

Regression networks for robust win-rates predictions of AI gaming bots

Ling Cen
EBTIC, Khalifa University,
UAE
cen.ling@kustar.ac.ae

Andrzej Ruta
ING Bank Slaski,
Katowice, Poland
andrzej.ruta@ingbank.pl

Dymitr Ruta
EBTIC, Khalifa University,
UAE
dymitr.ruta@kustar.ac.ae

Quang Hieu Vu
Zalora,
Singapore
quanghieu.vu@zalora.com

Abstract—Designing a robust and adaptable Artificial Intelligence (AI) opponent in a computer game would ensure the game continues to challenge, immerse and excite the players at any stage. The outcomes of card based games such as "Heartstone: Heroes of Warcraft", aside the player skills, heavily depend on the initial composition of player card decks. To evaluate this impact we have developed a new robust regression network in a context of the AAIA Data Mining Competition 2018, which tries to predict the average win-rates of the specific combinations of bot-player and card decks. Our network is composed of 2 levels: the entry level with an array of finely optimized state of the art regression models including Extreme Learning Machines (ELM), Extreme Gradient Boosted decision tree (XGBOOST), and Least Absolute Shrinkage and Selection Operator (LASSO) regression trained via supervised learning on the labeled training dataset; and just a single ELM at the 2nd level installed to learn to correct the predictions from the 1st level. The final solution received the root of the mean squared error (RMSE) of just 5.65% and scored the 2nd place in AAIA'2018 competition. This paper also presents two other runner-up models receiving RMSE of 5.7% and 5.86%, scoring the 4th and the 6th place respectively.

I. INTRODUCTION

Computer games, or more precisely computer-controlled games where players interact with objects displayed on computer screens, provide entertainment [1] and challenge players' physical and mental abilities. Beside entertainment, playing computer games has been found to combat stress, promote health and keep brain fit and active [2]. In recent years, fast development and penetration of Internet, multi-medial graphic devices, emergence of virtual reality, on-line open games led to the rapid growth of gaming popularity and combined with improved affordability, accessibility, ease and customization of gameplay, opponents choices, have driven the game industry to the enormous success and a bright future ahead [2].

To keep players interested and enthralled, computer games usually offer various stages and complexity levels to suit people from beginners to masters, and keeping them equally entertained for as long as possible. The fun of computer games is magnified when players play against their friends or other opponents from all over the world in on-line games since human opponents guarantee fresh, distinctive and engaging challenge [2]. With the recent advancement in Machine Learning (ML) and the Internet of Things (IoT), Artificial Intelligence (AI) has attracted increasing attention and heavily penetrated many industries including gaming industry. In many computer

games, designing a robust and adaptable AI opponent would ensure the games continues to challenge, immerse and excite the players at any stage, which is one of the most important aspects of success.

In the card based games such as Heartstone: Heros of Warcraft, aside the player skills, the outcomes heavily depend on the initial composition of card decks. To evaluate this impact, 2018 Advances in Artificial Intelligence and Applications (AAIA) Data Mining Competition was proposed and focused on the prediction of win-rates of 4 AI bot players, playing the Heartstone game among each other with different initial decks of cards and hero characters. The objective of the competition was to use these data to build the prediction model capable of accurately estimating win-rates of the same 4 AI bots but playing with one of the 200 new test card decks, gameplay of which and their results were not available to the contestants.

This paper presents a new robust shallow regression network to predict the average win-rates of the specific combinations of bot-player and card decks in a response to the context of AAIA Data Mining Competition 2018. Our network is composed of two levels. The first level is built with an array of individually trained regression models that have proven to be effective for sparse binary regression problems, including Extreme Learning Machine (ELM), Extreme Gradient Boosted Decision Tree (XGBOOST) and the Least Absolute Shrinkage and Selection Operator (LASSO) regression models, while the second level contains only a single ELM that learns to correct the predictions from the preceding level. The final solution submitted as a competitive entry in the AAIA'2018 Data Mining Competition received the RMSE of 5.65% and scored the 2nd place, marginally trailing the winning solution.

The remainder of the paper is organized as follows. AAIA Data Mining Competition 2018 is introduced in Section II. The feature extraction method and regression network for predicting the average win-rates of the specific combinations of bot-player and card decks are presented in Sections III and IV, respectively. The experimental results obtained through model evaluation are summarized in Section V, followed with a discussion in VI and the concluding remarks provided in Section VII.

II. COMPETITION DESCRIPTION

The AAIA Data Mining Competition 2018 is related to the turn-based card game of "Heartstone: Heros of Warcraft". In this game, two players choose their heroes with a unique power and compose a deck of thirty cards that represent various spells, weapons, and minions, and can be summoned in order to attack the opponent with the goal of reducing the opponent's health to zero and win the game. The outcomes of the game, aside the player skills, heavily depend on the initial composition of player card decks. To evaluate this impact, the competitors were expected to predict win-rates of four AI bot players, automatically playing many games against each other with different initial decks of cards and hero characters.

The training data provided by the competition contained a collection of JSON files describing in detail more than 300k games played by all pairs from the set of 4 different bots, each starting with one of 400 unique Hearthstone card decks. The data included the initial composition of card decks, heroes selected, the results of each game, and detailed turn-by-turn gameplay states and related statistics. The objective of the competition was to utilize these datasets to build the prediction model capable of accurately predicting win-rates of the 4 AI bots assigned to any previously unseen composition of card decks and related class of hero character. To evaluate the competitive models the win rates of all 4 bots were tested in combinations with specific 200 new test decks, however this time provided without any gameplay nor game results details to the contestants to properly simulate realistic predictive power of competing win-rates prediction models.

The solutions were evaluated using the root of the mean squared error (RMSE) measure. The preliminary score of each submitted solution was evaluated externally on a fixed 10% subset of the full test records and published on the competition leaderboard. The final evaluation on the complete testing set was performed after the completion, i.e. when the competitors submitted their final solutions with no further changes allowed.

III. FEATURE ENGINEERING

Estimation of average win-rates of the specific combination of bot-player and card decks can be solved via regression analysis that is a methodology for estimating the relationships between a dependent variable (response) and one or multiple independent variables (predictors). The dependent variable here was the win-rate expressed as a continuous real number from the $[0, 1]$ interval.

From the outset it has been decided, that since no gameplay details, beyond the initial deck, was available in the test stage, the training data need to be trimmed consistently down to the same content. It included the id of the player-bot and the initial Heartstone deck composition, i.e. the id of one of the 9 distinct hero characters and the cardinalities (0,1, or 2) of other cards from the pool of over 300 available card types. All above were cascaded to form a feature vector as shown in Fig. 1.

The initial modeling tasks involved generating features from the available data and after a brief experimentation with simple

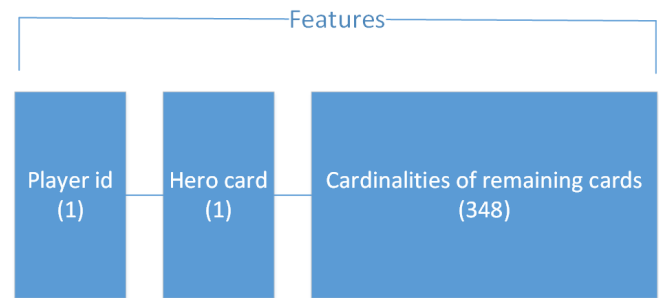


Figure 1. Feature representation.

linear regression models, the highest predictive power associated with the win-rate predictions appeared to come from numerical encoding of raw categorical features. The player id took the values of $[1,2,3,4]$ representing the 4 bot-players, and the hero card took the values of $[1,2,\dots,9]$ representing the 9 hero characters. The remaining card features took the values of $[0,1,2]$ depending on the cardinality of specific card types in the decks. For each data record associated with a single game, this formed a sparse 348-dimensional vector describing the cardinalities of card types appearing both in the training and test sets. The final feature set included $1 + 1 + 348 = 350$ features as shown in Fig. 1.

Initial feature selection experiments did not result in any improvement of the cross-validated performance measure, although in-sample (training-set) RMSE was reduced significantly after selection of around 100 greedily found card features. To prevent model overfitting, it was decided to include all 350 features in the model building phase. With these features, a robust regression network has been developed for predicting win-rates of four AI bots playing the "Heartstone: Heros of Warcraft" game against each other with different initial decks of cards and hero characters, which will be elaborated further in the following section.

IV. REGRESSION NETWORKS

Artificial neural networks (ANNs) have been successfully applied in various fields due to their ability to approximate complex nonlinear mappings directly from input samples as well as model natural and artificial phenomena that are difficult to express using classical parametric techniques. Gradient-based learning algorithms are commonly used to train neural networks and tune the parameters iteratively, which, however, requires long training time.

To improve learning efficiency of neural networks, Huang and his colleagues proposed extreme learning machines (ELMs) that are feed-forward neural networks with a single or multiple layers of hidden nodes. Instead of tuning the parameters of hidden nodes, the ELMs randomly choose hidden nodes and analytically determine the output weights of the network [6]. In Comparison to many state-of-the-art computational intelligence methods, such as the conventional back-propagation (BP) algorithm and Support Vector Machines

(SVM), ELMs have the advantage of much faster learning rate, ease of implementation, the least human intervention, and better generalization performance in terms of lower training error and smaller norm of weights. It has been reported by Huang et al. based on their experimental results that ELMs are able to achieve better generalization performance and learn thousands of times faster than traditional learning algorithms for feed-forward neural networks [6].

In order to extend the generalization performance of the ELM, a novel shallow regression network composed of 2 stages has been developed. In the first stage an array of finely optimized state-of-the-art regression models are trained directly on the input data to predict the desired regression outputs. The models shortlisted for this stage based on best preliminary ad-hoc evaluation included beside kernelized ELMs, XGBOOST, LASSO, SVM, Gaussian process (GP) and simple Multi-Layer Perceptron (MLP) models.

The outputs of all base models, i.e. the proposed regression outputs are passed on to the second and final stage of the shallow network in which just a single or multiple regression are trained again, however this time their inputs are multiple propositions of the predicted outputs, hence their role is just to learn to optimally correct multiple predictions to minimize final regression error. The decision to limit such corrective layers to just a single 2nd layer follows from extensive experimentations which confirmed that adding more corrective layers does not improve the performance but only contributes to the network complexity.

We have dedicated a lot of experimentation to the selection of the best subset of primary regressors as well as the final stage corrective models. We have, however consistently received ELM to be the single most effective 2nd stage corrective regressor, while also in the primary first layer ELM appeared to dominate in terms of performance but showed the best overall results if combined in the first layer with XGBOOST and LASSO regression models only.

A structure of the best performing network with 9 base kernelized ELMs, 1 LASSO and 1 XGBOOST models in the primary layer and a single ELM in the final layer is shown in Fig. 2.

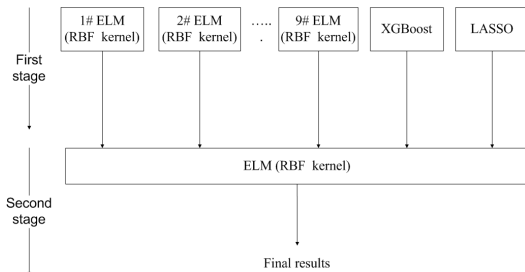


Figure 2. A sample structure of the learning model.

Multiple ELM models with radial-basis-kernels of increasing width parameter (gamma) from 20 to 60 dominated the

first layer of the network. The RBF kernel is defined as [7]

$$K_{\text{RBF}}(x, x') = \exp\left(-\frac{\|x - x'\|^2}{2\sigma^2}\right), \quad (1)$$

where $\gamma = 2\sigma^2$.

As mentioned above, these 9 ELM models in the optimized network setup have been complemented with just a single XGBOOST and LASSO models, therefore for completeness few details on only the added models are provided below.

- A decision tree builds a regression model in the form of a tree structure, which breaks down a dataset into multiple smaller subsets and incrementally builds a tree with decision nodes and leaf nodes for the purpose of classification or regression. XGBOOST, based on Extreme Gradient Boosting model [3], is an implementation of the gradient boosted decision trees algorithm with a goal of pushing the limit of compute resources for boosted tree algorithms [4]. In recent years, XGBOOST, due to its advantages of fast processing speed and high prediction accuracy, has been employed by many winning teams of a number of machine learning competitions, e.g. [5].
- LASSO regression is a shrinkage and variable selection technique aimed at enhancing the prediction accuracy and interpretability of the linear regression model it produces [8], [9], [10]. It attempts to find a subset of predictors that minimize the prediction error of the response variable, which is achieved by imposing a constraint on model parameters to make regression coefficients for some predictor variables shrink down to 0. Given the feature vectors encoding cardinalities of cards are very sparse, LASSO is employed as another base regression model in the first stage of our network. It attenuates and effectively excludes certain variables from the model, while the variables with non-zero coefficients are considered as strongly associated with the target variable.

Among other primary models that deserve some attention despite not being selected to the final network was a Multi-Layer Perceptron (MLP) with variable number of neurons. Among a wide range of configurations trialed we found a network with 50 input neurons, one hidden layer of size 20, and a single linear-activation output neuron to be the best performing model of this kind. Rectified Linear Unit (ReLU) activation [11] was set for all input- and hidden-layer neurons. It should be noted however, that we managed to maximize the generalization performance of this network only after introduction of recently popular regularization techniques: batch normalization and dropout [12] after the first two dense layers. We decided to use this particular model as a benchmark model for our regression network, yet did not include it in the network itself.

Each of the base regression models in the first stage was individually trained over the whole training set. The second stage was built on top of the first stage with a goal of learning to correct its predictions. Experimentations concluded very decisively that just a single ELM with optimized hyper-parameters is best at learning to correct the primary regressors'

outputs and hence to further improve the generalization ability of the whole network. As a result, the entire regression network became a hybrid model with a decision level fusion in the top layer realized using the ELMs. It was very important, however, for the robustness of the emerging 2-level regression network to train the second layer on the cross-validated outputs of the first layer such that the second layer regression used only out-of-sample rather than in-sample prediction outputs.

V. EXPERIMENT RESULTS

As already partly explained in the previous section, many experimental trials were performed to determine the best composition of the first and the second stages of the regression network as well as optimize all the individual and joint hyperparameters. All the experiments were based upon both k-fold cross-validation over the training dataset and the external feedback in a form of performance scores published in the web-based KnowledgePit platform and calculated for only 10% of the test examples. Eventually, the best structure of the network consists of 9 kernelized ELMs, an XGBOOST, and a LASSO regression models in the first level that are connected to another ELM model in the 2nd level, is shown schematically in Fig. 2.

The parameters of the individual regression models were optimized over the k-fold cross-validated training set using Bayesian or grid optimization. The optimal network setup included 9 ELM models with radial-basis-function kernels of width [20,25,30,35,40,45,50,55,60], XGBOOST model with learning rate 0.01, re-sampling rate 0.2, maximum tree depth 2 and 100000 iterations, and the LASSO regression model with 100 lambdas and up to 100 non-zero weights. The ELM in the second stage used RBF kernel with a small width $\gamma < 1$.

The final solution that we submitted to the competition received the RMSE of 5.0% based on the preliminary evaluation on the 10% of all test examples, and the final score of 5.65% on the whole test set. The best RMSE scores on the preliminary leaderboard evaluation achieved individually using each base regression model were 5.88% for ELM with 40-wide RBF kernel, 6.64% for XGBOOST, and 6.87% for LASSO. For comparison, our benchmark single-stage MLP regression model achieved RMSE of 5.69% on the same 10% subset of the test set and 5.86% on the whole test set (6th best score), showing robustness to over-fitting yet still remaining slightly behind the proposed two-level regression network.

The above figures prove that the introduction of the shallow hierarchy with just a single regressor in the 2nd level was an adequate choice leading to a noticeable performance improvement compared to the base models.

VI. DISCUSSION

It is found that better individual performers of base models may not lead to better combined output. Indeed the removal of GP and SVM regressors, although individually top in-sample performers, surprisingly led to improved performance of the whole network.

To further improve the network performance we have introduced specific regularization filter applied on the final

test outputs in order to enforce similar global (higher order) statistics observed in the training set. The filter included 3 constraints: shift towards the desired mean, stretching or compressing the variance around the desired mean and forcing the shift of the differences among bot-player individual win-rates towards the same relative differences observed in the training set.

Deeper structures with multiple concatenated ELMs in the 2nd level have also been tested to no statistically significant improvement in the generalization ability of the network compared to the architecture shown in Fig. 2. If 2 ELMs were concatenated in the 2nd stage, with different kernel widths, the resulting preliminary test RMSE was in a range of [5.05, 5.1]. Similarly, a network with a 4-ELMs chain in the second level received the same RMSE of 5.1%. These observations indicate that further attempts to correct regression errors bring no additional value to the design instead just modeling propagated noise and bringing re-optimization overhead.

VII. CONCLUSIONS

The regression network presented in this paper has been developed and submitted as a competitive entry to the AAIA Data Mining Competition 2018, concerned with the prediction of win-rates of four AI bot players, playing the game "Heartstone: Heros of Warcraft" among each other with different initial decks of cards and hero characters. The proposed regression was hierarchically designed to combine the advantages of Extreme Learning Machine and few other complementary state-of-the-art regression models in the first level and improve the final performance through supervised decision fusion and error correction in the second level. Our solution received the final RMSE of 5.65% and scored the 2nd place in AAIA'2018 Data Mining Competition.

REFERENCES

- [1] ScienceDaily, <https://www.sciencedaily.com/>.
- [2] Five reasons why online games have become so popular, <https://www.belfasttelegraph.co.uk/woman/life/five-reasons-why-online-games-have-become-so-popular-28656803.html>
- [3] J.H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Statist.*, vol. 29, no. 5, pp. 1189-1232, 2001.
- [4] XGBoost, <https://github.com/dmlc/xgboost/>.
- [5] XGBoost:Machine Learning Challenge Winning Solutions, <https://github.com/dmlc/xgboost/tree/master/demo#machine-learning-challenge-winning-solutions>, Retrieved 2016-08-01.
- [6] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, pp. 489-501, 2006.
- [7] J.P. Vert, K. Tsuda, and B. Schölkopf "A primer on kernel methods," *Kernel Methods in Computational Biology*, 2004.
- [8] R. Tibshirani, "Regression Shrinkage and Selection via the lasso," *Journal of the Royal Statistical Society, Series B (methodological)*, Wiley, vol. 58, no. 1, pp. 267-88.
- [9] L. Breiman, "Better Subset Regression Using the Nonnegative Garrote," *Technometrics. Taylor and Francis*, vol. 37, no. 4, pp. 373-384, 1995.
- [10] R. Tibshirani, "The lasso Method for Variable Selection in the Cox Model," *Statistics in Medicine*, vol. 16, pp. 385-395.
- [11] V. Nair and G. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines," *27th International Conference on Machine Learning*, pp. 807-814, 2010.
- [12] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "A Simple Way to Prevent Neural Networks from Overfitting", *Journal of Machine Learning Research*, vol. 15, pp. 1929-1958, 2014.