Prediction of outcome for two team games

1. Introduction

The speculation of games outcome has become a major part of game itself these days. Generally the past history of several games between players is used to rate players and predict future games. Though there are several rating systems, there is no definitive way of predicting the outcome of a given game involving two rated players/teams. This problem is a challenge in itself due to the scarcity of features which are general limited to only the players of a given game and the outcome. Inspite of huge historical data for a given game, the data for a given player is often very small making the prediction more challenging.

Note that any inferences or suggestions made in this study is based on historic data for games of Chess, but could be extended to other games with modifications.

2. Motivation

Most of the general rating models tend to predict passively due to the limitations mentioned. In other words, it is often the case that these models predict equal chances for both the players in a given future game. In addition to that, since games are generally organized between two equally skilled players, this problem becomes more evident. On the other hand, even when two players with notably distinct skills are playing a game, there is a unimaginably high uncertainty about the chance of weaker player winning. All the above factors would lead to high outcome prediction errors.

3. Data

We chose a training dataset with results of 65000 distinct games in Chess world between 8631 top rated players. The data is spanned over 12 years comprising games from 100 months. Test data comprises of distinct games over a period of 5 months. Few metrics that describe the nature of the data:

- Average number of games per player is 15 and standard deviation is 27.
- Maximum number of games between any pair of players is 15.
- Pairs of players that have never played a game in training data is approximately 99%. Though this seems surprisingly high, It is common scenario in games like Chess, where rated players play only a limited number of games.
- When we consider this as a graph with edges representing games between players, the average length between two players is 145 and max length between two players is 1200. These metrics are important for static/global models that try to obtain ratings using the connectivity factors.

On a different note, we observed that it is hard to cluster the data in to any well defined (concentrated) clusters representing wins, losses and draws using features from any of the below described approaches.

4. Error

We use root mean squared error as metric to evaluate the predictions since it incorporates both bias and variance of the estimator. Error is calculated as difference between each players actual wins and predicted chances of winning for every month. Root mean square error over predicted chances can clearly depict the minute changes obtained by each learning algorithm.

5. Model selection

5.1 BTL model

We initially started off with a pairwise comparison model, Bradley-Terry-Luce model [1] to generate partial rankings based on the pairwise results from each game. Once rating is obtained, the probability of a player winning is given as a function of ratings as follows:

P(player i beats player j at time t) =

$$\frac{\Pi i(t)}{(\Pi i(t) + \Pi j(t))}$$

where $\Pi_i(t)$ is rating of player i.

This model requires $O(n^2)$ number of training games for n players to give good skill rating. For the same reason when this model was tried over the mentioned dataset, the predicted chances for players were close to 0.5 most of the time.

5.2 Naive Bayes model

We tried using Naive Bayes algorithm to predict the outcomes of games. As expected the RMSE was high due to scarcity of per player-pair training data. For reference the equation is as follows:

P(player i winning given i and j are playing) =

$$\frac{P((i,j)|(i\,won))^* P(i\,winning)}{(P((i,j)|(i\,won))^* P(i\,winning) + P((i,j)|(i\,losing))^* P(i\,losing))}$$

5.3 Glicko rating system

Under the umbrella of incremental rating systems, we started off with Glicko rating system [2], which was one of the first systems to consider uncertainty in ratings given to each player (rating deviation). A high deviation indicates that a player may not be competing frequently, hence the players ratings are less reliable and viceversa.

5.4 True Skill

Using True Skill [3], one of the well known incremental rating systems, we first rated the players. After rating the players, we applied the same prediction equation from BTL model over these ratings to predict for test data. There was noticeable decrease in RMSE. This change can be attributed to the rating system of TrueSkill which allows scope for uncertainty through deviation factor for each player. We used the following equation to calculate skill ratings for each player from the mean and variance of their distribution:

$$\text{Skill} = \mu - 3\sigma$$

where μ is mean and σ is degree of uncertainty

5.5 Using features from True Skill

True skill has provided more defining features (ie mean and deviation) describing the skills of each player. We tried to use these features and find correlation between them to winning or loosing a game. The intuition behind applying different models over True Skill based features attributes mainly to the following observed nature of data: When the skill difference is over a certain threshold, the outcomes were more consistent with predictions. In this direction we made several attempts described below.

5.5.1 SVM over True Skill features

We used SVM to find a margin between the data using radial basis kernel and also linear kernel. SVM failed to find a satisfactory margin with fewer support vectors. To be more specific the number of support vectors were approximately 30000 for a training data of 65000. This once again iterates the problem of high level of uncertainty.

5.5.2 Logistic regression over True Skill features

We used logistic regression to fit the True Skill based features with the outcomes of each game in the training data. The parameters obtained were used to predict the outcomes for a given pair of players (represented by their mean and deviation). This model had the minimum RMSE when compared to any of the above mentioned approaches.

5.5.3 Logistic regression over True Skill through time

Normal True Skill has drawback of not being able to propagate information back in time. For example, if player 1 beats player 2, but in future player 2 beats a much stronger player 3, it does not adjust the previously given values to player 1. This is taken care of by True Skill through Time [4], which updates over a game multiple times by removing the affect of old games and adding the affect of new game. After obtaining mean and variance for each player using this approach, we ran logistic regression using the same features described in section **5.5.2**. The RMSE has dropped noticeably to 0.662 and possibly explains fact that even in real world, the latest game outcomes matter much more than the older outcomes.

5.6 Miscellaneous

We have noticed slight improvements by various other modifications specific to the game. For example, giving slightly more chance for white player improves the predictions and Chess literature actually claims that when two equally rated players play a game, white player has more than 50% chances of winning. To mention another one, approaches like True Skill over time implicitly give lesser importance to older games, but when we explicitly removed the very older game outcomes, the predictions improved (may be data dependent).

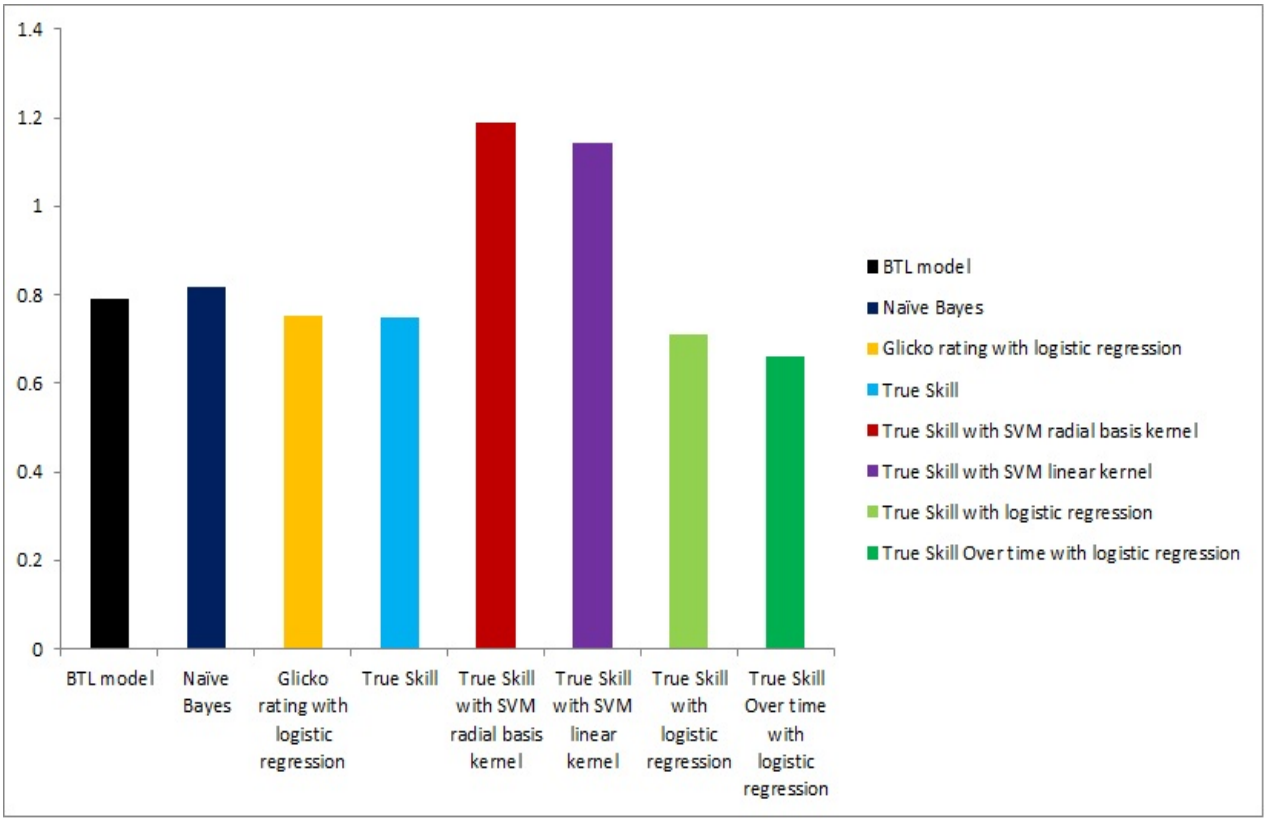
6. Final notes

We would like to finally note that, for a problem like game outcome prediction, something like True Skill would provide most basic set of features, over which prediction models can be built. Also additional features based on the nature of specific games should be considered. For example, home grounds can be advantageous for some games and for a game like tennis, court type would be an interesting feature.

On a slightly pedantic note, we would like to attribute this hardness of game outcome predictions to the ever lasting popularity and interest in games.

7. Results

Model	**RMSE**
BTL model	0.789
Naïve Bayes	0.817
Glicko rating system	0.754
True Skill	0.749
True Skill with SVM radial basis kernel	1.189
True Skill with SVM linear kernel	1.142
True Skill with logistic regression	0.709
True Skill over time with logistic regression	0.662



8. Acknowledgements

We would like to thank Kaggle [5] for making Chess datasets available publicly.

9. References

[1] Bradley, R.A. and Terry, M.E. (1952). *Rank analysis of incomplete block designs, I. the method of paired comparisons.* Biometrika, 39, 324–345

[2] Glickman, Mark E., and Jones, Albyn C. (1999). *Rating the chess rating system ,* Chance, 12, 2, 21-28

[3] Ralf Herbrich, Tom Minka, and Thore Graepel. *TrueSkill(TM): A Bayesian Skill Rating System, in Advances in Neural Information Processing Systems 20*, MIT Press, January 2007

[4] Pierre Dangauthier, Ralf Herbrich, Tom Minka, and Thore Graepel. *TrueSkill Through Time: Revisiting the History of Chess, in Advances in Neural Information Processing Systems 20*, MIT Press, 2008

[5] http://kaggle.com