

# Toward an Intelligent HS Deck Advisor: Lessons Learned from AAIA'18 Data Mining Competition

Andrzej Janusz<sup>\*†</sup>, Tomasz Tajmayer<sup>\*‡</sup>, Maciej Świechowski<sup>‡</sup>,  
Łukasz Grad<sup>†</sup>, Jacek Puczniewski<sup>†‡</sup> and Dominik Ślęzak<sup>\*</sup>

<sup>\*</sup>Institute of Informatics, University of Warsaw, Poland

<sup>†</sup>eSensei, Poland

<sup>‡</sup>Silver Bullet Labs, Poland

Contact Email: janusza@mimuw.edu.pl

**Abstract**—We summarize AAIA'18 Data Mining Competition organized at the Knowledge Pit platform. We explain the competition's scope and outline its results. We also review several approaches to the problem of representing Hearthstone decks in a vector space. We divide such approaches into categories based on a type of the data about individual cards that they use. Finally, we outline experiments aiming to evaluate usefulness of various deck representations for the task of win-rates prediction.

**Keywords**—Data Mining Contest; Data Representation Learning; Hearthstone: Heroes of Warcraft; Win-rates Prediction

## I. INTRODUCTION

AAIA'18 Data Mining Challenge: *Predicting Win-rates of Hearthstone Decks* was the fifth contest organized in association with the FedCSIS conference series. The topic was a follow-up of the previous edition of the challenge, related to a popular collectible card game *Hearthstone: Heroes of Warcraft* [1]. This time, participants were asked to predict win-rates of various Hearthstone (further abbreviated as HS) decks played by AI bots, based on games played with similar decks.

HS is a good framework for carrying out AI research. One kind of research is development of autonomous game-playing agents. The game is popular (more than 70M active players), highly competitive (one of the biggest eSport games) and yet it has combinatorial-game-like structure. Some notable bots reported in the literature are MetaStone (<https://github.com/demilich1/metastone>), Hearthranger (<http://www.hearthranger.com>), HearthBot [2] and Silverfish [3]. The second type of research revolves around analysis of the game in order to, e.g., help the players build better decks [4], [5]. A common need in all such investigations refers to an appropriate data representation that can be considered with respect to cards, decks or players. As one could see in the entries in the '17 installment of the competition, a good card representation is the backbone of ML-based playing agents. This aspect is even more crucial when it comes to win-rates prediction.

The competition outlined in this paper refers to both bot and non-bot research. On the one hand, we employ our AI algorithms [6] to generate massive bot vs. bot game logs data set. Our bots play using different decks and simulate different levels of real players. On the other hand, the competition task is related to another thread of our investigations, i.e., designing an advisory platform that helps players compose better decks [7]. Indeed, the top competition solutions, especially those

taking into account the importance of the aforementioned card representations, can lead us toward new insights with respect to what decides about the win-rates of particular decks.

The rest of the paper is organized as follows: In Section II, we summarize the competition. In Section III, we discuss several approaches to constructing hybrid vector representations of HS decks and compare empirically usefulness of the obtained representations for predicting the win-rates. In Section IV, we draw some directions for future research.

## II. AAIA'18 DATA MINING CHALLENGE

The competition (<https://knowledgepit.fedcsis.org/contest/view.php?id=123>) took place on April 3 – May 7, 2018, under the auspices of 13<sup>th</sup> International Symposium on Advances in Artificial Intelligence and Applications (<https://fedcsis.org/2018/aaia>) which is a part of the FedCSIS conference series. The purpose of this challenge was to discover reliable methods for predicting win-rates of HS decks. The task was to construct a prediction model that can learn win chances of new decks, based on the history of match-ups between AI bots playing with similar decks. To give participants freedom of choosing a representation of the data, apart from a preprocessed data set in a tabular format, there were provided JSON files with detailed descriptions of each game. We were interested whether the data regarding the way in which cards are played during the game can be useful in the proposed task.

The training data set contained logs from 299680 games played between four bots which used 400 decks. Another 200 decks – combined with the same bots as in the training set – were used as a test set. The win-rates of the bot-deck pairs from the test set were computed based on 300000 simulated play-outs. In those games, one of the bots used a deck from the training set, and the other one – from the test set. The decks were created by randomly mutating a set of 13 deck archetypes that at the time of the competition were commonly used in ladder matches by human players (12 top-rated archetypes and one group of decks consisting of only basic cards).

To generate the games, we defined four HS bots that differed in: a) available time limit for performing a move and b) available knowledge about the opponent hand (*full\_info* or *limited\_info*). Eventually, the following configurations were used: A1 – *limited\_info* & 1 second per move, A2 – *limited\_info* & 2 seconds per move, B1 – *full\_info* & 1 second per move, B2 – *full\_info* & 2 seconds per move.

TABLE I: Final RMSE results and number of submissions from top-ranked teams. The last row shows the result obtained by a baseline solution – a fully-connected neural network with two hidden layers, trained on the bag-of-cards representations of decks.

team name	rank	number of submissions	final result
hieuvq	1	195	5.5734
amy	2	149	5.6482
jj	3	225	5.6676
dymitruta	4	258	5.6962
amorgun	5	6	5.8474
...	...	...	...
baseline	26	–	8.8645

#### A. Evaluation of results and participation in the challenge

Submissions from participants were managed by Knowledge Pit [8]. Each submission had to be properly formatted, containing predictions of win-rates for every deck-bot pair from the test set. Each of the teams could submit multiple solutions. As a quality criterion for submissions, we selected the RMSE measure. The submitted solutions were evaluated on-line and the preliminary results were published on the competition leaderboard. The preliminary scores were computed on a subset of the test set, fixed for all participants. The size of this subset corresponded to randomly chosen 10% of the test decks. The final evaluation was conducted after competition’s completion using the remaining part of the data.

Apart from submitting their predictions, each team was obligated by competition rules to provide a brief report describing the approach used. The description had to cover utilized learning models, as well as the steps of data preprocessing and feature extraction. Only teams which sent a valid report qualified for the final evaluation. In this way, we were able to collect a vast amount of information regarding various representations of HS decks and the state-of-the-art approaches to this type of prediction problems. After completion of the challenge, the final results were published on-line. The scores obtained by top-ranked teams are presented in Table I.

#### B. Summary of the competition results

Our contest attracted 204 teams from 28 countries. The countries with the highest number of registrations in the challenge were Poland (119), Russia (28), United Kingdom (9), United States (8) and India (5). Among the participating teams, 82 submitted at least one solution file which was ranked at the public leaderboard. Over a half of those teams decided to disclose their approach by uploading short reports.

The top solutions were obtained by ensembles of regression models. The winners combined linear regression with deep neural networks. The second team blended a tree-based boosting model (XGBoost) with Extreme Learning Machines and LASSO regression. Other models that performed well were SVR (SVM  $\epsilon$ -regression) and Gaussian processes.

To represent the decks as vectors, many of the top-ranked teams encoded them as bags-of-cards, i.e., vectors of a size equal to the number of distinct cards in the data, where each element indicates how many cards of a corresponding type are present in a deck. The winners augmented such a representation using aggregated card properties, e.g., the total

health of minions in a deck, the number of spells, the number of minions with a taunt ability, etc. A few teams incorporated into their representations advanced knowledge about HS, e.g., indicators of cards’ relative strength defined by experts. Still, none of the top 20 teams used information from game logs to augment their representations of decks.

### III. REPRESENTATIONS OF HEARTHSTONE DECKS

Since one of our objectives for this competition was to find out whether information regarding the way cards are played during HS duel can be useful for predicting win-rates of decks, we further investigated this problem in a series of experiments. We created various card representations using three different data sources. Based on those representations, we built vector embeddings of decks and compared their usefulness by measuring performance of several prediction models. Let us discuss the obtained results.

#### A. Bag-of-cards and its transformations

The most common representation of HS decks is a bag-of-cards – by an analogy to a bag-of-words representation of textual documents. A deck is regarded as a set of card IDs. Its vector representation has a length equal to the number of available cards. The  $i$ -th vector’s position expresses the amount of copies of the  $i$ -th card in the given deck.

Such a simple representation turned out to be very effective for predicting win-rates. It was utilized by many of competition entrants, including all the top three teams. However, in nearly all cases, it was augmented by additional information extracted from the cards, e.g., a distribution of card mana costs. The augmentation was usually done by aggregating properties of cards included in the deck. For this purpose, participants often used external knowledge bases, such as the one provided by HearthstoneJSON API (<https://hearthstonejson.com/>).

The dimensionality of bag-of-cards representation can be reduced using some text mining techniques, such as SVD. A deck representation in the space of latent concepts can be used by itself. It can be also combined with the others to express combinations of cards often appearing in the same deck.

#### B. Aggregation of cards represented in a vector space

Representation of a deck can also be created by aggregating representations of individual cards. Information about the cards can be acquired from various sources, such as:

- a database with card properties and textual descriptions (e.g.: HearthstoneJSON, Wiki)
- a database of players’ decks (e.g.: HearthPWN.com)
- logs from games between human players or AI bots (e.g.: the data used in our competition)

Specific algorithms for creating vector embeddings of HS decks based on the data from the first two of the above sources are described in [7]. Game logs can be utilized to generate embeddings of cards, e.g., using a word2vec model [9] in which card IDs correspond to terms and their use sequences extracted from game logs are treated as documents.

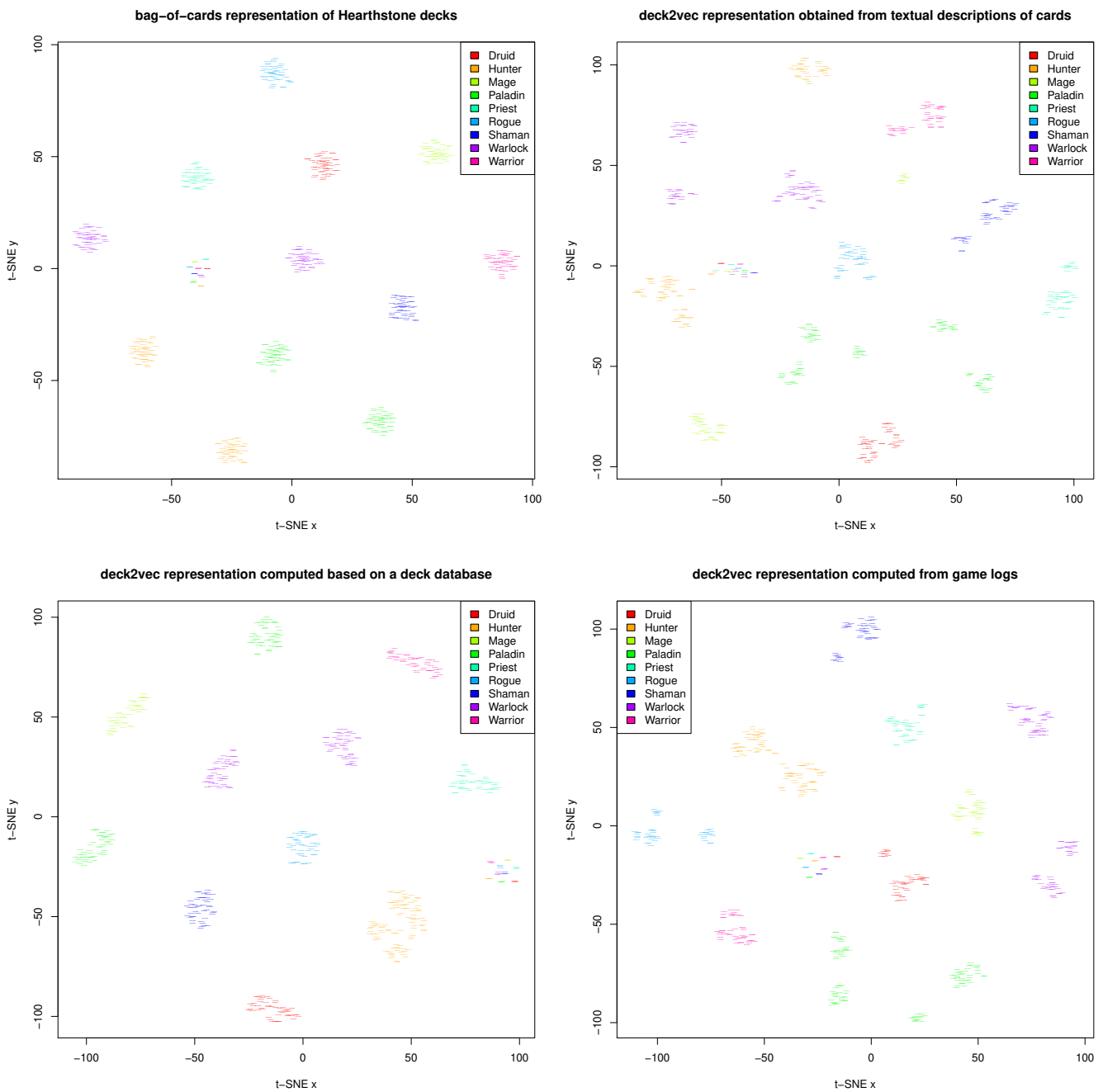


Fig. 1: A t-SNE-based visualization of four deck representations. Top-left: bag-of-cards. Top-right: aggregated embeddings derived from textual descriptions. Bottom-left: representation based on an on-line deck database. Bottom-right: representation based on the competition game logs.

The derived representations may cover different aspects of card similarity. For instance, the word2vec embeddings of cards computed from their descriptions can capture information about their basic properties. A representation derived from compositions of decks created by players may be better at expressing the aforementioned interchangeability [7]. Finally, a representation created from game logs may convey more information about the way cards are used during a game.

The aggregation of cards can be performed in many ways too. The simplest approach is to take a mean of the card vectors. To prevent losing too much information, such a deck representation can be extended by, e.g., min, max and standard deviation of card vectors (computed dimension-wise).

To visualize differences between the above approaches, we used them to represent the decks from our competition. Figure 1 shows these representations embedded into a two-

dimensional space using the t-SNE algorithm [10]. Only the bag-of-cards method allows to identify all deck archetypes which we used to generate the data (12 visible groups of decks for nine hero classes and one mixed group with decks containing only basic cards). For the representation obtained by analyzing a deck database, two groups of decks for the Hunter hero class were merged together. This is quite a good indicator given the fact that these archetypes were *FaceHunter* and *AggroBeast*, which share significant portion of cards and have a similar game plan. For the representation derived from textual descriptions, some archetypes were split into separate groups. Such a division is also visible when using representations extracted from game logs. Therein, however, apart from the group of decks composed of basic cards, there are no clusters with mixed decks from different hero classes.

### C. Predictive power – experimental evaluation

We performed a series of experiments to evaluate the impact of the deck representation methods on a predictive performance of various regression models. We trained four models, namely the aforementioned Gaussian Process Regression (GPR) [11], SVR [12], Multi-Layer Perceptron (MLP) and  $K$ -Nearest Neighbors (KNN) on the competition data and compared their results obtained for four representations of decks. Table II shows the final scores.

The best score was achieved by combination of the bag-of-cards and GPR. On the other hand, the same representation combined with KNN regression was the worst. This fact highlights a need of adjusting data representation for a given prediction model. The second best representation was the one derived from a deck data set. Its score was even slightly better than for the bag-of-cards used together with MLP and KNN. The most disappointing were results of the representation based on game logs. It suggests that more advanced methods of learning deck representations from logs need to be developed to fully utilize this source of information.

Nevertheless, motivated by a diversity of the deck clustering results, we decided to check if the learned representations can contribute some additional knowledge to prediction models. We concatenated each of the vector representations with the bag-of-cards and evaluated their performance in a combination with GPR. These results are included in Table II. Surprisingly, for every representation we obtained considerably better results than for the plain bag-of-cards. Moreover, when we concatenated all four representations, we achieved the best score (RMSE 5.465) among all entries in our challenge (see Table I). This confirms the benefit of using diverse sources of the data for constructing prediction models.

## IV. CONCLUSIONS

We summarized AAIA'18 Data Mining Competition organized at the Knowledge Pit platform, whose topic was predicting win-rates of Hearthstone decks. The outcomes of the competition show that various machine learning models are capable to accurately assess the quality of new decks, based on the data regarding performance of similar decks.

Our own comparison of deck representations created using various sources of information about cards revealed that the simplest approach, i.e., the bag-of-cards, can be successfully

TABLE II: RMSE scores obtained for four prediction models and discussed deck representation methods, where BC stands for bag-of-cards and DD, TD and GL denote representations derived from a deck data set, textual card descriptions and game logs, respectively. The last row shows results for selected concatenated vector representations.

representation:	BC	DD	TD	GL
GPR model	5.812	7.059	7.768	9.274
SVR model	6.185	7.107	7.435	7.873
MLP model	7.035	6.823	8.573	8.018
KNN model	10.158	9.152	9.348	8.960
representation:	BC+DD+TD+GL	BC+DD	BC+TD	BC+GL
GPR model	<b>5.465</b>	5.503	5.733	5.629

employed. Moreover, our experiments show that it is possible to considerably improve performance of prediction models by training them on combined representations from different sources. We believe that such a hybrid approach will move us one step further in our ultimate goal of designing an advisory platform for helping players in composing their decks.

## ACKNOWLEDGMENTS

This research was co-funded by Smart Growth Operational Programme 2014-2020, financed by European Regional Development Fund under GameINN projects POIR.01.02.00-00-0150/16 and POIR.01.02.00-00-0184/17, operated by National Centre for Research and Development in Poland.

## REFERENCES

- [1] A. Janusz, T. Tajmayer, and M. Świechowski, "Helping AI to Play Hearthstone: AAIA'17 Data Mining Challenge," in *Proceedings of FedCSIS 2017*, 2017, pp. 121–125.
- [2] A. R. da Silva and L. F. W. Goes, "HearthBot: An Autonomous Agent based on Fuzzy ART Adaptive Neural Networks for the Digital Collectible Card Game Hearthstone," *IEEE Transactions on Computational Intelligence and AI in Games*, pp. 170–181, 2017.
- [3] S. Zhang and M. Buro, "Improving Hearthstone AI by Learning High-level Rollout Policies and Bucketing Chance Node Events," in *Proceedings of IEEE CIG 2017*, 2017, pp. 309–316.
- [4] P. García-Sánchez, A. Tonda, G. Squillero, A. Mora, and J. J. Merelo, "Evolutionary Deckbuilding in Hearthstone," in *Proceedings of IEEE CIG 2016*, 2016, pp. 1–8.
- [5] A. Stiegler, C. Messerschmidt, J. Maucher, and K. Dahal, "Hearthstone Deck-construction with a Utility System," in *Proceedings of SKIMA 2016*, 2016, pp. 21–28.
- [6] M. Świechowski, T. Tajmayer, and A. Janusz, "Improving Hearthstone AI by Combining MCTS and Supervised Learning Algorithms," in *Proceedings of IEEE CIG 2018*, 2018, In print.
- [7] A. Janusz and D. Ślęzak, "Investigating Similarity between Hearthstone Cards: Text Embeddings and Interchangeability Approaches," in *Proceedings of IEEE SMC 2018*, 2018, In print.
- [8] A. Janusz, D. Ślęzak, S. Stawicki, and M. Rosiak, "Knowledge Pit – A Data Challenge Platform," in *Proceedings of CS&P 2015*, 2015, pp. 191–195.
- [9] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed Representations of Words and Phrases and Their Compositionality," in *Proceedings of NIPS 2013*, 2013, pp. 3111–3119.
- [10] G. Hinton and S. Roweis, "Stochastic Neighbor Embedding," *Advances in Neural Information Processing Systems*, vol. 15, pp. 833–840, 2003.
- [11] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2006.
- [12] A. J. Smola and B. Schölkopf, "A Tutorial on Support Vector Regression," *Statistics and Computing*, vol. 14, no. 3, pp. 199–222, 2004.