Recommending Bids on Dou-DiZhu Poker Games: A Deep Learning Approach

Bo Yuan and Shuqin Li
Computer School
Beijing Information Science and Technology University
Beijing, China
wikinet@163.com, lishuqin_de@126.com

2020 Chinese Automation Congress (CAC) | 978-1-7281-7687-1/20/$31.00 ©2020 IEEE | DOI: 10.1109/CAC51589.2020.9327775

***Abstract*—Computer aided game playing strategy provides a unique view towards artificial intelligence studies. In addition to symmetric games, establishing intelligent models for non-symmetric games are more challenging. A vivid example is the Dou-DiZhu game, a famous traditional Chinese poker game, which requires one player playing cards against the alliance of the other two players. This paper has made a first attempt to establish a machine-aided playing strategy to help player to call bids with a higher gain. In particular, a Convolutional Neural Network (CNN) based recommendation model is proposed to combine the feature of player's hand sequence and strength. To validate the feasibility of the proposed method, a real-world dataset is utilized to examine the performance of the recommendation results outputted the proposed model. The results of comparative experiments show that the proposed model achieves a higher precision, recall, F1 and accuracy performance than traditional machine learning models and single-feature based deep learning models.**

Keywords*—*Bid recommendation, Dou-DiZhu poker game, Machine-aided game playing, Convolutional Neural Networks

I. INTRODUCTION

With the advent of the era of big data, artificial intelligence technology has increasingly become the research focus of scholars. As an important indicator for testing the development level of artificial intelligence, the research on machine game theory has also made considerable progress [1]. Machine game theory refers to the construction and training of computer systems that can simulate a series of behaviors of humans in the game environment of information acquisition, information analysis, intelligent decision-making, and automatic learning. This theory reflects the level of development of artificial intelligence to a certain extent [2].

At present, the research of game theory has penetrated into various fields of current society, such as economy, military, entertainment, etc., affecting our lives in all aspects. In these applications, game theory is generally divided into two types: complete information games and incomplete information games. In a complete information game, participants can observe the complete game state, and both parties can fairly see the complete game information except for the game strategy [3]. For games with incomplete information, players cannot see all the game information. Some of the game information exists in a non-public form, which leads to the complexity and variability of the game situation. Most poker, mahjong, Dou-DiZhu poker games, etc. are all incomplete information games [4,5].

As a highly competitive zero-sum game, the Dou-DiZhu poker game has the characteristics of incomplete information game, phased game process, coexistence of cooperation and confrontation, which makes this type of game very similar to the game scene of economic society. So it has high research value [6,7]. In this type of game, bidding, as an important procedure, plays an important role in the final win or loss of the game. A good bidding strategy can give the most suitable bids and points according to the hands on the game field, so as to solve the problem of difficult bidding due to different hands in each game. In recent years, through the development and exploration of related researchers, intelligent game systems and bidding game strategies have achieved certain developments, some of which have been applied in some game platforms [8].

Aiming at the problem of user bidding recommendation in Dou-DiZhu games, this paper proposes a method for bidding recommendation based on a deep convolutional neural network that integrates card sequence and card strength features in Dou-DiZhu games. This method is constructed and integrated in a targeted manner. The user's card strength and card order feature vectors are extracted and analyzed at different granular levels through a one-dimensional convolutional network, which effectively solves the problem that general machine learning and deep learning models cannot effectively handle the precise and efficient bidding of Dou-DiZhu games.

II. RELATED WORK

Early research on game games was mainly carried out on some foreign complete information game such as chess games. The backgammon agent developed by Berliner defeated the then world champion Villia [9]. In 2016, AlphaGo based on deep reinforcement learning and Monte Carlo tree search defeated human Go players [10,11], and the deep learning algorithm it used has received extensive attention and research. Although these algorithmic researches have achieved good results in human-computer games, board games are a complete information game, and their research results cannot be fully used in poker game research.

Since the 1990s, incomplete information games have become one of the hot issues in the field of artificial intelligence research. In 1999, Billings et al. [12] developed the first Texas Hold'em machine game program. Although the modified program is a rule-based method, it laid the foundation for subsequent research. Davidson et al. [13] conducted a study on restricted Texas Hold'em, which used neural networks to model opponents and predict the behavior of opponent players, thereby improving the accuracy of the prediction results. The well-known game program Poki [14] used the research of Texas Hold'em in practice by combining a variety of modeling methods, and achieved good results. In 2016, Yakovenko et al. [15] proposed a supervised learning model based on convolutional neural networks to deal with poker problems. This method requires a large amount of labeled data during training and cannot adjust strategies in real time during the game with different opponents. Wu et al. [16] aimed at the problem of too large state space in poker machine games, and proposed an algorithm based on hole card

Authorized licensed use limited to: University of London: Online Library. Downloaded on December 28,2024 at 23:18:48 UTC from IEEE Xplore. Restrictions apply.

abstraction and hand evaluation. This algorithm uses new data features to improve the strategy bias modeling, which can not only reflect the opponent's weakness. Li et al. [8] proposed a bid evaluation model based on a classification algorithm. The model uses the Adaboost algorithm to train a classifier from the Dou-DiZhu game data, and then evaluates the opponent's card strength to determine whether to bid. Li Saisai et al. [17] conducted a study on card playing based on the CNN training model, extracted dominant features from the perspective of the landlord based on certain historical card game information, and predicted the hand of the upper and lower farmers based on the characteristics of the landlord’s hand. Although the above game methods proposed for card games have achieved some good results, most of them need to rely on a large number of training data sets, and the feature values are not effectively distinguished.

To sum up, although the above methods have made some good progress in the research of game theory, these methods do not effectively deal with the problem of accurate and efficient scoring in Dou-DiZhu games. Based on the deep convolutional neural network, this paper first extracts and analyzes the features at different granular levels through the one-dimensional convolutional network, and then integrates the card order and card strength features in the Dou-DiZhu game, so as to make accurate score recommendations for players.

III. PROBLEM DEFINITION

In a Dou-DiZhu poker game, three players take turns drawing from a standard shuffled deck of 54 cards until the remaining three cards are not drawn. At this point, in accordance with certain rules (such as by the previous round of the declarer) from a player, the counterclockwise start to take turns calling bids. Each participant may call only once, and the score of the call may be: ‘1’, ‘2’, ‘3’, or not call (i.e. ‘0’). If the previous player's score is less than 3 points, the next player can choose to "not call" or choose the greater points to contest the “landlord” role (i.e. the one player who plays cards against the alliance of the other two players.). If the called bid reaches 3 points, the next player can choose to the "double" operation to contest the landlord, at this time, the score reaches 6 points (3×2), and so on, the maximum score can reach 24 points, i.e., 0, 1, 2, 3, 6, 12, 24. After the calling is completed, the first participant with the highest score becomes the "landlord" and gets the remaining 3 cards in the draw phase, with the remaining two players as the counterparty.

It can be seen from the above process that the disadvantage of the "landlord" participant in the game is that he has no companions, so he has fewer hand cards than the amount of cards of two players. At the same time, the "landlord" has the advantage of 3 extra cards, which gives him a greater chance of getting a more powerful hand cards. Since the "landlord" can get a higher reward after winning a game, and similarly, the "landlord" can get a greater punishment after losing a game, players need to evaluate their hands when they choose to call as "landlord", so that they can play all their hands as soon as possible to win in the next process of playing cards.

According to the above introduction, the task of the biding recommendation model is to make suggestions for call points according to the hands obtained by the current game participants, so as to provide them with reference, or serve as the instruction for the robot to make decisions in the game. Specifically, the bidding recommendation model can be simplified and abstracted as the problem of whether to contest the "landlord". The formula is expressed as follows:

$$f(X) = \begin{cases} 1, & \text{contest the "landlord"} \\ 0, & others \end{cases} \tag{1}$$

where $f(X)$ is the prediction model, and X is the hand of a certain round accepted by the model and other characteristics of the round that can be obtained. When the output of $f(X)$ is 1, it indicates that the model recommends the participant to call for a higher score to get the role of "landlord". Similarly, when the output of $f(X)$ is 0, it indicates that the model recommends participants to give up competing for the role of "landlord". X can be further defined as follows:

$$X = \langle x_{card}, x_{exrta} \rangle \tag{2}$$

where x_{card} is the sequence information of the participant's hand in that round, and x_{exrta} is the additional feature extracted from the hand and the game. The specific definitions of x_{card} and x_{exrta} are given in the following model design section.

IV. THE MODEL DESIGN

The purpose of feature extraction is to transform the hand card sequence of the player into an input vector that can be recognized by the neural network model. Therefore, the feature extraction in this paper consists of two parts, namely, constructing the card sequence vector and the card strength vector.

First, for the card sequence vector, the Dou-DiZhu poker game only considers the value of the cards, regardless of the suit of the cards. Moreover, each card type from A to K has four cards of the same shape but different suits and trumps of different sizes, so there are a total of 15 card types in Dou-DiZhu poker game. Considering the large amount of data of card game information, this paper adopts one-hot encoding to encode different cards. That is, each card corresponds to a 15-dimensional vector. When this card belongs to one of the 15 card types, the corresponding position of the vector is 1, and the rest bits are 0. Such as "A" of the corresponding vector for [1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0] . "2" the corresponding vector for [0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0], and so on. Finally, the game participants' hand card sequence will be transformed into a 17×15 matrix, constituting the card sequence vector sequence characteristics.

In addition to the hand sequence, the card shape also determines the size of the participant's "card strength". However, different from a simple number ratio, the Dou-DiZhu poker game has extra size rules for different card types. When the participant plays a particular card, the subsequent participant must plays a hand of the same type or a hand of the same face of all four cards (i.e. "bomb") or both joker cards (i.e. "king bomb"), otherwise, it cannot be regarded. This process continues and continues until the end when two participants do not play, and the player chooses a new type of card to continue playing. In terms of probability, the more the number, the more complex the card shape is, the more difficult it is to match, so the player has a greater advantage in the game. However, the above rules are made by human beings, which may not be recognized by the neural network model.

Authorized licensed use limited to: University of London: Online Library. Downloaded on December 28,2024 at 23:18:48 UTC from IEEE Xplore. Restrictions apply.

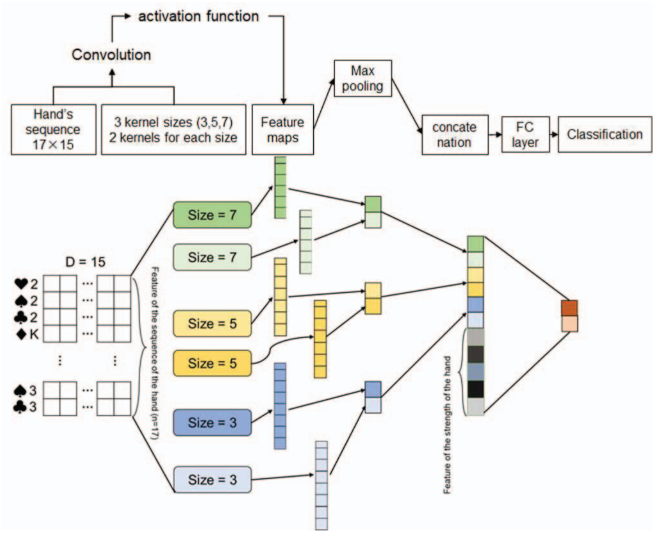

Fig. 1. Card recommendation model of two-player poker game based on the card order vector and the card power vector.

Therefore, this paper proposes a method to extract the card power characteristics from hand cards according to the artificially formulated card type.

Thus, this paper selects the statistics of the number of 7 kinds of cards as a supplement and combines with the combined card type extraction results in 8 kinds of games to form the card strength vector. The card strength vector is combined with the previously defined card order vector to form an 18×15 model input matrix.

Inspired by TextCNN model, this paper proposes a convolutional neural network model as shown in Fig. 1, which is used to card recommendation in two-player poker game.

First, let the first part of the input to the model be an $n \times D$ card order matrix. That is, each piece of data has n segments, and each segment is represented by a vector of D dimensions. The matrix is convolved with three different size convolution kernels (K_1,K_2,K_3). Let $X_{i:i+j}$ represent all values from X_i to X_j, and with a convolution kernel K of height h and width d convolved with X_i through $X_{(i+h)}$, then the activation function is used to get the corresponding Feature maps F. Define * to be the convolution operation and *activation*() to be the activation function, then the convolution operation can be expressed as:

$$F(i) = activation(K * X_{i:i+h}) \quad (3)$$

In the traditional CNN network, the convolution characteristic dimension remains unchanged, but the convolution dimension in TextCNN model will be reduced. Therefore, after convolution operation, an n-h+1-dimensional vector C can be obtained. Its formula is as follows:

$$c = [c_1, c_2, \ldots, c_n] \quad (4)$$

Since convolution kernel with different heights is used in the convolutional layer process to capture features at different scales, the channel characteristic dimensions of output after convolution will be inconsistent according to the size of the convolution kernel. Therefore, in order to unify dimensions and avoid dimension explosion problem, we used the k-Max-pooling method to construct pooling layer. Pool each convolution eigenvector and sample the reduced dimension into k values, that is, the maximum k values of each convolution eigenvector are extracted to represent the feature, and it is assumed that these maximum values represent the most important features extracted.

$$c' = max_k(c) \quad (5)$$

After pooling operation, we concatenate c' with the card power vector e, and transmit the signal through two fully connected layers, and finally output the predicted value of the model. Then the signal is propagated through the two fully connected layers. Finally, output the predicted value of the model. This step is shown in the following formula.

$$\hat{y} = fc\big(dropout\big(fc(c' \bowtie e)\big)\big) \quad (6)$$

When *dropout*() is the forward propagation of the neural network, let the activation value of a neuron stop working with a certain probability. The aim is to make the model more generalized so that it does not rely too heavily on some local features. $fc()$ is the full connection layer, and its formula is:

$$fc(x) = activation(W \cdot x + b) \quad (7)$$

where W and b are weight and bias respectively, and they are both trainable parameters in the model.

The predictive value of the model $\widehat{y_c}$ represents the classification task prediction of whether to grab landlords or not. Finally, the model selects the Cross-Entropy Loss function as the Loss function of the model:

$$L_c = -[y_c \log \widehat{y_c} + (1 - y_c) \log(1 - \widehat{y_c})] + \|w\|_2 \quad (8)$$

where $\|w\|_2$ is the regular term of all trainable parameters. The training of the model is verified by tenfold crossing. The training process is only applied to the training set, and the result of the final model on the test set is used as the performance evaluation basis of the model. During the training process, the model adopts Adam optimization for gradient descent.

V. Experiments and Results

A. Dataset

The experimental data used in this paper comes from the real-world data of Dou-DiZhu poker game provided by a well-known online game platform in China named "Ourgame". The original data and pre-processed data sets are shown in Table 1. For the classification task of recommending whether the user competes for the landlord, this paper only measures whether the user finally wins after competing for the landlord as the reference basis of the model. Its physical meaning can be regarded as "If the user competes for the landlord, is the user likely to win in the end?"

Table 1. The description of the data set.

Item	#	Description
Match numbers	5,401,939	The total number of games in the original data
User numbers	2,426,146	Contains the total number of users
Training set size	3,341,650	The data were cleaned and pretreated after retention, of which 1,680,177 were positive examples
Testing set size	61,940	After cleaning and preprocessing the retained data, of which the positive cases were 30,970

Authorized licensed use limited to: University of London: Online Library. Downloaded on December 28,2024 at 23:18:48 UTC from IEEE Xplore. Restrictions apply.

We select the following parameters and settings. First, the size of the card order matrix input by the model is 17×15, that is, n=17 and D=15. The size of the convolution kernel is $K_1 = 3$, $K_2 = 5$, $K_3 = 7$. We also select 1-max-pooling for pooling operation, used ReLu function as the activation function and set the dropout rate to 0.5. In the training process, the Learning rate of the model was set to 0.0001, the batch size was 1024, and the total training rotation was 30.

The operating environment of the experiment was a server with Intel I9-7900x CPU, 32GB memory, 4 Nvidia GTX1080 Ti as GPU and Ubuntu 16.04 LTS as operating system. The deep learning framework uses Keras (version 2.1) and TensorFlow (version 1.12) as a backend component. All experimental program implementations are implemented in Python (version 3.6).

B. The contrast baselines

Support vector machine (SVM): it is a common machine learning model, which is widely used in forecasting and classification tasks with fewer super parameters and stronger generalization and interpretability.

Long and short term memory network (LSTM): it is a specific structure of a circulating neural network. Its advantage is that sequential data can be transmitted between networks in the long and short term, so as to realize the sequence-based supervised learning task.

Text convolutional neural network (TextCNN): it creatively applies one-dimensional convolution to the processing of serialized text information, thus realizing feature extraction and convolution operation under different lengths, and having fewer parameters and faster training process than RNN.

C. The experimental results

The performance comparison between the proposed model and the comparison baselines are shown in Table 2 and Fig. 2. It is observed from the results that the model proposed in this paper achieves the best predictive performance in four indicators: accuracy rate, recall rate, F1 value and accuracy rate. Since the number of positive and negative examples in the training set and the test set is almost equal, there is no significant difference between F1 and accuracy in the comparison model. In addition, the poor SVM results indicate that traditional machine learning methods may have the over-fitting problem when processing the massive data mentioned above. In terms of the deep learning model, the performance of LSTM and TextCNN is weaker than the model proposed in this paper. This indicates that the structure of the model can extract the implied characteristics of the players' hands more effectively, predict the possibility of winning rate by using this hand, and thus realize more accurate suggestions of landlord competition.

Table 2. The performance results of comparison models.

Models name	Precision	Recall	F1	Accuracy
SeqStgCNN (Ours)	0.802077	0.798063	0.800065	0.800565
SVM	0.605272	0.603552	0.604411	0.604973
LSTM	0.772310	0.771811	0.772061	0.772134
TextCNN	0.783026	0.781434	0.782229	0.782451

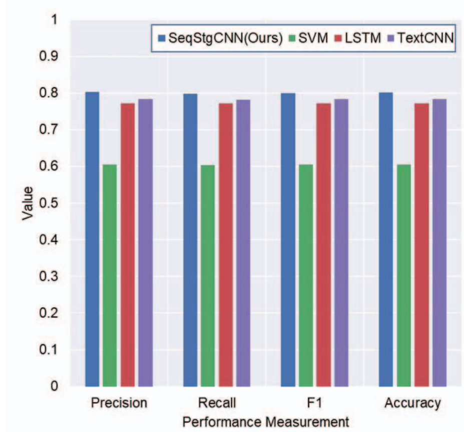

Fig. 2. The performance of the comparison models on the recommendation task of landlord contention in four classification indexes.

VI. CONCLUSION

This paper focused on the biding recommendation in a typical non-symmetric Chinese poker playing game named Dou-DiZhu. A CNN-based network is proposed to establish recommendation and prediction on choosing the sides according to player's hand sequence and strength. Compared with SVM, LSTM and TextCNN, our proposed model derived a better performance in precision, recall, F1 and accuracy rates. We believe these results would bring a novel view for other more non-symmetric computer-aided game playing studies.

ACKNOWLEDGMENT

The authors would like to thank Ourgame for accessing their gameplaying data. This work is supported by 2019 Key Research and Cultivation Project to Promote Connotation Development and Scientific Research Level of Beijing Information Science and Technology University under grant NO. 5212010937, General Project of Science and Technology Plan of Beijing Information Science and Technology University under grant KM201911232002, and University Connotation Development Promotion Project under grant NO.5112011019.

REFERENCES

[1] J. Zheng, "Research and Application of Computer Game of Imperfect Information," South China University of Technology, 2017.

[2] J. Li, "Study of Evaluation Algorithm in Imperfect Information Game," Harbin Institute of Technology, 2014.

[3] J. Lin, "Imperfect Information Games Based on Q-Learning," Harbin Institute of Technology, 2009.

[4] X. Xu, et al., "Challenging issues facing computer game research," CAAI Transactions on Intelligent systems, vol. 3, no. 4, pp. 288-293, Aug. 2008.

[5] P. Wang, "Research on Imperfect Information Machine Game Based on Deep Reinforcement Learning," Harbin Institute of Technology, 2017.

[6] Y. Li, "Research on Machine Game Model and Algorithms of Bridge Bidding," Beijing university of posts and telecommunications, 2019.

[7] Z. Zhang, "Design and Implementation of Bridge Play Program," University of Science and Technology LiaoNing, 2008.

[8] S. Li, et al., "Design and Implementation of an Adaboost-based Landlord Bidding Strategy," Control & Decision Conference, 2017, pp. 7627-7631.

Authorized licensed use limited to: University of London: Online Library. Downloaded on December 28,2024 at 23:18:48 UTC from IEEE Xplore. Restrictions apply.

[9] Hans J. Berliner, “Backgammon Computer Program Beats World Champion,” Artificial Intelligence, vol. 14, no. 2, pp. 205-220, 1980.

[10] S. David, et al., “Mastering the Game of Go with Deep Neural Networks and Tree Search,” Nature, vol. 529, no. 7587, pp. 484-489, 2016.

[11] S. David, et al., “Mastering the Game of Go Without Human Knowledge,” Nature, vol. 550, no. 7676, pp. 354-359, 2017.

[12] DR. Papp, “Dealing with Imperfect Information in Poker,” University of Alberta, 1998.

[13] A. Davidson, et al., “Improved Opponent Modeling in Poker,” International Conference on Artificial Intelligence, 2000, pp. 1467-1473.

[14] A. Davidson, “Opponent Modeling in Poker: Learning and Acting in a Hostile Environment,” University of Alberta, 2002.

[15] N. Yakovenko, et al., “Poker-cnn: a Pattern Learning Strategy for Making Draws and Bets in Poker Games Using Convolutional Networks,” Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, 2016, pp. 360-367.

[16] T. Wu, “Research on Incomplete Information Machine Game Algorithm and Opponent Model,” Wuhan University of Technology, 2018.

[17] S. Li, et al., “Research on Fight the Landlords’Single Card Guessing Based on Deep Learning,” International Conference on Artificial Neural Networks, 2018, pp. 363-372.

Authorized licensed use limited to: University of London: Online Library. Downloaded on December 28,2024 at 23:18:48 UTC from IEEE Xplore. Restrictions apply.