WIN PREDICTION BASED ON HERO COMBINATION IN GAME DOTA 2 USING NEURAL NETWORK BACKPROPAGATION ALGORITHM

Jaka Septian

Teknik Informatika – Universitas Komputer Indonesia
Jl. Dipatiukur 112-114 Bandung
E-mail : jksptn@gmail.com

ABSTRAK

DOTA 2 is a cooperation-oriented game involving two teams for each match, each team consist of five players. Each player controls a character called Hero. The result of the match can be predicted based on selected heroes. But predicting the result of each match is not easy, because the Hero that can be chosen totaled 110 Heroes, then there are about 4.689 x 1013 possible combinations Hero that can occur in one match.

Neural Network algorithm has the capability of memorization and generalization. Memorization ability is the ability of Neural Network to recall perfectly the pattern has been learned, with this ability it can recognize patterns Neural Network game DOTA 2 match that have been happen to guess the outcome of the match with a same pattern in the future. Generalization ability is the ability of Neural Network to produce an acceptable response of the input pattern that similar (but not identical) to the patterns that have been previously studied. Based on above fact, a simulator that is capable of predicting the winner based on a hero combination in game DOTA 2 using Neural Network algorithm is built.

Keywords : Neural network, Backpropagation, Prediction.

1. INTRODUCTION

DOTA 2 is a game of multiplayer online battle arena (MOBA). MOBA is the kind of cooperation-oriented game involving two teams for each match, each team, consist of five players who must destroy each other tower and the main bastion opponent to win the game. Each player controls a character called Hero, then one player and another player must choose a different Hero [1].

Played hero has the ability and the different roles with other heroes. Because every Hero has a different role capability, then the combination of the Hero chosen by both squads will affect the match outcome and have possibility to do predictions. By predicting the outcome of the game based on a combination of Hero, players can choose an appropriate hero to fight a combination of opponents hero and increase the chances of winning the game.

Other than usual match in DOTA there are also many tournaments within the game officialy or unofficialy tournaments that held by the third party. Winners of the tournaments get the prize that is usually not the least and quite tempting [1]. Therefore to predict the outcome of a match wins can help players determine the step in order to achieve victory.

Predicting the outcome of each match is not easy, because of heroes that can be selected is totaled 110, so there are about 4.689 x 10^{13} possible combinations Hero that can occur in one match. Each match that has been made will form the pattern of victory based on a combination of the selected Hero. The pattern can be used as a reference to predict a victory, but the result pattern very large. Neural network algorithm is an algorithm that is often used in studying and recognizing patterns in data. Neural network algorithms have the ability of memorizing and generalizations. Memorisation ability of Neural network is the ability to recall perfectly the pattern that has been learned, with this ability to it can recognize match patterns in DOTA 2 games that have been happen to guess the result of the match with a similar pattern in the future. Generalization ability is the ability of neural network to produce an acceptable response to the input patterns that are similar (but not identical) to the patterns that have been previously studied [2].

Backpropagation is one of the Neural network architecture that has forward learning process and backward error correction. This model is widely used both for the process of recognition and prediction with a pretty good degree of accuracy [3]. Sandro Radovanovic and Milan Radojičić conduct research using Backpropagation method to predicting victory football match based on the player who played in the match. From these studies we can conclude Backpropagation method can be used to predict the outcome of a game based on a combination of variables were selected [4]. In this final project simulator for predicting victory based on a combination of hero in DOTA 2 game using neural

network Backpropagation neural network will be built.

The aim of this study is to help the players to predict the outcome of a match based on a combination of heroes chosen by both teams and to determine the performance such as accuracy and optimal parameters for Backpropagation method to predicting match outcome.

1.1 Neural network

Neural networks or artificial neural networks is one of the studies on artificial intelligence. Neural network is used to solve the problems that are complex and difficult to understand, which along large amounts of data on the issue have been collected. Neural networks are looking for patterns and relationships in very large data that is too complex and difficult to analyze human use of hardware and software that resemble the patterns of processing in the human brain

. Neural network component consists of interconnected neurons. Neurons that will transform the information received via the output connection toward other neurons. In neural network relationships between neurons is known as the weights [5]. Neural network components as shown in Figure 1 consists of inputs (information) weight (certain values), the activation function works when the input is entered in accordance with the specified value of threshold, otherwise the activation function is not activated, and when neurons is activated, the neuron will send output through weights

.

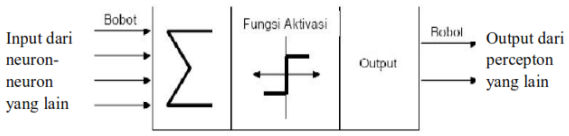


Figure 1. Neural Network Component

1.1 Backpropagation

Backpropagation is a type of neural network that uses supervised learning methods. In the supervised learning there is a pair of data inputs and outputs are used to train the neural network to obtain the desired weight. Backpropagation method using the error output to change the value of the weights in the backward direction. To get this error, advanced propagation phase (forward) must be done first. At the time of forward propagation, the neurons activated by using sigmoid activation function, as shown in equation 1.

Activation function transform the input total on a neuron to produce an output signal (outgoing activity). Activation function used in this study is the sigmoid function. This function is used for Neural Network trained using backpropagation method. Sigmoid function has a value in the range of 0 to 1. This function is often used for networks that require the output value lies in the interval 0 to 1. However, this function can also be used by the neural network with output value of 0 or 1.

$$f(x) = \frac{1}{1+e^{-x}} \quad (1)$$

Backpropagation Neural network architecture consists of three layers, namely input layer xi, hidden layer zj, and the output layer yk. Input layer and hidden layer weights associated with vij and between the hidden layer and output layer weights connected by wjk .. The architecture can be seen in Figure 2.

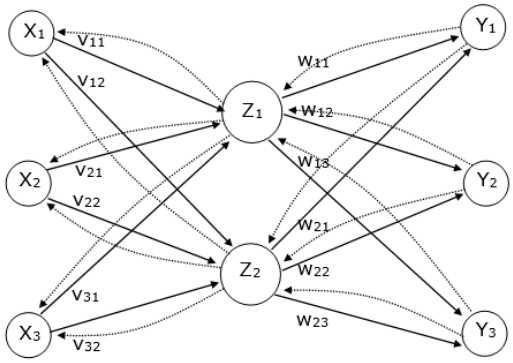


Figure 2. *Backpropagation* Architecture

There is no specific formula to determine the number of hidden layers and the number of neurons of it layer to calculate it, but there is a rule-of-thumb that is often used either to determine the number of hidden layers and the number of hidden neurons. Rarely problems that require two hidden layers, although a neural network with two hidden layers can represent any kind of shape functions. There is no theoretical reason for using a neural network more than two hidden layers. In fact for a wide variety of problems there is no reason to use more than one hidden layer [7]. To determine the number of neurons in the hidden layer is very important in building network architectures. Using too few neurons in the hidden layer is likely to produce underfitting, otherwise use too many neurons in the hidden layer will likely result an overfitting and if the training data is very large there will be a very long time to do the training [7].

Rules-of-thumb that are often used in determining the number of neurons in the hidden layer is as follows:

1. The number of neurons in the hidden layer is between the number of neuron in the input layer and neurons in the output layer [7].
2. The number of neurons in the hidden layer is 2/3 the input layer neuron number, plus the number of output layer neurons [7].
3. The number of neurons in the hidden layer should be less than double the number of input layer neurons [7].

4. the number of neurons in the hidden layer influenced the amount of training data [8].

To train Backpropagation, Neural network given input pattern as a training pattern then the pattern will be heading to units in the hidden layer to transfer to the units that are in the output layer. Output of the hidden layer will be forwarded to the output layer and output layer will provide feedback so-called temporary output yk. when yk ≠ tk where tk is the expected output, then the difference (error) output while u'k will be propagated back (backward) in the hidden layer and forwarded to units in the input layer. Therefore, the process is called backpropagation) [6].

2. Research Content

2.1 The used method.

In this study, the method used is the Backpropagation Neural network algorithm.

2.1.1 Backpropagation method

In the Backpropagation method, there are two stages of the process, as is follows:

1. Training Stage, at this stage back propagation method given amount of training data and its target.
2. Testing stage or use test is done after a learning phase.

Stopping condition can be used by two ways, as is follows:

1. Limit the iteration.
   For example, the network will be train until 500 training iteration
2. Limit the error.
   For example, determine the mean square error.

Generally steps in training using Backpropagation Neural networks are as follows:

1. Determine the number of inputs (input pattern), hidden layer and output (target training).
2. Giving the initial random values for all weights between the input-hidden and hidden-output layer.
3. Do the feedforward
   a. Each unit of input (xi) receives the input signal and the signal is sent to all units in the hidden layer.
   b. Each unit in the hidden layer (Z_inj) summing the weighted input signals:

$$z_in_j = v_{0j} + \sum_{i=1}^{n} x_i v_{ij} \qquad (2)$$

use the activation function to calculate the output signal:

$$z_j = f(z_in_j) \qquad (3)$$

   c. Each unit in the output layer (Y_ink) summing the weighted input signals:

$$y_in_k = w_{0j} + \sum_{i=1}^{n} z_i w_{jk} \qquad (4)$$

use the activation function to calculate the output signal:

$$y_k = f(y_in_k) \qquad (5)$$

4. Do the backpropagation
   a. Each unit in the output layer (yk) receiving a target pattern associated with the input pattern, calculate error information:

$$\delta_k = (t_k - y_k)\, f'(y_in_k) \qquad (6)$$

Then calculate the correction weights (which will be used to improve the value of w_{jk}):

$$\Delta w_{jk} = \alpha\, \delta_k\, z_j \qquad (7)$$

Also calculate the bias correction (which will be used to improve value of w_{0k})

$$\Delta w_{0k} = \alpha\, \delta_k \qquad (8)$$

Send δ_k to all unit in the under layer

   b. Each unit in the hidden layer (Z_inj) delta summing inputs (of units that are on the top layer):

$$\delta_in_j = \sum_{k=1}^{m} \delta_k w_{jk} \qquad (9)$$

multiply this value by the derivative of the activation function for calculating error information:

$$\delta_j = \delta_in_j f'(z_in_j) \qquad (10)$$

then calculate the correction weights (which will be used to improve value of v_{ij}):

$$\Delta v_{ij} = \alpha\, \delta_j\, x_i \qquad (11)$$

Also Calculate the bias correction (which will be used to improve value of v_{0j}):

$$\Delta v_{0j} = \alpha\, \delta_j \qquad (12)$$

   a. Each of output unit (y_in_k) improve the bias and its weights:

$$w_{jk}(new) = w_{jk}(old) + \Delta w_{jk} \qquad (13)$$

each hidden unit (z_in_j) improve the bias and its weights :

$$v_{ij}(baru) = v_{ij}(lama) + \Delta v_{ij} \quad (14)$$

5. Test the stopping condition
   a. Is the minimum error limit has been reached.
   b. Is the maximum iteration limit has been reached.

2.2 Data Analist

2.2.1 Data Source

The data used are taken using the Steam Web API in .xml file that is subsequently converted and stored into a database.

2.2.2 Input data

The data used in win predicting is entire heroes data which totaled 110 heroes. The data is organized into variable X1 to X110 into the data input to the system. Each variable can have a value of -1 if chosen by the opposing team, 0 if not selected and 1 if the chosen by ally team as shown in Table 1

Table 1. Input Data.

Value	Description
-1	Heroes variable that chosen by opponent
0	Heroes variable that not selected
1	Heroes variable that chosen by ally

2.2.3 Output data

Output data or output generated by the system in the form of a representation of a two-state value that may occur as a result of the end of a match. Output data can be seen in table 2 below.

Table 2. Output data.

Value	Description
1	Win
0	Not Win

2.3 Train the Backrpopagation Mehtod

Input on the training process is the history of the match to see the selected heroes and team winner of the match. Forms the input to input signals based on the selected hero for both squads as in Table 3.

Table 3. Selected Heroes

No	Ally Heroes	Opponent Heroes
1	Juggernaut (X8)	Anti-Mage (X1)
2	Kunkka (X60)	Dark Seer (X55)
3	Slardar (X65)	Weaver (X88)
4	Ancient Apparation (X68)	Gyrocopter (X72)
5	Nyx Assasin (X88)	Troll Warlord (X95)

With target output is 1 (ally team win).

The data will be used for training the network architecture that is already built. The network has three layers; input layer, hidden layer and output layer. In this case, the input layer has 110 neurons and one neuron bias, hidden layer neuron has 2 and 1 node bias, while the output layer has one node. Network architecture can be seen in Figure 3.

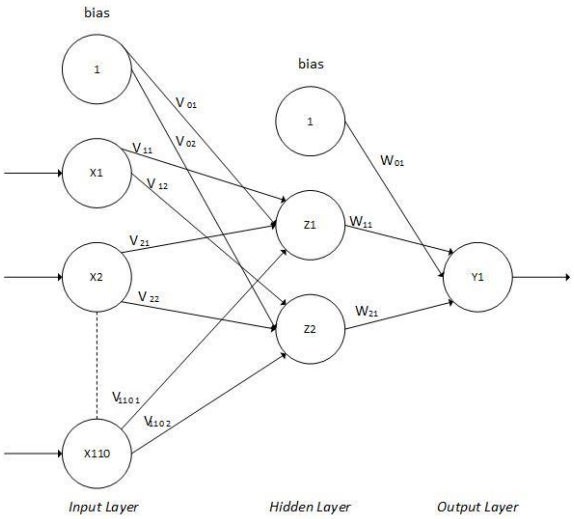


Figure 3. Backpropagation Neural Network Architecture

Step for the training process are as follows:

1. Determine the maximum iteration and learning rate. In this example is used the following limits:
   Maximum Iterations = 1000
   Learning rate = 0.05
2. Giving the initial random values for all weights between the input-hidden and hidden-output layer.
3. While maximum iteration has not been reached do the follows :
   a. *feedforward*
   b. *backpropagation*

In the second data, also carried out the same operations using weights of the first data processing results as the initial weights. This process is repeated until the maximum iteration is reached.

2.4 Pengujian Metode *Backpropagation*

Tests conducted to test the networks with weighted that have been given the training process using training data. The testing process just did feedforward and compare the results with the threshold specified value.

Results of the activation function in the output layer compared with a certain threshold value. Suppose taken threshold value = 0.5, meaning that if

the value of yk > 0.5 then the output value is 1, but if the value yk < 0.5 output value is 0.

2.5 Research Result

Here are the results of research conducted prior testing of the final result.

2.5.1 Performansi

Performance testing aims to determine the performance and the optimal parameters of backpropagation method implemented in the simulator predictions of victory game DOTA 2. The test method is used by 10-fold cross validation, the data overall is divided into 10 subsets where each subset has different data. The data used in this test is the data matches during March to April 2015. The total amount of data used were 1440 data, the data is divided into 10 subsets of data consisting of 140 games for each subset of data. Each subset will be tested using the other as a subset of training data, so each test performed 10 times. This testing resulted in massive accuracy of back propagation method based on the amount of output data are very close to or in accordance with the target output. The test is performed using the simulator predictions of victory DOTA 2 games that have been made.

In the back propagation method that is implemented there are three variables that can be changed is the number of hidden layer, learning rate, and the maximum number of iterations. Changes in the value of the maximum iteration is done to limit the training process backpropagation method, while the change of the number of hidden layer and learning rate affects the speed and accuracy of learning of back propagation method.

The accuracy of this method can be seen from the number of predicted outcomes simulator that is close to or in accordance with the original results. The number of results accurate prediction value per cent to ease in knowing the value of the accuracy of the system.

1. Testing Scenario 1

In the first scenario testing, testing is done by changing the method of backpropagation epoch boundary from 100 to 1000, while the other variable is filled with the same value. Results of the first test scenario can be seen in Table 4.

Table 4 . Result of first scenario testing

Epoch	Learning Rate	Jumlah *Hidden Neuron*	Akurasi
100	0.05	5	49.93 %
200	0.05	5	48.82 %
300	0.05	5	49.52 %
400	0.05	5	50.76 %
500	0.05	5	50.14 %
600	0.05	5	50.14 %
700	0.05	5	50.35 %
800	0.05	5	50.56 %
900	**0.05**	**5**	**50.97 %**
1000	0.05	5	49.17 %

Limits epoch chosen based on the value of the greatest accuracy. Based on the above test, selected epoch limit of 900 because it has the greatest accuracy is 50.97%. 900 value is then used to perform testing to determine the value of learning rate.

2. Testing Scenario 2

In the second scenario testing, testing methods backpropagation done by changing the value of learning rate of 0.05 to 0.5 and limit the epoch of 900 according to the results of the testing scenario 1, while the other variable is filled with the same value.Results from the second test scenario can be seen in Table 5.

Table 5 . Result of second scenario testing

Epoch	Learning Rate	Jumlah *Hidden Neuron*	Akurasi
900	**0.05**	**5**	**50.97 %**
900	0.1	5	47.92 %
900	0.15	5	50.59 %
900	0.2	5	50.42 %
900	0.25	5	50.28 %
900	0.35	5	48.54 %
900	0.3	5	49.24 %
900	0.4	5	48.54 %
900	0.45	5	50 %
900	0.5	5	49.52 %

Learning rate values are chosen based on the greatest accuracy. Based on the above test, chosen value of learning rate 0:05 because it has the greatest accuracy is 50.97%. 12:05 learning rate value is then used to perform tests to determine the number of neurons in the hidden layer.

3. Testing Scenario 3

In the third scenario testing, testing backpropagation method is done by changing the number of neurons in the hidden layer of 5 to 50 neurons, epoch limit of 900 according to the results of the testing scenario 1, and the value of learning rate scenario 0:05 according to the results of the test 2. The results of the test scenario 3 can be seen in Table 6.

Table 6 . Result of third scenario testing

Epoch	Learning Rate	Jumlah *Hidden Neuron*	Akurasi
900	0.05	5	50.97 %
900	0.05	10	51.81 %
900	0.05	15	49.44 %
900	**0.05**	**20**	**52.01 %**
900	0.05	25	50.14 %
900	0.05	30	51.53 %
900	0.05	35	50.35 %
900	0.05	40	50.56 %
900	0.05	45	50.97 %
900	0.05	50	49.17 %

The number of neurons in the hidden layer is selected based on the greatest accuracy. Based on the above test, selected neuron number by 20 because it has the greatest accuracy is 52.01%.
Tests of this scenario is the final test to determine the optimal parameters of Algorithm Neural Network Backpropagation. Epoch optimal limit is 900, the value of the optimal learning rate is of 0.05 and the number of neurons in the hidden layer is optimized as much as 20 neurons. The accuracy achieved from this performance testing amounted to 52.01%.

2.5.2 Beta Test

Beta testing is testing conducted objectively. The method used in this beta test is by giving questionnaires to the players game DOTA 2. Samples taken as many as 30 players, and the results of the calculation tersebutakan questionnaire for assessment conclusions will be drawn.
This test aims to determine whether the simulator built to achieve the research objectives. Given the question "Is this simulator enables you to play the game DOTA 2?" Using a Likert scale cider scale of 1 to 5.

Here is the assessment score given by using a Likert scale on each of his statement, which is shown by Table 7.

Table 7. Answer Score

Skala Jawaban	Keterangan	Skor
SS	Sangat Setuju	5
S	Setuju	4
KS	Kurang Setuju	3
TS	Tidak Setuju	2
STS	Sangat Tidak Setuju	1

Results from beta testing can be seen in Table 8

Table 8. Testing Result

Keterangan	Skor	Responden	Jumlah Skor
Sangat Setuju	5	4	20
Setuju	4	20	80
Kurang Setuju	3	5	15
Tidak Setuju	2	-	-
Sangat Tidak Setuju	1	1	1
Jumlah		**30**	**116**

To find the value of the percentage of answers Likert scale used formula as follows:

$$\boldsymbol{P = \frac{S}{Skor\ Ideal} \times 100}$$

$$Y = \frac{116}{150} \text{ X } 100\% = 77\%$$

Based on the percentage value, the total score obtained is 116. So based on these data, it can be concluded that the simulator predictions of victory DOTA 2 game helps players in general is 77% with respondents 30 players from the expected 100% and located in the area Less Agree. The following is a continuum scale results shown in Figure 8

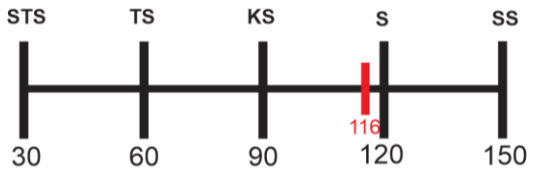


Table 8. Result Scale

3. CLOSING

Based on the results of research, analysis, system design, implementation and testing, it could be concluded that the simulator predictions of victory bersarkan combination hero in the game DOTA 2 uses an algorithm Backpropagation Neural network:

1. Simulator less petrified players predicting the winner of the game DOTA 2.
2. Can determine the performance and optimal parameter algorithm Backpropagation Neural network in data processing to be used as a prediction game.

Based on the results of the study and implementation of the accuracy of the algorithm Backpropagation Neural network in this study has not yet reached a high accuracy. To improve performance and better results, it is proposed some suggestions as follows:

1. Data collected should match much so that the algorithm Neural networks can learn more considering the pattern of the hero combinations that may occur very big.
2. Consider using algorithms or other methods to achieve high accuracy.

REFERENCE

[1] Valve Corporation, *DoTA 2 - The International*, Februari 2015. [Online]. Tersedia: http://www.dota2.com/international/overview/. [Diakses 12 Februari 2015]

[2] D. Puspitaningrum, *Pengantar Jaringan Saraf Tiruan*, Yogyakarta: Andi, 2006.

[3] C. Dewi dan M. Muslikh, *Perbandingan Akurasi Backpropagation Neural network dan ANFIS Untuk Memprediksi Cuaca*, Journal of Scientific Modeling & Computation, vol. 1, no. 1, pp. 7 - 13, 2013.

[4] S. Radovanović and M. Radojičić, *Premiere League Prediction Using Neural network*, Neuroph Java *Neural network* Framework, 2013. [Online]. Tersedia: http://neuroph.sourceforge.net/tutorials/SportsPrediction/Premier%20League%20Prediction.html. [Diakses 20 Juni 2015].

[5] S. Kusumadewi, *Artificial Intellegence (Teknik dan Aplikasinya)*, Yogyakarta: Graha Ilmu, 2003.

[6] W. Anggraeni, *Aplikasi Jaringan Syaraf Tiruan Untuk Peramalan Permintaan Barang*, *JUTI*, vol. 5, no. 2, pp. 99 - 105, 2006.

[7] J. Heaton, *Introduction to Neural networks for C#, 2nd Edition*, Chesterfield: Heaton Research, Inc, 2008.

[8] M. Leo, "Automatic Inspection of Aircraft Components Using Thermographic and Ultrasonic Techniques," in *Recent Advances in Aircraft Technology*, Intech, 2012, pp. 384-398.