Project Report

On

Supervised Learning –

CSGO Round Winner Project

Table of Contents

Problem Statement

The objective of this project is to build and compare multiple machine learning algorithms for the classification of round winners in the game CS: GO.

Machine learning models showed promising results in recent studies, improving predictions of outcomes, both, in traditional 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. 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.

Project Objective

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.

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. 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. 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. 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.

Data Description

CS: GO is a tactical shooter, where two teams (CT and Terrorist) play for a best of 30 rounds, with each round being 1 minute and 55 seconds. There are 5 players on each team (10 in total) and the first team to reach 16 rounds wins the game. At the start, one team plays as CT and the other as Terrorist. After 15 rounds played, the teams swap side. There are 7 different maps a game can be played on. You win a round as a Terrorist by either planting the bomb and making sure it explodes, or by eliminating the other team. You win a round as CT by either eliminating the other team, or by disarming the bomb, should it have been planted.

CSGO round winner dataset contains 122410 records with 97 attributes. Implement all machine learning algorithms that you have learnt. Explore the data and apply feature selection and engineering. Check which machine learning models perform best on this dataset.

Sites providing API services are posing a very convenient way to obtain large chunks of data. For a considerable fee, data are given within hand's reach as there is no need to program any scraper to start working. There is a trade-off of getting only a fraction of features that can be obtained in other ways.

First of the two sites in table above that require scraping is called Wewatch. It provides complete match feed (rounds and kills in chronological order), meaning most of the features that are otherwise provided by other applications could be potentially computed after the scrape. Potentially new features could also be computed afterwards. Unfortunately, Wewatch only provides a very small dataset of matches.

Data Pre-processing Steps and Inspiration

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.

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 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.

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".

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.

Choosing the Algorithm for the Project

Choosing the same models as previously described in the Background chapter, we end up with three non-neural network machine learning models: -

- Logistic Regression
- Random Forest classifier
- k-Nearest Neighbors classifier

First in the list, the logistic regression, is implemented from module sklearn.linear_model. The second listed model, the Random Forest classifier is implemented from module sklearn.ensemble. Lastly, k-NN classifier is implemented from module sklearn.neighbors. For these models, calling a function fit once is enough for the process to be finished.
Machine learning models showed promising results in recent studies, improving predictions of outcomes, both, in traditional 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.

Motivation and Reasons for Choosing the Algorithm

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 way 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. Machine learning models showed promising results in recent studies, improving predictions of outcomes, both, in traditional 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.

Model Evaluation and Techniques

With Scikit's Random Forest Classificatory having native attribute of feature importance, that can be plotted, it makes sense to observe, what features managed to make a difference. Using the MDI method described in Scikit's documentation, we plot feature importance. Only features with mean decrease in impurity of value equal to 0.12 and higher are shown, to make the plot a bit cleaner. Unsurprisingly, elo averages are one of the features showcased, as elo consistently dominated all selected models throughout the thesis. Other than that, KD ratios and, also, separately kills and deaths show as important features, which makes sense, considering the nature of the game.

In pursue of this goal, first, a dataset of substantial volume had been scraped and stored in a database specifically designed to allow for future work and scalability, exceeding any other datasets freely available, both, in size, and features offered. The models selected to approach this task then consist of various neural networks, standard machine learning models and an Elo rating-based model. 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.

Inferences from the Same

Machine learning models showed promising results in recent studies, improving predictions of outcomes, both, in traditional 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. 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 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 way 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.

Future Possibilities of the Project

In pursue of this goal, first, a dataset of substantial volume had been scraped and stored in a database specifically designed to allow for future work and scalability, exceeding any other datasets freely available, both, in size, and features offered. 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 way 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. The models selected to approach this task then consist of various neural networks, standard machine learning models and an Elo rating-based model. 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.
Machine learning models showed promising results in recent studies, improving predictions of outcomes, both, in traditional sports and esports. This thesis tries to come up with different sample representations and prediction models to predict outcomes of CS: GO matches.

Conclusion

This thesis' goal was to perform an exploratory machine learning study regarding outcome prediction in professional matches of CS: GO.

In pursue of this goal, first, a dataset of substantial volume had been scraped and stored in a database specifically designed to allow for future work and scalability, exceeding any other datasets freely available, both, in size, and features offered. Three standard statistical and three neural network machine learning models were used to perform a series of testing. These machine learning models are directly compared to a model based on Elo ratings of players, and a baseline only predicting one side based on the observed data skewness. Accuracy of 64,0 % has been achieved with the best performing model based on the Elo rating of players, followed by 63,0 % accuracy with the random forest model and 59.8 % accuracy with the convolutional neural network, all beating the baseline model with 55.0 % accuracy. Compared to the 62.0 % accuracy reported in a previous work [17] on a significantly smaller dataset, two of our models managed to achieve better predictive accuracy.

Although beating the selected baseline, the Elo rating being the highest-ranking model signalizes a lot of room for improvement. Seeing Elo perform so well, a question arises whether some better rating-based models, such as TrueSkill 2 or Gecko ratings, could perform even better.

Moreover, the predictions might also profit from other sample representations based on features like tournament phase, vote-ban phase, map number in a series, previous head-to-head team performances, and such, providing further context to each match.