Deep Learning for Evaluation of Game and Puzzle Positions

Takafumi Wake[†] Hiromu Okada[†] Shuji Jimbo[†]
[†]Okayama University

Abstract

In recent years, it has been demonstrated that a powerful game AI can be implemented in Go with the combination of well-learned neural networks and traditional search algorithms. In this study, we made an equivalent to the policy network in the Go AI for, FreeCell, a solitaire card game, by deep learning. We also made an equivalent to the value network in the Go AI for Pentago, a two-player board game. The network structure, ResNet, was adopted for the above cases of deep learning, and supervised learning was performed with the evaluation values of moves in FreeCell and positions in Pentago as teacher data. It is well-known that ResNet is useful for deep learning with deep convolutional neural networks.

1. Introduction

In recent years, AI technology which we applied deep learning to in various fields is used, and it is expected in future that the utilization range spreads. AlphaGo proves that it is possible to implement the powerful game AI of Go by using deep learning, which has long been considered impossible due to the search tree is too large. In this paper, we performed situation evaluation using deep learning about Pentago which was FreeCell that was card-game for one player and board game for two players. We hope that we reduce search space by using those our results for making game AI. In addition, by incorporating neural network into the search, it is possible to allocate processing on the GPU instead of on the CPU, leading to a cost reduction when trying to obtain a large number of solutions in a short time. It is hoped that this becomes a model case of reduction in cost technique when we were going to apply AI technology in other fields. In this paper, we focus on experiment for card game FreeCell. In addition, we experiment on situation evaluation so far about Pentago which is boardgame for two in this laboratory. Pentago is game that complete analysis by Irving [5][6] has been already per-formed, and data which player wins when we continue choosing the best moves. As a result of constructing a deep neural network that evaluates wins and losses for a specific number of turns (called slices) from the start of the game, we have now obtained a correct answer rate of about 85% for slice 15. In the future, we plan to evaluate wins and losses in the final stages of the game and compare them with conventional learning performance. In this paper, we describe the experiments we conducted on the single-player card game FreeCell.

2. Method

This section describes the steps required to build a learning model.

2.1 Plan

We build a deep learning model in which supervised learning is performed by using situation information as learning data and evaluation of the situation as a teacher. The evaluation for the situation was the optimal next move derived using the solver program created by the team. This corresponds to the policy network in AlphaGo, which is a Go AI, and it is expected that the AI that implements this network will reduce the width of the search space.

We used Keras [8], which is a deep learning framework, for model development, and Tensorflow as a backend. Each version is shown in Table 1. ResNet (Residual Network) [9], which is also used in Alpha Go, is adopted as the method of the network model, and in learning FreeCell, the number of filters is increased instead of the vertically deep network like normal ResNet. As a result, we constructed a wide network laterally, and constructed WideResNet [10] that has the performance of ordinary ResNet with many layers and facilitates parallel processing. ResNet is a method devised to solve the problem that the gradient disappeared due to the extremely small difference between the input and output when the layers of the network are deepened, and the accuracy decreases. It is a network that has resistance to gradient disappeared by implementing a shortcut structure in which the input flows directly into the output of a certain layer and by learning the residual difference between input and output.

2.2 Creating a dataset

Determine the format of data given to the network. In the free cell, the information of positions used as the input data is represented by a one-dimensional array in which the switching of tableaus is represented by delimiters. In

Table 1: Development environment

Keras	2.2.4
Tensorflow-gpu	1.9.0

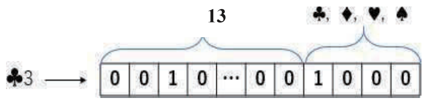

Figure 1: Example of card information

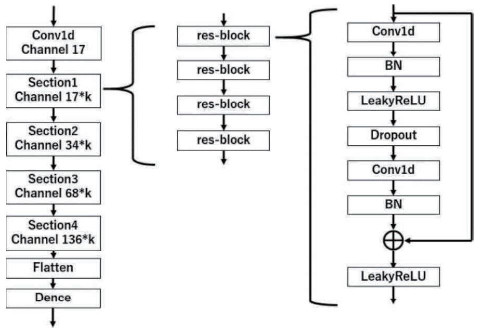

Figure 2: Network structure

addition, the information on the card is represented as 17 channels by connecting two one-hot expressions so that 4 suits and 13 numbers (ranks) can be distinguished (Fig. 1). As data sets, 2 million training data and 500,000 verification data generated by the solver program under development were used.

2.3 Experiment

In FreeCell, when moving cards, there are a total of 12 locations, 4 free cells and 8 tableaus, and 16 destinations, including 4 home cells. Therefore, 192 types of launch patterns can be envisioned by these combinations, and from this, 168 types of launch patterns can be envisioned, excluding improbable actions such as movement between free cells, the same free cell, and tableau. According to this, 168 multi-class classifications were performed with the expected move as the correct label. Although it is desirable to use the accuracy rate as an evaluation index, there are situations in which multiple moves are best for a free cell at the same time. The purpose of this research is to make the search tree of the solver program small by using the learning results, and it is not necessary to find a single optimal move, so a small number of move candidates should be obtained. Therefore, top_4_accuracy, which represents the proportion of correct labels among the top four labels in the predicted probability distribution of each label, was used as the evaluation index this time.

The created network is WideResNet with 33 layers of convolution depth (Fig. 2). As shown on the right side of the figure, the residual block including two convolutional layers is constructed, and the number of filters is doubled for every four residual blocks. Constant k in the figure is a coefficient of the width of the WideResNet network, and was set to 3 in this experiment. LeakyReLU is used as the activation function. The batch size was 1000, the filter size was 8, and the dropout rate of the Dropout layer in res-block was 0.3. As the optimizer, Adam [11], which is gene-rally used in machine learning, and Momentum SGD [12] were tested, and the results obtained by learning under the same conditions other than the optimizer showed more accurate results. Detailed tuning was performed on the model using.

3. Result

The learning results are shown in Table 2. Figure 3 shows the state of learning at the highest accuracy. In the FreeCell, the learning result top_4_accuracy was 0.831. The number of epochs is set to 30. The main causes of over fitting are lack of learning data and too complicated learning model .[13] Therefore, the training data was reduced to 1.28 million and the validation data to 320,000 without changing the model.

As a result, the top_4_accuracy of the verification data was 0.702, which was about 13% less accurate. In addition, at this point, there was a discrepancy between the accuracy of the training data and overtraining. Furthermore, in order to simplify the model, the dataset was not modified and the number of Sections in the learning model was reduced to three, which resulted in a decrease in learning accuracy.

4. Discussion

Since the learning accuracy decreases due to the reduction of the data set and the accuracy decreases due to the simplification of the learning model, it is speculated that the main cause of over fitting is not due to the complexity of the model but due to lack of training data. it can. However, it is necessary to try because there are means to suppress over fitting through model tuning, such as adding regularization and adjusting the dropout rate. We will continue to increase the amount of datasets. Regarding regularization, it is conceivable to add the weight regularization provided to the Keras layer and to implement the regularization method proposed in ResNet .[14] [15]

The FreeCell solver program currently used to generate the dataset may include meaningless moves or detours. This is not the case, and it is thought that it is difficult to achieve extremely high accuracy with this learning. In addition, in the development of a free cell solver with a small search tree, which is the final objective of this re-

Table 2: Experimental result

Network	Accuracy
WideResNet with 33 convolutional layers and $k = 3$	top_4_accuracy = 0.831 (accuracy = 0.480)
WideResNet with 33 convolutional layers and $k = 3$ (small amount of data)	top_4_accuracy = 0.702

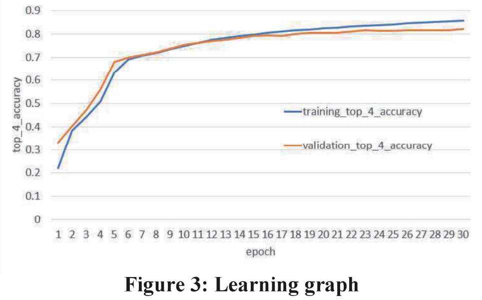

Figure 3: Learning graph

search, learning results are prioritized in search. Since it is planned to be used to determine, the search tree can be expected to be reduced to some extent even at the present time. Since it is thought that a better quality data set can be created by it, even if it is the present condition, by applying the learning result to the solver program, more accurate learning can be performed.

In addition, as a new method, attention has recently been paid to the field of image processing [16], and we will confirm whether the Self-Attention layer, which is a natural language processing technology, is also useful in this experiment.

5. Conclusion

In this paper, we constructed a network that evaluates the situation in two types of games, FreeCell, a single player card game, and Pentago, a two-players board game. The model construction was based on ResNet. In the FreeCell, the predictors were evaluated with the proportion (top_4_accuracy) of the top four probability distributions in which the correct labels were predicted, and the prediction accuracy of the verification data was 0.831 at 30 epochs. There is a tendency for over-learning in the current learning results, and we aim to further improve accuracy by trying to suppress over fitting. Moreover, by applying the learning results to the solver program, a better quality data set can be expected and further improvement in accuracy can be expected. In the future, we will carry out re-experiments while incorporating new methods such as Self-Attention.

Acknowledgment

Part of this research was carried out with the support of Earthrise Company.

References

[1] 斉藤康己. アルファ碁はなぜ人間に勝てたのか. ベストセラーズ, 2016.

[2] 大槻知史. 最強囲碁 AI アルファ碁解体新書深層学習, モンテカルロ木探索, 強化学習から見たその仕組み. 翔泳社, 2017.

[3] Silver, David, et al. Mastering the game of Go with deep neural networks and tree search, nature 529.7587 484, 2016.

[4] Silver, David, et al. Mastering the game of go without human knowledge, Nature 550.7676 ,pp. 354,2017

[5] Geoffrey Irving. Pentago is a first playerwin. https://perfect-pentago.net/.

[6] Geoffrey Irving. Pentago is a first playerwin: Strongly solving a game using parallel in-core retrograde analysis. arXiv preprint arXiv:1404.0743,2014.

[7] 新谷敏朗. カードゲーム「フリーセル」の性質に関する考察. 福山大学工学部紀要=The Memoirs of the Faculty of Engineering, Fukuyama Universitypp. 63-68, 2018.

[8] Chollet,François and others. Keras. https://keras.io, 2015.

[9] He, Kaiming and Zhang, Xiangyu and Ren, Shaoqing and Sun, Jian. Deep residual learning for image recognition, pp. 770–778, 2016.

[10] Zagoruyko, Sergey and Komodakis, Nikos. Wide residual networks. arXiv preprint arXiv:1605.07146, 2016.

[11] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization, arXiv preprint arXiv, 1412.6980, 2014.

[12] Ruder, Sebastian. An overview of gradient descent optimization algorithms., arXiv preprint arXiv:1609.04747, 2016.

[13] Hawkins, Douglas M. The problem of overfitting. Journal of chemical information and computer sciences, 44.1, pp.1-12, 2004.

[14] Huang, Gao and Sun, Yu and Liu, Zhuang and Sedra, Daniel and Weinberger, Kilian Q. Deep networks with stochastic depth, pp. 646–661. Springer, 2016..

[15] Gastaldi, Xavier. Shake-shake regularization. arXiv preprint arXiv:1705.07485, 2017.

[16] Zhang, H., Goodfellow, I., Metaxas, D., & Odena, A. (2018). Self-attention generative adversarial networks. arXiv preprint arXiv:1805.08318.

Appendices

Rule of the games

The description of the outline of each game to handle in this report.

FreeCell

FreeCell is a kind of solitaire that is a card game for one person, 52 cards without joker are arranged in 8 rows (tableau) in random order, and all cards are suit to a goal called home cell (It is a complete information game that rearranges each type in ascending order. The tableau card can be moved only in the heaven (bottom in the figure) card, and there is a restriction that the cards must be moved so that the colors alternate in descending order. Also, once placed in the home cell, the card cannot be moved, and there are four spaces called free cell where one card can be placed freely.[7] Figure 4 shows an example of the initial position of the FreeCell

.

Pentago

Players take turns when placing marbles on the game board and twisting one of the game blocks 90 degrees. A player is free to rotate any of the game blocks, regardless of which game block the player placed the marble on. The player who arranged five own marbles either before and after of the rotate gets victory. A winning row of five marbles can occur vertically, horizontally or diagonally, anywhere on the board and will span two or three game blocks. Figure 5 is an example of the situation of the Pentago.

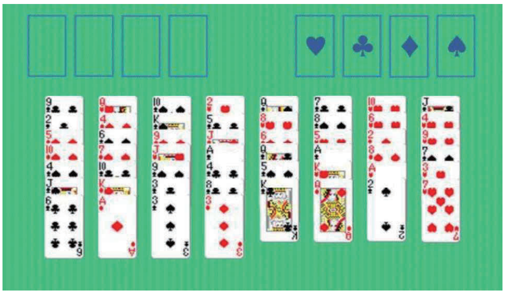

Figure 4 :FreeCell

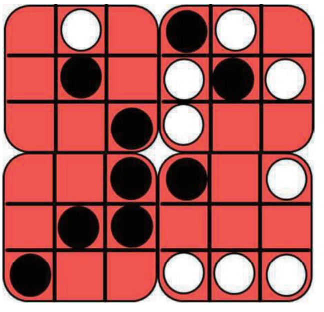

Figure 5: Pentago