Tennis Winner Prediction based on Time-Series History with Neural Modeling

Amornchai Somboonphokkaphan *, Suphakant Phimoltares †, and Chidchanok Lursinsap ‡

Abstract— **Tennis is one of the most popular sports in the world. Many researchers have studied in tennis model to find out whose player will be the winner of the match by using the statistical data. This paper proposes a powerful technique to predict the winner of the tennis match. The proposed method provides more accurate prediction results by using the statistical data and environmental data based on Multi-Layer Perceptron (MLP) with back-propagation learning algorithm.**

Keywords: Tennis, Neural Network, Multi-Layer Perceptron, Back-Propagation Learning Algorithm, Time-Series

1 Introduction

Nowadays, tennis is one of the most popular sports in the world. In every year, there are four major Grand Slam tennis events which are Australian Open, French Open, US Open and Wimbledon. These four grand slam tournaments are considered to be the most famous tennis tournament in the world. According to the four major grand slams, court surfaces of these tournaments are different; Australian and US Open is played on hard court, French Open is played on clay and Wimbledon is played on grass. Each court surface has its own characteristics and makes difference in speed and bounce of the ball. Clay court has a slower paced ball and a fairly true bounce with more spin. Hard court has a faster paced ball and very true bounce. Grass court has a faster paced ball and more erratic bounce. Moreover, the scoring system of Grand Slam tournament is also different. Typically for both men's and women's matches, the first player with two-sets winning wins the match. Unlikely to the general match, in the Grand Slam Tournaments, the first player who wins three sets wins the match [1].

*Department of Mathematics, Faculty of Science, Chulalongkorn University, Bangkok, 10330 Thailand. Email: amornchai.s@student.chula.ac.th.

†Department of Mathematics, Faculty of Science, Chulalongkorn University, Bangkok, 10330 Thailand. Email: suphakant.p@chula.ac.th.

‡Department of Mathematics, Faculty of Science, Chulalongkorn University, Bangkok, 10330 Thailand. Email: lchidcha@chula.ac.th.

Due to the growth of sport betting, predictions are widely used in many kinds of sports, especially tennis. The tennis prediction model is created to evaluate the chance of winning and the expected length of the match that players will face. Most people believe that the first serve person in the set has more advantage than another because most of the games often go like that so the first serve affect to the games' score [2]. Additionally, lots of players always make fault in the first serve and do better in the second serve so second serve might affect to the games' score too. Nevertheless, the first serve and the second serve affect to the games' score but there is another thing, that might be refuting an advantage of serves, it is strongly returns of serve. Moreover, the surface characteristics also affect to the players, e.g., some players perform better on grass but they may get worse on clay.

The first tennis model was proposed by Kemeny and Snell [3] which has only one parameter; probability of each player winning a point. Furthermore, Barnett and Clarke [4] proposed the prediction of a match played at the Australian Open 2003 by using Markov chain model set up in Microsoft Excel which has the probability of player A winning a point if player A is serving and the probability of player B winning a point if player B is serving as inputs.

Many research papers interested in the statistics on winning percentage of players on both serving and receiving. To use the statistical data, there are three problems associated with using these statistics as inputs to predict the tennis match. First, the statistics will be slightly out of date, unless the match in the first round. This problem is called *out of date data problem.* Second, the statistics are too detailed for the proposed method. This problem is called *too detail of data problem.* The third problem is to combine the individual player's statistics such as when two players meet on given surface; this problem is called *without environmental data problem.* Therefore, Barnett and Clarke [1] are covered the first and the second problems by updating the statistics as tournament progress and give more weight to more recent matches to cover the *out of date data problem* and manipulating statistics only the percentage of points won on serve and return

of serve for each player to cover the *too detail of data problem*.

The major purpose of this paper is to perform an advanced tennis model which provides more accurate prediction results by using the statistical data and environmental data based on Multi-Layer Perceptron. In this paper, back-propagation algorithm, a standard algorithm for supervised learning pattern, is used. In order to build the good tennis prediction model, the appropriate input features, which are based on two main types of data: statistical data and environmental data, are selected for the model. Certainly, these selected features are effective to the games' score. MLP is a basic sort of Artificial Neural Networks (ANN). ANNs are powerful technique to solve real world classification problems and have the learning ability from experience to improve their performance. ANNs are particularly effective for predicting outcome when the networks have large database of prior examples to draw on and able to deal with incomplete information or noisy data. ANNs can be classified based on topology; Single-Layer, Multi-Layer, Recurrent, etc.

In the next sections, the proposed method is applied to predict the winner of the tennis match and shows how to select input features of the MLP. In section 3, the experiments are set up and the results of the proposed method are presented. Subsequently, the comparison between our method and the other existing techniques is discussed in Section 4. Finally, the conclusion is given in the last section.

2 Proposed Method

2.1 Multi-Layer Perceptron (MLP)

An Artificial Neural Network (ANN) is a mathematical model or computational model based on biological neural network [5]. The network consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation. ANN is an adaptive system for which its structure can be changed using external and internal information flowing through the network during the learning phase. For the learning models, there are three major types of learning: supervised learning, unsupervised learning, and reinforcement learning.

The Multi-Layer Perceptron (MLP) is a supervised learning neural network with the input layer, hidden layer, and output layer. One input fed to one node of the network on the input layer corresponds to one input feature. In the case, N neurons are used to represent the N features of the input vector. The input layer gives out the corresponding input vector to each neuron in the hidden layer. In addition to the vector, there is a bias, a constant input of 1.0, included. In the hidden layer, the weighted sum (u_j), which is calculated from a set of connecting weights, w_{ji}, and the input vector, is fed into a transfer function, σ, which outputs a value h_j. The outputs from the corresponding hidden node of the hidden layer are also moved to the output layer. Then, in the output layer, the output value from each hidden neuron is multiplied by a weight (w_{kj}), and the results from weighted values are used to produce a combined value v_k. The weighted sum (v_k) is fed into a transfer function, σ, which outputs a value y_k, where y_k is the output k of the network.

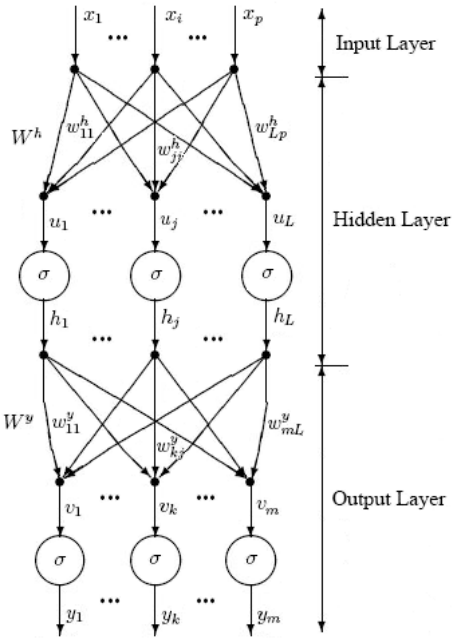

Figure 1. Multi-Layer Perceptron (MLP)

The goal of the training MLP process is to find the set of weight values that cause the appropriate output vector of the neural network to match the real target values as closely as possible. There are several issues for training a MLP network: defining how many number of the hidden layers used in the network, deciding how many number of neurons to use in each hidden layer, finding a technique to avoid local minima, converging to an optimal solution in a reasonable period of time, and validating the neural network to avoid *over fitting problem*.

To get more accurate prediction results, the MLP is applied to create the tennis model.

2.2 Input Features

Most of the researchers concentrate only on the statistical data such as percentage of first serve, winning percentage on the first serve, winning percentage on the second serve which directly affect to the match result.

To reduce the *out of date data* and *too detail of data problems*, this paper has manipulate these statistical data by collecting the statistical data of each player in the past

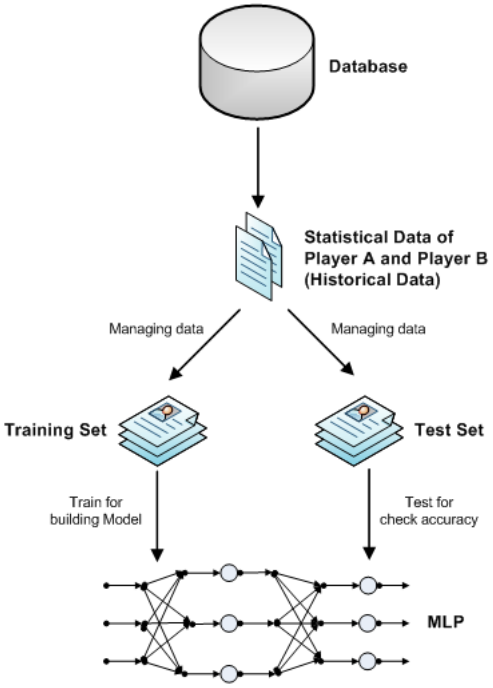

Figure 2. Work Flow

few years until the day before prediction, instead of using the statistical data that announce when the tournament starts. To reduce the *without the environmental data problem*, the court surface is selected to be one of input features. According to the court surface, it produces an effect to the individual statistic of the player. For example, some players do a better job on Grass but some players do not.

In this paper, both statistical data and environmental data are used. The selected statistical features consist of winning percentage on the first serve, winning percentage on the second serve, winning percentage on return serve, winning percentage on break point, winning percentage of played match and total point win. For the environmental data, the court surface is selected to be one of the input features. All input features used as input vector of the MLP can be shown as follows:

1. *Winning percentage on the first serve*, this feature represents a chance of the player to get point on the first serve.

$$Winning\ \%\ on\ 1^{st}\ Serve = \frac{1^{st}\ Serve\ Win}{Total\ 1^{st}\ Serve} \quad (1)$$

2. *Winning percentage on the second serve*, this feature represents a chance of the player to get point on the second serve

$$Winning\ \%\ on\ 2^{nd}\ Serve = \frac{2^{nd}\ Serve\ Win}{Total\ 2^{nd}\ Serve} \quad (2)$$

3. *Winning percentage on return serve*, this feature represents a chance of the player to gets point on receiving from opponent's serve.

$$Winning\ \%\ Return\ Serve = \frac{Return\ Serve\ Win}{Total\ Return\ Serve} \quad (3)$$

4. *Winning percentage on break point*, this feature represents a chance of the player to get point when he faces the break point game.

$$Winning\ \%\ Break\ Point = \frac{Break\ Point\ Win}{Total\ Break\ Point} \quad (4)$$

5. *Winning percentage of played match*, this feature represents a chance of the player to win the overall matches played.

$$Winning\ \%\ Match\ Played = \frac{Match\ Played\ Win}{Total\ Match\ Played} \quad (5)$$

6. *Total point win*, this feature represents an average of wining point per match.

$$Total\ Point\ Win = \frac{Point\ Win}{Number\ of\ Matches} \quad (6)$$

7. *Hard Court*, this feature represents the match that play on hard court.
8. *Clay Court*, this feature represents the match that play on clay court.
9. *Grass Court*, this feature represents the match that play on grass court.

In the tennis match, it is played between two players (singles) so the input data in 1-6 is needed to have two sets; the data set of player 1 and the data set of player 2 so the input vector consists of 15 parameters.

3 Experiments and Results

To evaluate the proposed method, the high performance computer with the specification of Pentium Core2Duo 2.53 GHz and 2 GB of RAM is used for training MLP. Next subsection describes data managing for our model.

3.1 Data Managing

Clarke and Norton [6] show the way to collect the statistical data which release after the end of the match played so most of tournaments use their techniques to collect the data. The proposed method gets the collected data from the tournaments and manipulates all the data to be the input of MLP.

Assume that Roger Federer and Novak Djokovic have been played only one tournament in the past at the French Open 2008 so the collected data from the tournament could be representing in Table 1 and Table 2. To manipulate these collected data, there are three steps below;

Player1 (*)	Player2	Round	1st Serve Win	Total 1st Serve
Roger Federer	Diego Hartfield	First	39	47
Roger Federer	Fabrice Santoro	Second	32	42
Roger Federer	Janko Tipsarevic	Third	95	107
Roger Federer	Tomas Berdych	Fourth	44	59
Roger Federer	James Blake	1/4	49	65
			259	320

Table 1: The statistical data of Roger Federer at French Open 2008

Player1	Player2(*)	Round	1st Serve Win	Total 1st Serve
Benjamin Becker	Novak Djokovic	First	42	50
Simone Bolelli	Novak Djokovic	Second	34	45
Samuel Querrey	Novak Djokovic	Third	36	48
Lleyton Hewitt	Novak Djokovic	Fourth	45	62
David Ferrer	Novak Djokovic	1/4	44	58
			201	263

Table 2: The statistical data of Novak Djokovic at French Open 2008

- Selects all the historical data of each player. In this step, the data in table 1 and table 2 are the historical data of Roger Federer and Novak Djokovic.
- The value in the 1^{st} *serve win* column and *total* 1^{st} *serve* column are summarized.
- The *Winning Percentage on* 1^{st} *Serve* is calculated by equation (1).

For example, the *summation of* 1^{st} *serve win* of Roger Federer is 259 and the *summation of total* 1*st serve* of Roger Federer is 320. Then, the *winning percentage of* 1^{st} *serve* of Roger Federer is $\frac{259}{320} = 0.81\%$. For Novak Djokovic, the *summation of* 1^{st} *serve win* of Novak Djokovic is 201 and the *summation of total* 1^{st} *serve* of Novak Djokovic is 263. Then, the *winning percentage of* 1^{st} *serve* of Novak Djokovic is $\frac{201}{263} = 0.76\%$. Therefore, other input features are calculated by using the equations above (equation (2) - equation (6)).

3.2 MLP Modeling

3.2.1 Models

This paper proposed three models of MLP which are *StatEnv Model*, *AdvancedStatEnv Model* and *TimeSeries Model*. These three models have different input features.

The input vector of *StatEnv Model* consists of 3 nodes which are winning percentage of played match of player 1, *winning percentage of played match* of player 2 and *court surface*.

The input vector of *AdvancedStatEnv Model* consists of 15 nodes of all input features that explain in Input Feature section.

The *TimeSeries Model* uses the same input as *AdvancedStatEnv Model* to be the first 15 nodes of input vector and use the collected data in the past one year of player which are *Winning percentage on the first serve*, *Winning percentage on the second serve*, *Winning percentage on return serve*, *Winning percentage on break point*, *Winning percentage of played match*, and *Total point win* (the first 6 features that represent in Input Feature Section). Therefore, the input vector of *TimeSeries Model* consists of 27 nodes.

3.2.2 Parameters

To find the suitable MLP model, the learning parameters are adjusted until the error is reduced into acceptable value. The appropriate value of each parameter is shown in the table 3.

Model	Hidden Node	Learning Rate	Momentum
StatEnv Model	20	0.3	0.2
AdvancedStatEnv Model	50	0.3	0.2
TimeSeries Model	150	0.3	0.2

Table 3: The appropriate value of parameters in MLP models.

The *StatEnv Model* has 3 input nodes but *AdvancedStatEnv Model* has 15 input nodes so the hidden node of *AdvancedStatEnv Model* should be increase from 20 nodes to 50 nodes to get the acceptable value of error. Therefore, the *TimeSeries Model* which has 27 input nodes, use 150 hidden nodes. All the numbers of hidden nodes that show in the table come from the experimental.

3.3 Training data

The statistical data and environmental data of match played obtained from *OnCourt* System, www.atpworldtour.com (ATP World Tour), www.australianopen.com (Australian Open), http://2008.rolandgarros.com (French Open), http://championships.wimbledon.org (Wimbledon) and www.usopen.org (US Open).

For the schedule of events in Grand Slam Tournament, the Australian Open is the first event in the year, second

event is French Open, third event is Wimbledon and then US Open is the last event.

The training set of *StatEnv Model* is collected from the year 1990 - 2002. For the *AdvancedStatEnv Model*, the training set is collected from the year 2003 until the year of prediction. For example, if the prediction is Australian Open 2006, the training set is collected from the beginning of the year 2003 to the end of year 2005. *TimeSeries Model* uses the same data as *AdvancedStatEnv Model* and also uses the collected data only in the past one year to be the input data (365 days before prediction).

3.4 Results

3.4.1 StatEnv and AdvancedStatEnv Models

Tournament	Barnett and Clarke Model	StatEnv Model
Australian Open 2003	72.4 %	75.5906 %

Table 4: The accuracy of Barnett and Clarke model and StatEnv Model in Australian Open 2003

The result of Barnett and Clarke model [1] is 72.4% [7]. To compare Barnett and Clarke model [1] with the *StatEnv Model*, the accuracy from *StatEnv Model* is 75.5906% which is more than Barnett and Clarke model.

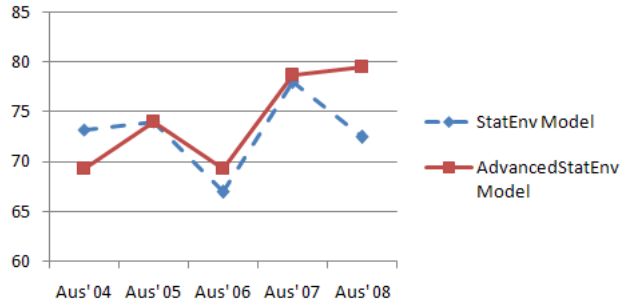

Figure 3. The accuracy of *StatEnv Model* and *AdvancedStatEnv Model* in Australian Open 2004-2008

Tournament	StatEnv Model	AdvancedStatEnv Model
Australian Open 2004	73.2283 %	69.2913 %
Australian Open 2005	74.0157 %	74.0157 %
Australian Open 2006	66.9291 %	69.2913 %
Australian Open 2007	77.9528 %	78.7402 %
Australian Open 2008	72.4409 %	79.5276 %

Table 5: The accuracy of *StatEnv Model* and *AdvancedStatEnv Model* in Australian Open 2004-2008

As the result from figure 3 and table 5, the accuracy of Australian Open 2004 and 2005 from *StatEnv Model* is higher than *AdvancedStatEnv Model* because the training data of *AdvancedStatEnv Model* is not enough for the model. The collected data starts at the year 2003 so there are only data of year 2003 to be the training data for predicting the year 2004. Generally, the *AdvancedStatEnv Model* has more accuracy than the *StatEnv Model* especially in the year of 2008 with the accuracy of 79.5276% because *AdvancedStatEnv Model* uses the collected data in year 2003-2007 to be training set.

3.4.2 TimeSeries Model

Tournament	TimeSeries Model
Australian Open 2007	81.1024 %
French Open 2007	78.7402 %
Wimbledon 2007	80.3150 %
US Open 2007	73.2283 %
Australian Open 2008	80.3150 %
French Open 2008	70.8661 %
Wimbledon 2008	73.2283 %
US Open 2008	77.1654 %

Table 6: The accuracy of *TimeSeries Model*

To compare *TimeSeries Model* with *AdvancedStatEnv Model*, there are 2 tennis events that can be compared which are Australian Open 2007 and Australian Open 2008. As result in Table 5 and Table 6, the accuracy of *TimeSeries Model* is more than the *AdvancedStatEnv Model*. This can conclude that the experience of the player in the past one year directly affect to the prediction results.

4 Discussion

As mentioned in the previous section, the models that based on MLP give the better prediction results than the Barnett and Clarke [4]. Additionally, the comparison between *StatEnv Model* and *AdvancedStatEnv Model* shows that the result of the *AdvancedStatEnv Model* is also more accurate than the *StatEnv Model*. The reason, that leads *AdvancedStatEnv Model* has more accurate results, is the set of input features chosen.

For the *Barnett and Clarke Model*, only two parameters; winning percentage serve of player 1 and winning percentage serve of player 2 are used as input of the model. *AdvancedStatEnv Model* has 7 features of input which provide strongly tennis model for predicting the winner of the match.

Moreover, the *TimeSeries Model* provide more accurate prediction results than other models which are *Barnett and Clarke Model*, *StatEnv Model*, and *AdvancedStatEnv Model* because the *TimeSeries Model* focuses on the experience of players.

One more reason is that most of the current tennis model concentrates only on the statistical data and ignores the environment data which directly affect to the match score. As the result, the court surface is selected to be an additional input parameter for all of *StatEnv Model*, *AdvancedStatEnv Model* and *TimeSeries Model*, which are very well done for predictions.

5 Conclusion

In this paper, the new approach to create the tennis prediction model is shown. To get more accuracy than the current techniques, the Multi-Layer Perceptron is applied to predict the winner of the tennis matches. Three proposed models, which consist of different set of input parameters, are shown that the selection of appropriated parameters extremely affect to the prediction. From comparison among the models, the MLP Model, the appropriated input features, and concentrated on the experience of players in the past one year provide more accuracy than the current tennis models. In the future, the model will be extended to predict the probable length of match that players will face.

References

[1] Barnett T. and Clarke S.R., "Combining player statistics to predict outcomes of tennis matches," *IMA Journal of Management Mathematics*, 16(2), 113-120, 2005.

[2] Pollard G and Barnett T, "Fairer service exchange mechanisms for tennis when some psychological factors exist," *In Proceedings of the Eighth Australian Conference on Mathematics and Computers in Sport*, Coolangatta. J. Hammond and N. de Mestre (eds), 189-198, 2006.

[3] J.G. Kemeny and J.L. Snell, "Finite Markov chains," *Princeton*, New Jersey D. Van Nostrand, 1960.

[4] Barnett T. and Clarke S.R., "Using Microsoft Excel to model a tennis match," *In Proceedings of the 6M&CS*, G. Cohen and T. Langtry eds., 63-68, 2002.

[5] Simon Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd Edition, Prentice Hall , 1999.

[6] S.R. Clarke and P. Norton, "Collecting statistics at the Australian Open tennis championship," *In Proceedings of the 6M&CS*, G. Cohen and T. Langtry eds., 105-111, 2002.

[7] Barnett, Tristan J.; Brown, Alan; Clarke, Stephen R., "Developing a tennis model that reflects outcomes of tennis matches," *Proceedings of the 8th Australasian Conference on Mathematics and Computers in Sport,*, Coolangatta, Queensland, pp. 178-188, 2006.