Statement of Work

CHESS GAME OUTCOME PREDICTION

AI ALGORITHMS - AIDI 1002 – 02

Facilitator – Marcos Bittencourt

Submission By

Dakshvir Singh Rehill - 100799805

Table of Contents

Problem Statement:

Create a prediction system that uses initial chess moves of a chess game to predict whether black player or white player will win the game.

Rationale Statement:

Chess is one of the most popular classic games of all time. It involves strategy and wit to beat the opponent. Due to the volatile nature of the game, it is not possible to look at the state of the board at any given time to determine who is going to win. Knowing the outcome of a game before the game ends helps to improve efficiency and strategy of the game. Due to this reason, the project plans to create a prediction system for game win or loss condition, using machine learning algorithms.

Data Requirements:

The problem statement mentions that we are trying to predict chess game's win or loss condition using a trained model. To perform this task, we need the following core data values for our training, testing, validation datasets:-

1. **Chess Moves –** We need a dataset that can provide us with detailed chess moves of a game in a machine readable format. The standard format for this type is the PGN format.
2. **Game Type –** Chess can be played in various ways. It is important to know the type of the game to properly predict the outcome.
3. **Game Rules –** The complex game has different sets of rules. Before prediction, we need to understand the rules.
4. **Game Result –** To properly train, test and validate our model, we need to know the outcome of the game so that we can verify the accuracy of our model.
5. **Player Rating –** FIDE ratings of the player can be used to provide weightage to the data to make the prediction algorithm more accurate.

Data Sources:

Chess.com has a huge repository of data on various games that have been played on the platform. It also provides a handy API to collect and gather the data. A fellow data researcher has already compiled a big dataset from Chess.com and this dataset is readily available for public use via Kaggle. We plan to use two datasets and both these datasets are uploaded and available on Kaggle by the same user. Here are the datasets explained:-

Chess Games of Woman Grandmasters (2009 - 2020)

This dataset comprises of games played by Woman Grandmasters of Chess on Chess.com. The dataset was graciously provided on Kaggle https://www.kaggle.com/rohanrao/chess-games-of-woman-grandmasters . The moves of each game is available in the PGN format. The dataset also has metadata information about the game, players, results and ratings. This is a complete dataset which will allow for a diverse prediction algorithm. The data source has a csv file comprising of the 15 features, namely, **game_id, game_url, pgn, time_control, end_time, rated, time_class, rules, wgm_username, white_username, white_rating, white_result, black_username, black_rating, black_result.** Out of these, the **pgn, time_class, rules, white_result, black_result, white_rating, black_rating** features look useful for satisfying our data requirements at first glance. Further exploratory data analysis will determine how and what will be used from the available features.

Chess FIDE Ratings (2016 - 2020)

This dataset comprises of all FIDE ratings of all players between the year 2016 and 2020. These ratings are official International Chess Federation ratings and follow the Elo Rating System which is the globally used as the primary rating system. The dataset was provided on Kaggle by the same user https://www.kaggle.com/rohanrao/chess-fide-ratings . The dataset comprises of yearly csv files depicting the ratings of each player and a player csv depicting personal information about the player. At first glance, the rating of the player in different game modes will be used to satisfy our data requirements. Further analysis might determine the usefulness of this data source.

Data Source Assumptions

Since our main data source comprises of games played by Woman Grand Masters, it doesn't represent the entire player pool. This may cause irregularities in prediction. An assumption that, both men and women would approach the game of chess in a similar way, has been made. We have data of 275,453 games in our main dataset. It is assumed that the data is almost evenly distributed among the different game types so that each type can be equally assessed while making the prediction. More clarity will be formed after Exploratory Data Analysis.

Model(s)/Architecture Approach:

The problem statement requires a way to analyze chess boards after various initial moves and predict the outcome of the game. After some research, the following approaches were devised to solve the aligned problem statement:

Support Vector Machines

Given that this prediction problem is a simple classification problem of Win/Loss/Draw, we can use classification algorithms like Support Vector Machines. There isn't much overlap of the above mentioned classes which is why it would be easy to get the decision boundary (hyperplane) between the classes. SVM models have good results on non-linear data and chess moves are non- linear.

Decision Tree

Decision Trees are simplest classification solutions but can easily get quite big or easily over fit according to the training data. Since computer chess players are made up of MiniMax trees, decision tree was chosen as an approach for this problem. This classifier would allow us to visualize the validity of our solution as well.

Random Forest

Random Forest solves the problem of over fitting but are memory intensive. Since our prediction problem can be solved using classification, Random Forest is a really good option. Random Forest randomizes the result by using multiple decision trees which increases the memory footprint of the model but increases the accuracy of the result. We will decide which model to use after final cross validation.

Regression Algorithms

Since the classes are very less and chess moves can easily be represented in the form of ndarrays, it is possible to implement different regression algorithms like Multinomial Naïve Bayes, Linear Regression, Logistic Regression, etc. These regression algorithms will also be tried to ensure more models are compared before deciding on the final model in the approach.

Since each of these models will provide different accuracies and results, we will use various cross validation methods to decide on which model is the best for our problem.

Testing and Cross Validation:

The chosen data source has a good amount of data values which allows for a better Train, Test split. We plan to use a 70% training and 30% testing split for testing our models individually. The 30% test data will be further split into testing and validation sets. The percentage of that split is planned to be 60% test and 40% validation. This added test of accuracy is added to ensure the quality of the model.

Different modeling approaches will be undertaken to ensure better accuracy. These modeling approaches will be compared to each other using various scores and approaches. The following are a few comparison scores:

1. **Accuracy Score –** Accuracy Score is a convenience function provided in ScikitLearn library. This function uses the real class and compares it to the predicted class. The final score is a 0-1 value of how accurate the result was. The split test and validation data will be subjected to this test and the best performing model will be determined.
2. **F1 Score –** F1 Score is the weighted average of recall and precision. Precision is the fraction of positive identifications that were actually true over total positive identifications. Recall is the fraction of actual positives over actual positive and false negative identifications. The F1 Score provides a 0-1 value that shows the balanced accuracy of the result.
3. **Confusion Matrix –** This is a way to calculate the number of predictions that were true positives, true negatives, false positives or false negatives. This will help us determine how many slip ups does our model do when it comes to predicting the outcome.

Each of these cross validation techniques will be analyzed before coming to a conclusion of which model will be used as the final learning model.

Conclusion:

The above statement of work clearly defines the problem statement, data requirements and possible approaches to solve the business problem. Next steps include performing a thorough statistical and exploratory data analysis of the data sources to find key features, use the results to preprocess and clean data and finally train models and test the results.