

Tennis Match Trend Prediction Based on LSTM

Yichen Jiang^{*,#}, Chuansen Yuan[#]

DUT School of Software Technology & DUT-RU International, School of Information Science and Engineering, Dalian University of Technology, Dalian, China, 116620

* Corresponding Author Email: 19159132068@163.com

[#]These authors contributed equally.

Abstract. The focus of this research is on the data processing, including data mining, data cleaning, feature extraction, feature engineering and other aspects. Whether the data can be used to fit the game trend well is the key to the final effect of the model. At the same time, there are many reasons (i.e., characteristics) that affect the trend of the game. The purpose of this study is to find out the most drastic variables and make corresponding tactical changes according to them. The model based on LSTM network constructed in this paper has a good ability to handle variable-length inputs, and uses the attention mechanism to enhance the "memory" of long-distance inputs, and uses the Dropout layer to avoid model overfitting to a certain extent. Finally, the random forest method was used to extract several variables with the highest sensitivity for the sensitivity analysis of the model.

Keywords: Data Mining, Random Forest, LSTM, Attention Mechanism, Sensitivity Analysis.

1. Introduction

1.1. Background

In the 2023 Wimbledon men's singles final, 20-year-old Carlos Alcaraz defeated 36-year-old Djokovic, ending Djokovic's long winning streak at Wimbledon dating back to 2013. In tennis, the dynamics of the game are often altered by so-called momentum, invisible forces that influence the direction of the match and the mindset of the players. In this final, momentum fluctuated particularly dramatically, and we witnessed how it shifted between players to change the dynamics of the game.

Understanding and measuring momentum in tennis matches is inherently challenging. Exactly what causes momentum to be generated and transferred, whether it can be accurately measured, and how players use changes in momentum to their advantage during a game are among the difficult questions that sports scientists and psychologists have tried to answer.

1.2. Literature Review

LSTM was proposed by Hochreiter & Schmidhuber in 1997. After decades of development and iterations by many technicians, it has now formed a systematic and complete LSTM framework, and has been widely used in many fields.

In order to solve the gradient vanishing problem caused by too many LSTM layers, Li JY et al. improved it and proposed Layer Trajectory LSTM [1]. In order to meet certain requirements and sacrifice some accuracy to obtain better true positives, researchers have also proposed a method of combining LSTM network and Bayesian classifier, which can be applied to detect network attacks [2] and other aspects. The recent development of IoT technology has also given rise to the combination of LSTM and CNN, which can well detect security attacks against IoT[3].

Overall, the memory properties of LSTM networks can be borrowed from many fields. We would like to apply it to tennis match prediction to improve the performance of players.

1.3. Marginal Contribution

We tried simple models such as Markov chain, multiple logistic regression, LightGBM model, random forest model, LSTM neural network model, etc., hoping to better fit the data.

Markov chains are commonly used to model objective chains of facts [4]. In tennis, the same applies to us. We want to quantify the "momentum" on the field as objective factors, so that we can model the progression of the game with a Markov chain. However, quantifying "momentum" is harder than we thought.

Multiple logistic regression is used to check whether it is necessary to continue modeling. There is little risk of overfitting using simple models, whose simple linear bounds give a good indication of whether these data are perfectly distributed as random or normal. Logistic regression finally gives an accuracy of around 65%, which suggests that the extended linear additive model will have better performance. Based on this, we simulate the momentum situation on the field each time a score occurs.

LightGBM [5] and Random Forest [6] models are decision tree-based algorithms. We want to use the decision tree clustering algorithm to extract the features that have the most influence on the result. In fact, the feature extraction results of random forests are quite satisfactory. We also use LSTM for model construction and adjust its hidden layer structure. We can think of the hit information for each ball as a bunch of temporal data. As a matter of fact, the mentality of tennis players does change periodically with the duration of the match, and the use of LSTM is also justified.

2. How to deal with the abstract feature: momentum

Based on the information such as the score data of each player during the game, it is estimated whether their momentum performance is good. By quantifying the various factors that affect the player's momentum during the game, and finally calculating the results, the player's momentum can be comprehensively evaluated. The factors that affect the player's momentum include: continuous scoring, winning streak, service initiative, whether the successful break of serve, running distance on the court, the influence of large scores, etc.

By normalizing the data for calculation and using a long short-term memory network to remember the previous state, it is possible to quantify the momentum of the players and see that each player is performing better at a certain point in the game.

Here is a summary of the momentum of the two players in the 2023-Wimbledon 1301 match.

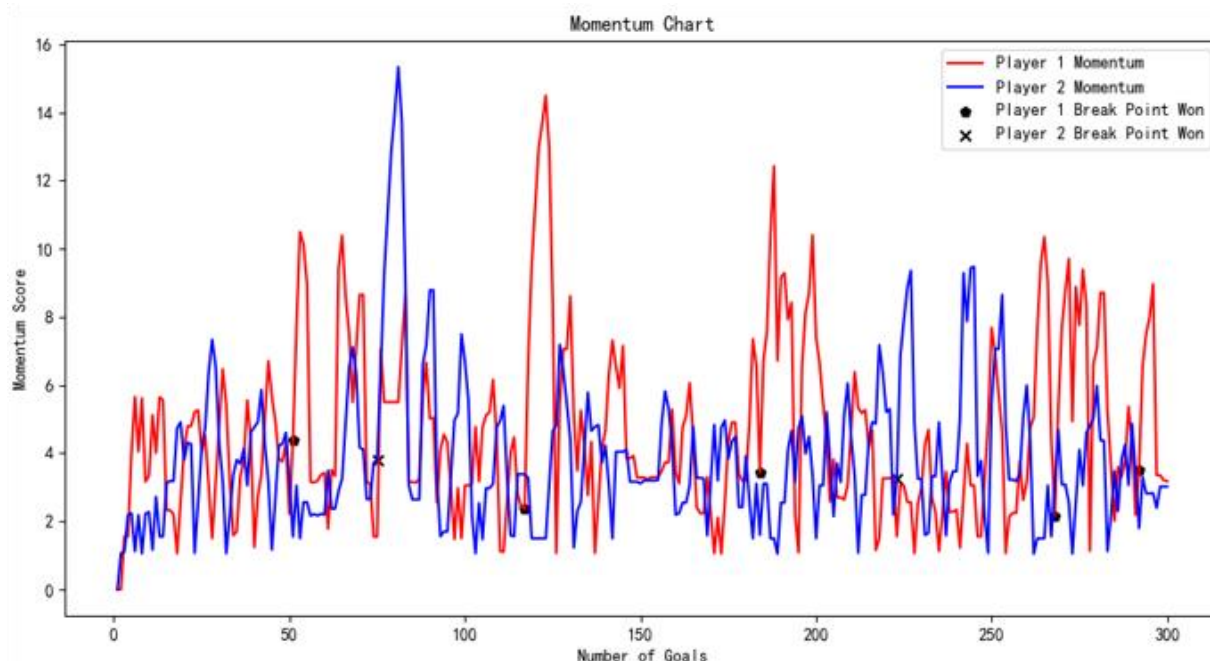


Figure 1. Momentum of two players 1.

It can be seen from Figure 1 that player 1 has a relatively high momentum score at the beginning since he is the first player to serve. Each time an opposing player successfully breaks, Player 1's momentum increases or decreases accordingly.

By artificially setting the lowest point of momentum to 0, it is observed from figure 2 that when a player successfully breaks, their momentum score increases. In addition, the momentum of the player will continue to decrease with the increase of the game time and the distance the player runs.

3. Ensemble learning model Building and solving

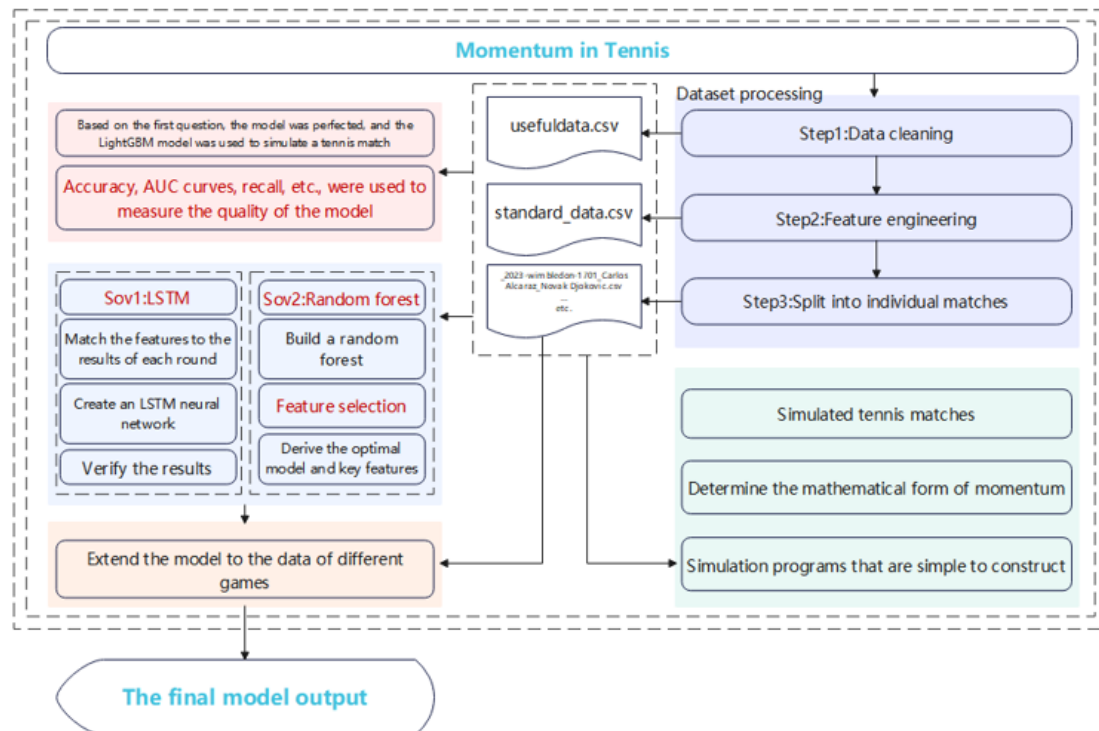


Figure 2. Model concept diagram.

3.1. Collect data

To get the data for this research, we looked at all the matches played at Wimbledon 2023 and came up with a data set that contains information about players on both sides, shots taken, scores scored, and a lot of other information that could affect momentum.

3.2. Build model

Figure 2 shows the conceptual diagram of our model, and our model building can be roughly divided into five steps as shown. Firstly, data cleaning and missing value completion are performed on the obtained data to make the obtained data complete. Then, the data is rearranged in chronological order, and the processed data with time characteristics is used to train several mainstream machine learning models. In the training process, feature engineering is first performed on the processed data to extract useful features, and mathematical operations are performed on the connections between some features and real-world situations, and appropriate cross terms are added. The extracted data is then normalized to reduce model operations and improve model performance. The AUC curve of the model is shown below (two plots for the test and training set) in Figure 3. Taking the best performing LightGBM model on the test set as an example, its prediction accuracy can reach 74%, recall rate can reach 76%, precision rate can reach 74%, F1 score can reach 0.75, and area under the AUC curve can reach 0.83. Thus, there is no doubt that player performance is not completely random, and while good players can control their play trends to be close to random, certain events in the game do affect player momentum, and this effect can be predicted.

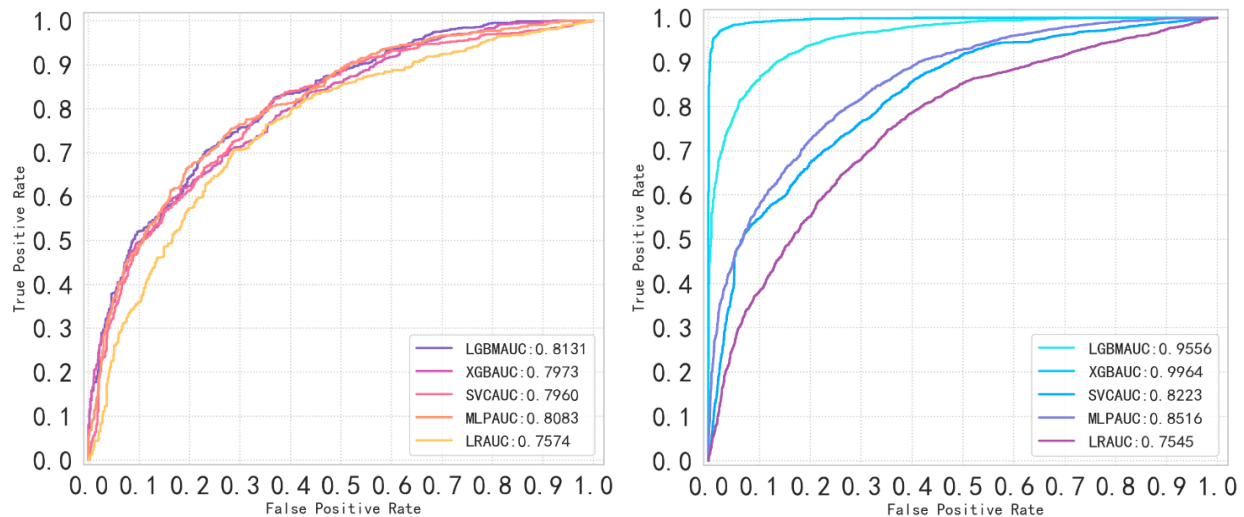


Figure 3. AUC curves for multiple models.

4. Building and solving LSTM models

We will continue to use the data we collected to build an LSTM neural network model. The LSTM neural network is finally selected to predict the probability of winning the game for a given score. The LSTM model can effectively process temporal data. Here are a few types of hidden layers used in this study:

4.1. LSTM layer

Features: The Long Short-Term Memory (LSTM) layer is a special type of recurrent Neural Network (RNN) layer designed to process sequential data, solving the problem of vanishing or exploding gradients in traditional RNNs when dealing with long sequences [7]. It effectively captures long-term dependencies by introducing three gates: forget gate, input gate and output gate to control the retention and forgetting of information.

Influence: LSTM layers can help the model learn long-distance dependencies in the input sequence, which is essential for tasks such as time series analysis and natural language processing. By stacking multiple LSTM layers, the model can learn more complex sequential patterns, but it also increases the computational burden and the risk of overfitting.

4.2. Dropout layer [8]

Features: Dropout layers randomly drop a percentage of neurons (i.e. set the output to 0) during training to prevent the model from overfitting. This amounts to training a different subset of the model in each iteration.

Influence: The Dropout layer reduces the complex co-adaptation between neurons and forces the model to learn more robust feature representations. This usually improves the generalization ability of the model to unknown data, but may lead to less efficient learning or underfitting of the model if the dropout ratio is set too high.

4.3. Dense layer

Features: Each neuron is connected to all the neurons in the previous layer. The Dense layer is mainly used for nonlinear transformation and combination of features.

Influence: By stacking multiple dense layers, and using nonlinear activation functions within them, the model is able to learn complex feature representations. The number of Dense layers and neurons directly affects the capacity of the model [9] (i.e. the ability to learn complex functions). However, too many Dense layers or neurons may lead to overfitting, especially when training data is limited.

4.4. The combined impact of model performance

Superposition of layers: With a reasonable superposition of LSTM and Dense layers, a model that can handle complex sequence data and perform effective feature extraction and classification can be constructed. The LSTM layer is used to capture the temporal dynamics of the sequence, while the dense layer is used to further extract features and make decisions.

Overfitting and generalization: The Dropout layer is used to balance the complexity and generalization ability of the model and prevent overfitting. Proper selection of Dropout ratio and model layer configuration is crucial to achieve a high-performance model.

Parameter tuning: The final effect of the model largely depends on the adjustment of parameters, including the number of LSTM units, the number of neurons in the dense layer, and the Dropout ratio. The choice of these parameters needs to be tuned to the specific task and dataset, and it may be necessary to experiment to find the optimal configuration.

4.5. LSTM architecture

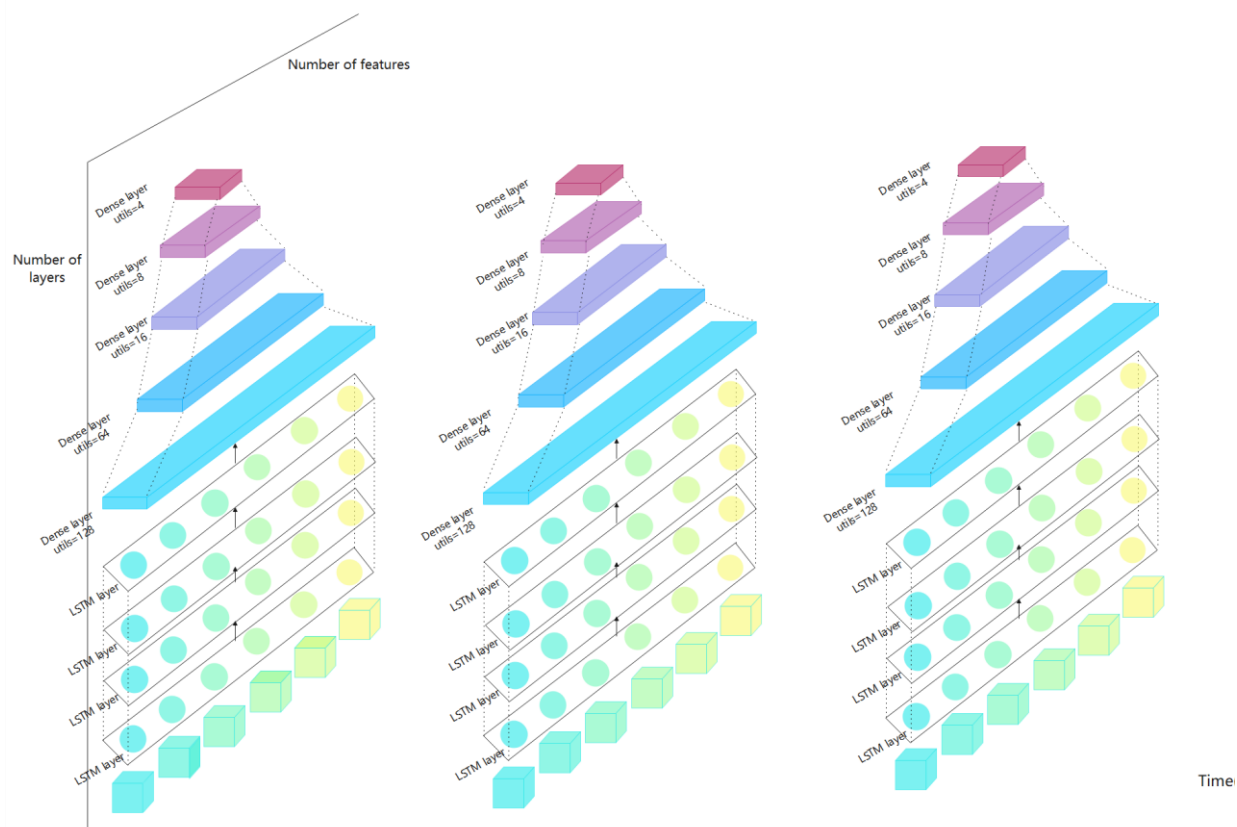


Figure 4. Our LSTM architecture.

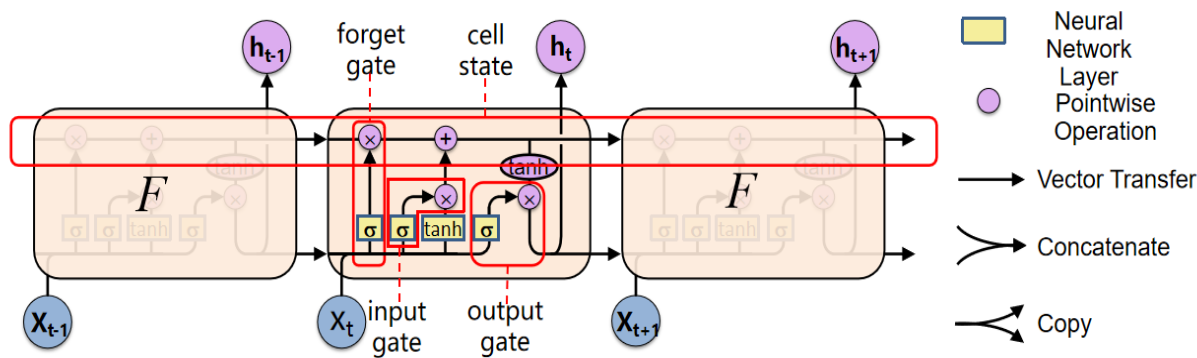


Figure 5. LSTM cell architecture.

Figure 4. Shows the overall architecture of our model, which consists of four LSTM layers, five fully connected layers, and each LSTM cell structure is shown in Figure 5.

In the preliminary data preparation stage, the dataset adjusted by the preprocessing step is first imported and in-depth preparation is performed to ensure that it meets the requirements of the subsequent model training step. Ensuring the proper preparation of the data was the main goal of this phase, for which the first three columns in the dataset were removed after successfully importing the preprocessed data. The exclusion of these columns is based on two considerations: first, they may contain non-numerical information, which is not suitable for further analysis; second, they may carry data that is not critical for the following analysis and model building phases. We then converted the remaining elements of the dataset to float values to ensure consistent processing and a high degree of computational accuracy in subsequent mathematical and statistical analyses. This conversion to a floating-point number provides a solid foundation for subsequent processing of the data.

Next, the dataset is split into training and test sets. This split allows us to train the model on a subset of the data and evaluate the performance and generalization of the model on an independent test set to ensure the effectiveness and robustness of the model.

Then, the data is also normalized before model training in order to reduce the computational complexity of model training and improve the performance of the model. The data attributes are adjusted to the same magnitude to avoid adverse effects on model training due to large differences in the size of the absolute values of the attributes. Normalization helps speed up the model training process and may improve the predictive power of the model on unseen data.

Finally, in order to make the data conform to the input data shape of the LSTM neural network, it is processed into a 3D shape and the label is converted into One Hot encoding format [10]. At the same time, some callback functions are set to monitor the training process of the model, and early stop is set, if there is no significant improvement in the validation set metrics in the SET patient iteration, the training is stopped early and the model weights are restored to the best performing weights. The line chart of the training effect is shown in Figure 6.

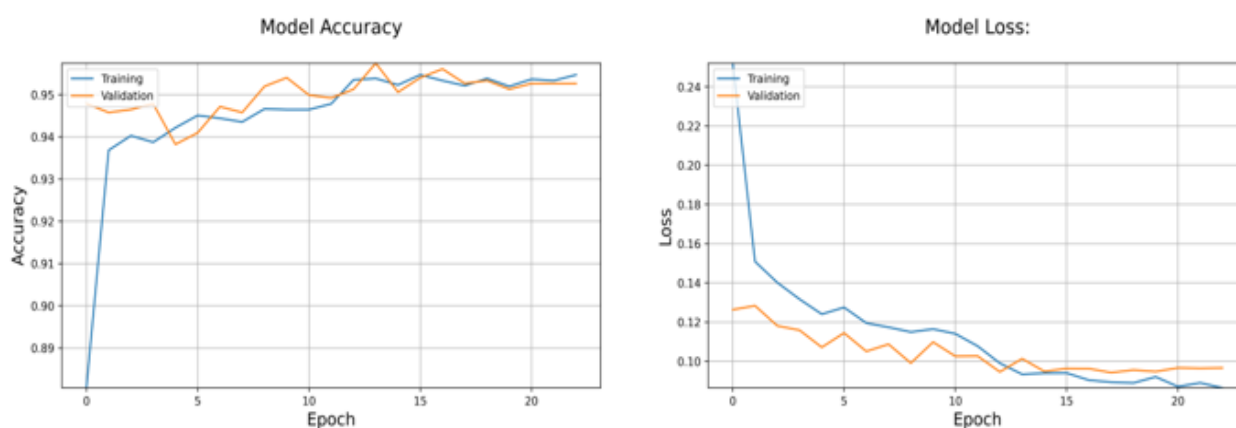


Figure 6. Line plot of training effect.

4.6. Evaluate the model using the test set

Table 1. Model Evaluation Sheet.

	precision	recall	F1-score	support
0	0.95	0.96	0.95	750
1	0.95	0.94	0.95	707
accuracy			0.95	1457
macro avg	0.95	0.95	0.95	1457
weighted avg	0.95	0.95	0.95	1457

Table 1 shows the evaluation sheet for model training, it can be confirmed that it is possible to be effective after knowing the base case for a certain score.

4.7. The importance of relevant factors

Random forests are used to determine the relevance of all features. The result is shown in Figure 7.

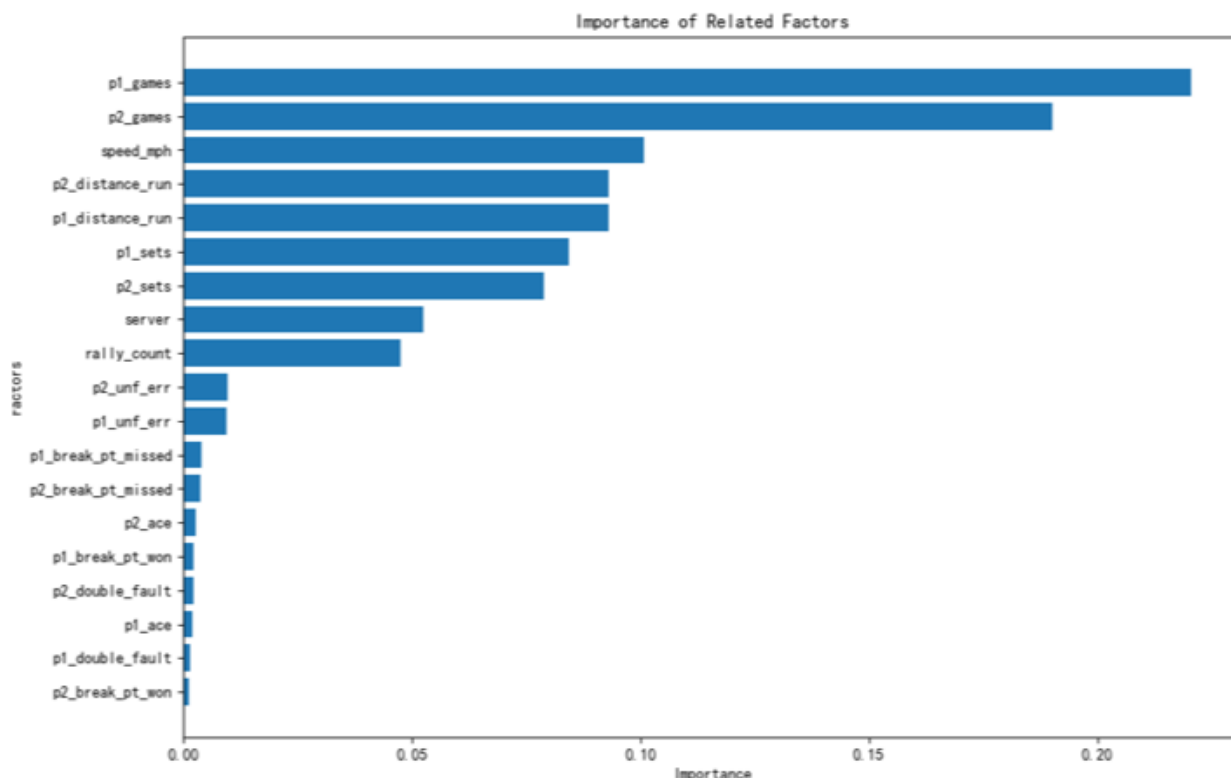


Figure 7. The importance of relevant factors.

5. Advantages and disadvantages of LSTM model

5.1. Pros

LSTM is a variant of RNN with strong memory capability [11], which is suitable for modeling and prediction of processing sequence data. By introducing a gating mechanism, LSTM can effectively capture and process long-term dependencies [12], and perform well in tasks that require memorizing and understanding contextual information. In tennis, the previous ball is likely to affect the subsequent performance of the player, so it is advantageous to use LSTM. LSTM can process various types of input and output, including sequence data, time series data, multi-dimensional data, etc., and can make single-step or multi-step predictions. This multi-step prediction can conveniently provide a prediction of the current situation on the field for future game results.

5.2. Cons

The gate control mechanism in LSTM increases the complexity and computational complexity of the model, resulting in slower training and inference, especially when dealing with large-scale data. There are multiple hyperparameters that need to be tuned in LSTM, such as the number of hidden units, learning rate, etc. Therefore, in practice, tuning the parameters requires a certain amount of time and experience. Due to the strong memory ability of LSTM, it is prone to overfitting when the training data is insufficient or the model complexity is too high. It is necessary to employ appropriate regularization methods to alleviate overfitting.

6. Conclusions

After research, it is found that the LSTM model provides the most satisfactory results, and LightGBM and random forest also perform reasonably well and can be used as complementary models. After the analysis of random forest method, several variables with the strongest correlation can be obtained. Through the control of these variables, coaches can help players to perform better to the greatest extent.

The LSTM model can deal with the problem of tennis momentum, and its excellent processing ability of time-dependent input makes it have better performance than other deep learning models on this problem. The addition of attention mechanism [13] can further improve the performance of the model. At the same time, this model has the potential to be extended to most sports, and applying this research to sports in the future can greatly improve the performance of athletes and improve the enjoyment of games. At the same time, it provides a possible solution for another similar research.

References

- [1] Li JY, Liu CL, Gong YF. Layer Trajectory LSTM[j]. 19TH Annual conference of The International Speech Communication Association (INTERSPEECH 2018), 2018, 1-6:1768-1772.
- [2] Li Y, Lu YF. LSTM-BA: DDoS Detection Approach Combining LSTM and Bayes[j]. 7th International Conference on Advanced Cloud and Big Data (CBD), 2019, 180-189.
- [3] M. S. Alshehri, J. Ahmad, S. Almakdi, M. A. Qathrady, Y. Y. Ghadi and W. J. Buchanan, SkipGateNet: A Lightweight CNN-LSTM Hybrid Model with Learnable Skip Connections for Efficient Botnet Attack Detection in IoT[j], IEEE Access, 2024, 12:35521-35538.
- [4] Aziz Abdulla P, Ben Henda N, Mayr R. Decisive Markov Chains [J]. arXiv e-prints, 2007: arXiv: 0706.2585.
- [5] Liang YX, Wu JY, Wang, W, Cao, YJ, Zhong BL, Chen ZK, Li ZZ. Product Marketing Prediction based on XGboost and LightGBM Algorithm [J]. 2019 2ND International Conference on Artificial Intelligence and Pattern Recognition (AIPR 2019), 2019, 150-153.
- [6] Utkin LV, Kovalev MS, Meldo AA, Coolen FPA. Imprecise Extensions of Random Forests and Random Survival Forests [J]. Proceedings of the Eleventh International Symposium on Imprecise Probabilities: Theories and Applications (ISIPTA 2019), 2019, 103:404-413.
- [7] Hu, Xiaojian, Liu, Tong, Hao, Xiatong, Lin, Chenxi. Attention-based Conv-LSTM and Bi-LSTM networks for large-scale traffic speed prediction [J]. Journal of Supercomputing, 2022, 78 (10):12686-12709.
- [8] Guo J, Qi L, Shi Y, et al. PLACE Dropout: A Progressive Layer-wise and Channel-wise Dropout for Domain Generalization [J]. ACM Transactions on Multimedia Computing, Communications and Applications, 2023, 20(3): 1-23.
- [9] Xie Y, Liang RY, Liang ZL, Zhao L. Attention-Based Dense LSTM for Speech Emotion Recognition [J]. IEICE Transactions on Information and Systems, 2019, E102D (7):1426-1429.
- [10] Bhattacharya S, Maddikunta PKR, Meenakshisundaram I, Gadekallu TR, Sharma S, Alkahtani M, Abidi MH. Deep Neural Networks Based Approach for Battery Life Prediction [J]. CMC-Computers Materials & Continua, 2021, 69(2):2599-2615.
- [11] Sari WK, Rini DP, Malik RF. Text Classification Using Long Short-Term Memory [J]. 2019 3RD International Conference on Electrical Engineering and Computer Science (ICECOS 2019), 2019, 150-155.
- [12] Yang HF, Hu JJ, Cai JH, Wang YP, Chen X, Zhao X, Wang LL. A New MC-LSTM Network Structure Designed for Regression Prediction of Time Series [J]. Neural Processing Letters, 2023, 55(7):8957-8979.
- [13] Zeng Z, Jiang Z, Men A, et al. A plug-and-play LSTM-based attention module for person re-identification[C]//Eleventh International Conference on Graphics and Image Processing (ICGIP 2019). SPIE, 2020, 11373: 83-89.