons from previous layer impacts neurons in the next layer, however, thanks to this, number of learnable weights is a lot smaller, making the network a lot more flexible, and also, much more observable, especially in case of classificating images [29].

Process of calculating the output values of a convolutional layer is illustrated in figure 2.5. Many parameters can be set, in respect to the kernel. Size determines number of weights in each kernel. Number of kernels is set by the output dimension of convolutional layer. Stride stands for the size of increment in each sliding of the kernel.