
MM-LSTM: A Momentum LSTM Model for Tennis Match Prediction and Analysis

Summary

Tennis is a widely acclaimed sport, and during matches, incredible swings often occur due to "momentum" which cannot be directly observed. To investigate the existence of this "momentum" and predict its trends, we utilize our **innovative MM-LSTM model** to model players' match states.

For problem 1, we analyze the **Spearman correlation coefficients** among various parameters in tennis matches and dimensionally reduce the data based on these correlations. We employ the **Momentum-Long Short Term Memory (MM-LSTM) model** to model the **sequential variation** in the match. Its Cell state, Input Gate, Forget Gate, and Output Gate are used to simulate the player's match status, observations of the current match situation, forgetting past match situations, and the action of hitting the ball, respectively. Based on its output, we can judge the player's performance. The model's performance is validated on the test set, achieving a loss value of **0.047** and an impressive accuracy of **94.52%**.

For problem 2, we first conducted a **forward derivation, disregarding the momentum** and considering the impact of various factors on the predictive probability, thus establishing a **stochastic probability model**. Then, using the MM-LSTM model as a starting point, we **reverse-explain the rationale** behind the stochastic probability model. We tested these two models on the same dataset. In the final results, the MM-LSTM model achieved a significantly higher prediction accuracy than the stochastic probability model, with an **improvement of 26 percentage points at 94.23%**. This clearly illustrates the importance of momentum.

For problem 3, we delve beyond the surface of the MM-LSTM model, discovering a subtle relationship between **the gradient of the function** representing the relative score difference of athletes and their relative momentum and use it as an indicator representing the progress of the game. We considered the relationship between score changes and predicted probabilities, and established a bridge to represent momentum through **probability transition**. Taking more factors into account, we conducted a **gradient importance analysis** for all input factors. By calculating the partial derivatives of the input to the output, we determined the impact of various indices on momentum. Based on this, we proposed **5** effective recommendations.

For problem 4, we compare the MM-LSTM model's probability output **predictions** with the **actual** values of the mentioned tennis match, achieving an accuracy of **90.71%** and **visualizing the fitting curve** between predicted and real values. Additionally, we expand the model to include factors from both **subjective and objective** perspectives, **such as environmental conditions and psychological traits**. Lastly, we collect 2024 WTT table tennis match data and evaluate the **generalizability** of our MM-LSTM model to **table tennis**, finding that the model's performance did not match that of tennis due to the smaller dataset and insufficient description of match states. Increasing the amount of data will bring more improvement in model performance.

Finally, we draft a memorandum summarizing our research findings, demonstrating the role of momentum in matches for coaches, and providing detailed advice for players on handling momentum.

Keywords: Momentum LSTM Gradient

Contents

1	Introduction	2
1.1	Problem Background	2
1.2	Problem Restatement	2
1.3	Our Work	3
2	Assumptions and Notations	4
2.1	Assumptions and Justifications	4
2.2	Notations	5
3	Problem 1: Time Series Match Prediction Based on Long Short-Term Memory Network	6
3.1	Data Preprocessing	6
3.2	Model Structure	8
3.3	Training and Predicting Results	9
4	Problem 2: Momentum Modeling and Evaluation of Long Short-Term Memory Networks (MM-LSTM)	10
4.1	Stochastic Probability Based Winner Prediction	10
4.2	An LSTM Model is Equivalent to a Momentum-Based Model	12
4.3	Comparison of Stochastic Probability Model and Momentum-Based Model	13
5	Problem 3: Momentum Shift Predictor Based on MM-LSTM	13
5.1	Model Structure of the Momentum Shift Predictor	13
5.2	Factor Contribution Analysis	15
5.3	Match recommendations based on momentum changes	16
6	Problem4: Validation and Expansion of MM-LSTM	18
6.1	Prediction of Momentum Swings	18
6.2	Future Model Influence Factors Analysis	20
6.2.1	Objective Factors	20
6.2.2	Subjective Factors	20
6.3	Testing the generalizability of the model on table tennis matches	21
7	Model Analysis	21
7.1	Strengths and Weaknesses	21
7.2	Sensitivity Analysis	22
8	Memorandum	23

1 Introduction

1.1 Problem Background

Tennis has always been a sporting event that is watched by the masses and in the men's final of the 2023 Wimbledon Tennis Championships, 20 year old Spanish star Carlos Alcaraz defeated 36 year old Novak Djokovic in a tightly contested match. At the start of the match, Djokovic showed overwhelming power that seemed to signal an easy victory. But the match took a surprisingly volatile turn during the course of the match. In the end it was Carlos Alcaraz, who was not favored at the start of the match, who emerged victorious.

1.2 Problem Restatement

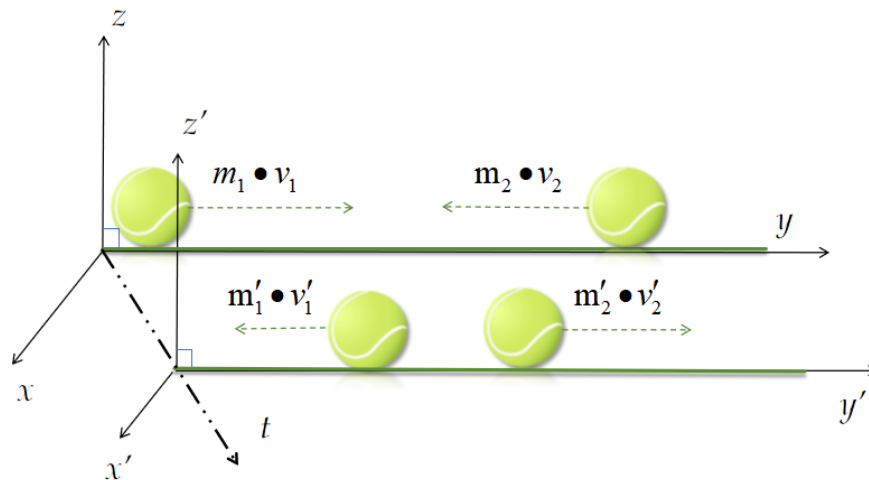


Figure 1: The Momentum of Tennis

Match swings are often linked to a player's momentum — the power or advantage gained. It's unclear how this momentum is generated or altered, prompting an examination of its role in matches and its predictive potential.

Given data from the 2023 Wimbledon Gentlemen's matches, including match ID, players' names, and elapsed time, we aim to use momentum observations to address the following problem:

- Establish a model capable of analyzing player performance during a match, based on which the match process can be visualized and logically represented.
- Establish a traditional random probability model for comparison, assessing the role of momentum in the match.
- Analyze factors influencing turning points in a match from the following two perspectives:
 - Develop a model for predicting match swings to analyze influencing factors.
 - Provide players with recommendations to improve their chances of winning.

- Evaluate the established predictive model for match swings based on the following two criteria:
 - Analyze the accuracy of swing predictions, considering factors that may contribute to improving the model's accuracy when performance is subpar.
 - Consider adding additional factors to enhance the model's generalizability.

1.3 Our Work

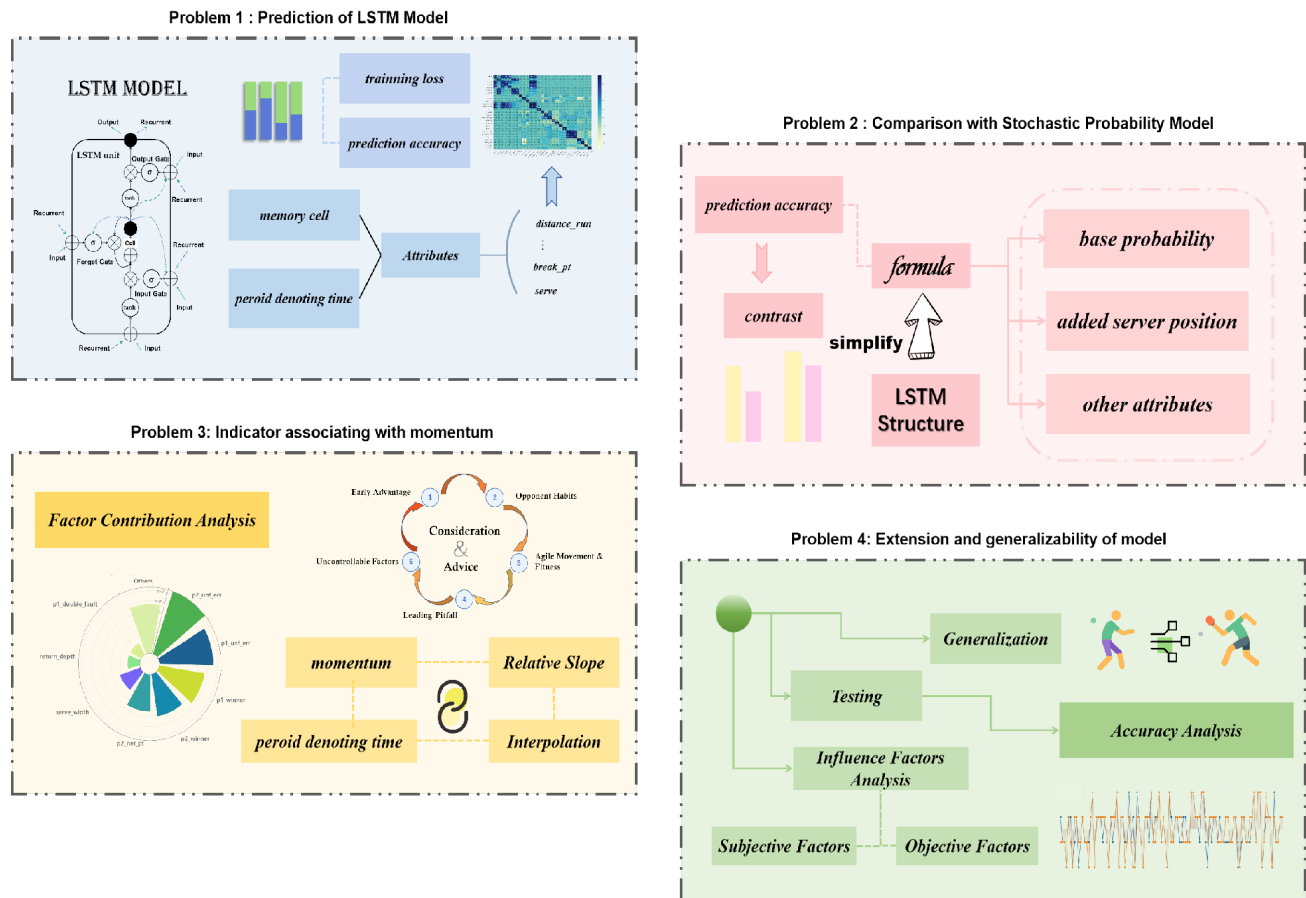


Figure 2: Overall Architecture Diagram

In problem 1, we select LSTM as the foundational model based on the condition of momentum changing over time, focusing on observing the temporal characteristics of momentum. Firstly, we digitize the string values in the given data, followed by normalization to remove scale differences. Subsequently, we analyze the correlations among various indicators, selecting those with relatively strong independence as inputs for the MM-LSTM model. Additionally, we divide the dataset into training and testing sets. After about 6 hours training, we obtain a highly accurate model output.

In terms of problem 2, we conduct predictive calculations using a random probability model and compare it with the model in problem 1. We begin by analyzing the principles of the LSTM model, extracting its core formulas, and then abstracting and simplifying these formulas to obtain a computationally simplified traditional probability model. This approach reduces computational

complexity while maintaining calculability. In this model, we further reduce the number of influencing factors. Despite obtaining lower probabilities, the deviation from the estimated true probabilities is relatively small. This also underscores the significance of momentum in determining match scores.

After our analysis of the problem 3 prompt, in contrast to choosing prediction approaches involving other complex factors, **we dissect the subtle momentum changes hidden within the historical win-loss dynamics**. Through precise mathematical derivations, we ultimately **establish the relationship between the relatively differential slope and momentum, uncovering the reasons for momentum changes and forecasting conditions**. Simultaneously, we conduct gradient importance analysis during training to identify the primary factors influencing accuracy. Based on these analyses and computations, we propose recommendations for athletes based on momentum. We compare momentum swings during the start and throughout the match, perform calculations related to physical factors, and ultimately summarize and **present five effective suggestions**.

In addressing problem 4, we initially employ the results predicted by the MM-LSTM model to generate momentum swings images, **achieving an accuracy exceeding 90% and confirming the model's excellence**. Simultaneously, we expand on the original factors, conduct literature reviews, and **analyze other potential factors** from both subjective and objective perspectives. Additionally, we collect table tennis match data from online sources and apply the established MM-LSTM model to predict and analyze table tennis match data, **yielding favorable outcomes**.

2 Assumptions and Notations

2.1 Assumptions and Justifications

To simplify the model, we make the following assumptions:

Assumption 1: Regarding influencing factors, we assume that we do not consider factors other than those providing data that may affect the results and model. We only consider direct factors, and these factors are assumed to be independent of each other. Due to the complexity of factors influencing momentum in real life, even though more than 40 factors have been listed, there is a possibility that other factors could have a decisive impact on the final results.

Assumption 2: For the composition of momentum, we assume that it has properties similar to physical momentum. We consider the momentum discussed in this paper to be essentially equivalent to the sum of psychological and physical effects. Since the athlete's inherent abilities do not change during the match, the most important influencing factor for their performance is assumed to be psychological. Both this type of momentum and physical momentum in physics can be seen as the combined results of multiple factors, with varying degrees of influence.

Assumption 3: For the data itself, we assume that the provided data is accurate and reliable. Additionally, the data collected from other matches with reference links from the internet is assumed to be true and trustworthy. During model generalization, we used data from table tennis matches for training and comparison.

Assumption 4: Regarding symbol and constant replacements during the model formula simplification process, we assume that they are not considered, even though this may result in around a 10% error in the results. This operation is to simplify the complexity of model calculations. Initially, we did not simplify the model during the initial calculations, which resulted in longer training times. In subsequent calculations, various factors had a significant impact.

Assumption 5: For the athletes in the model, we assume that the influencing factors for their performance mainly consider recent matches, and we do not consider the impact of previous matches. This allows our model to remain closed for a period of time on the dataset, improving its stability and reducing the consideration of irrelevant factors.

Assumption 6: For time changes, we assume that the time for each score is average, without considering the time interval of each score. This allows us to quantify time and better consider changes in various attributes.

2.2 Notations

Symbol	Definition
A	Replace the function related to the memory cell in the formula of the LSTM
B	Replace the coefficient function before the tanh function in the LSTM
b_i	Bias in the input mapping
I_t	Function of the input gate
O_t	Function of the output gate
C_t	Function of the memory state
\tilde{C}_t	Function of the candidate memory state
F_t	Function of the forget gate
h_t	The value of the hidden state.
k_i	The slope of the function representing the change in scores over a period of time
m_i	the strength value of player
P_i	The probability of scoring in the i-th attempt
x_t	Inputs associated with the variation in time
ρ	The Spearman's rank correlation coefficient
N_{false}	The number of errors in probability statistics
N_{true}	The number of correct outcomes in probability statistics
N_{total}	The total number of outcomes in probability statistics
P_{base}	the win rate value of a player base, set to a fixed value of 0.5
P_{server}	the portion of a player's win rate that is increased over the base win rate as a server
R_{server}	The probability of a server's advantage obtained through statistical analysis
$x_{\text{normalized}}$	The normalized inputs
η_{accuracy}	The statistically obtained prediction accuracy
$\sigma(x)$	The sigmoid activation function
$F_i(x)$	Referring to an activation function and mapping composition operation
$R(x)$	The rank order of a variable
$K_{\text{relative}}(n, m)$	The mean of the relative differences in slope during the time interval from m to n

3 Problem 1: Time Series Match Prediction Based on Long Short-Term Memory Network

3.1 Data Preprocessing

Within the dataset provided by MCM, there exists a comprehensive collection of 46 attributes pertaining to tennis matches, comprising 38 numerical attributes and 8 categorical ones.

The following is the data processing method we adopted:

- We numerically encoded 4 of the categorical attributes: winner_shot_type, serve_width, serve_depth, and return_depth.
- Given the convention in tennis scoring where "AD" signifies an advantage, which is challenging to quantify, we use the fields "p1_score" and "p2_score" as equivalent numerical representations of the scores.
- For handling missing values in numerical data and ensuring the continuity of the data, we use the Lagrange interpolation method. For a given set of $n + 1$ data points, the Lagrange interpolation polynomial is given by:

$$P(x) = \sum_{i=0}^n f(x_i) \cdot L_i(x) \quad (1)$$

where $L_i(x)$ represents the Lagrange basis function.

- There are still 42 fields remaining, except for the four fields match_id, player1, player2, and elapsed_time. Due to the large number of fields, we conducted dimensionality reduction on the data.
- Because of the non-uniformity of data dimensions, it is not possible to observe the degree and trend of changes in various attributes simultaneously. Therefore, we performed Min-Max Normalization processing.

$$x_{\text{normalized}} = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (2)$$

Additionally, before dimensionality reduction, we need to calculate the correlation between each field, and we choose to calculate the Spearman rank correlation coefficient between each field to measure the monotonic relationship between two variables, which not only takes into account the linear relationship between the variables, but also takes into account their rank relationship, which is more robust to data that do not satisfy the assumption of normal distribution. The formula for the Spearman's rank correlation coefficient is as follows:

$$\rho = \frac{\sum_{i=1}^n \left(R(x_i) - \overline{R(x)} \right) \cdot \left(R(y_i) - \overline{R(y)} \right)}{\sqrt{\left(\sum_{i=1}^n \left(R(x_i) - \overline{R(x)} \right)^2 \right) \cdot \left(\sum_{i=1}^n \left(R(y_i) - \overline{R(y)} \right)^2 \right)}} \quad (3)$$

In this equation, ρ denotes the Spearman rank correlation coefficient, n is the number of data points, $R(x_i)$ and $R(y_i)$ denote the ranks of variables x and y , respectively, and $\overline{R(x)}$ and $\overline{R(y)}$ denote the means of the ranks of x and y , respectively.

The Spearman's rank correlation coefficient analysis results between any two fields take values between -1 and 1. The closer the value is to 1 or -1, the stronger the correlation between the two fields. Conversely, the closer the value is to 0, the weaker the correlation between the two fields. A correlation heatmap between some fields, calculated based on Spearman's rank correlation coefficient, is shown in Figure 3.

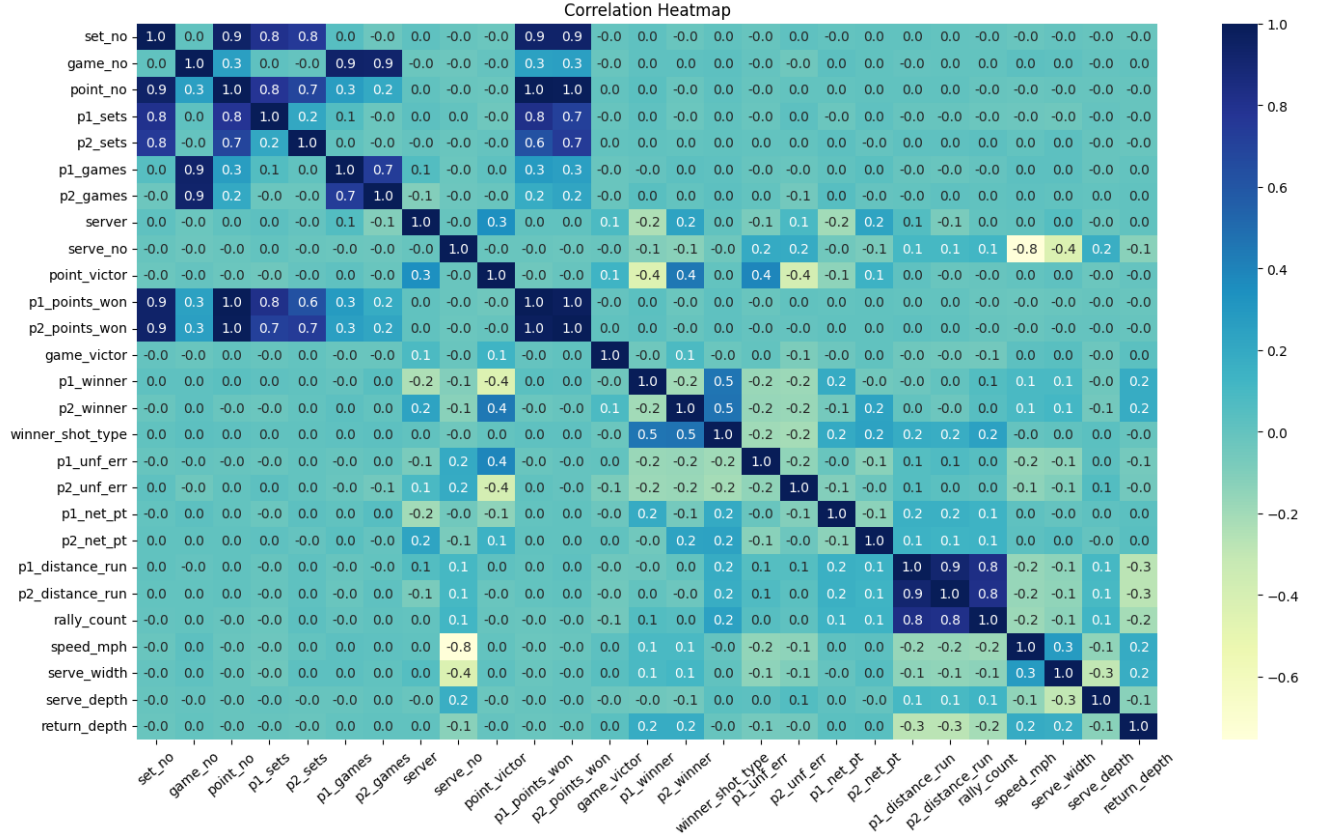


Figure 3: Correlation Heatmap Based on Spearman Correlation Analysis

According to the correlation heatmap, the correlation coefficients between the distance run by the two players in the match ("p1_distance_run", "p2_distance_run") and the number of strokes in the match ("rally_count") are ≥ 0.8 , which indicates that there is a strong correlation between them. It can be understood that the more the number of hits in the game, the longer the distance run in the game, so we only keep the number of hits in the game ("rally_count") field, and delete the distance run by the two players in the game ("p1_distance_run", "p2_distance_run"), so as to achieve the effect of the data dimensionality reduction. Based on this kind of correlation analysis, we did dimensionality reduction on the data, extracted some fields that are more independent from each other, and finally extracted 27 fields for subsequent model training.

3.2 Model Structure

On a real field of play, the game situation has the following insight into the impact of the game on the player: the closer the point is to the current game, the greater the impact on the player.[1] Therefore, to simulate this effect, we modeled a long and short-term memory network based model to capture the flow of points occurring while the game is in progress.

Long Short-Term Memory (LSTM) is a special kind of Recurrent Neural Network (RNN), widely used in the field of Natural Language Processing. The core idea of LSTM is its so-called "Cell state" and its interaction with three gate controllers: Input Gate, Forget Gate, Output Gate. During the training process, the network learns the parameters of these three gate controllers dynamically. Through the collaboration of the three gates, the LSTM can selectively remember or forget information, and therefore can efficiently process sequences of different lengths. Figure 4 shows the exact structure of the LSTM model we built.

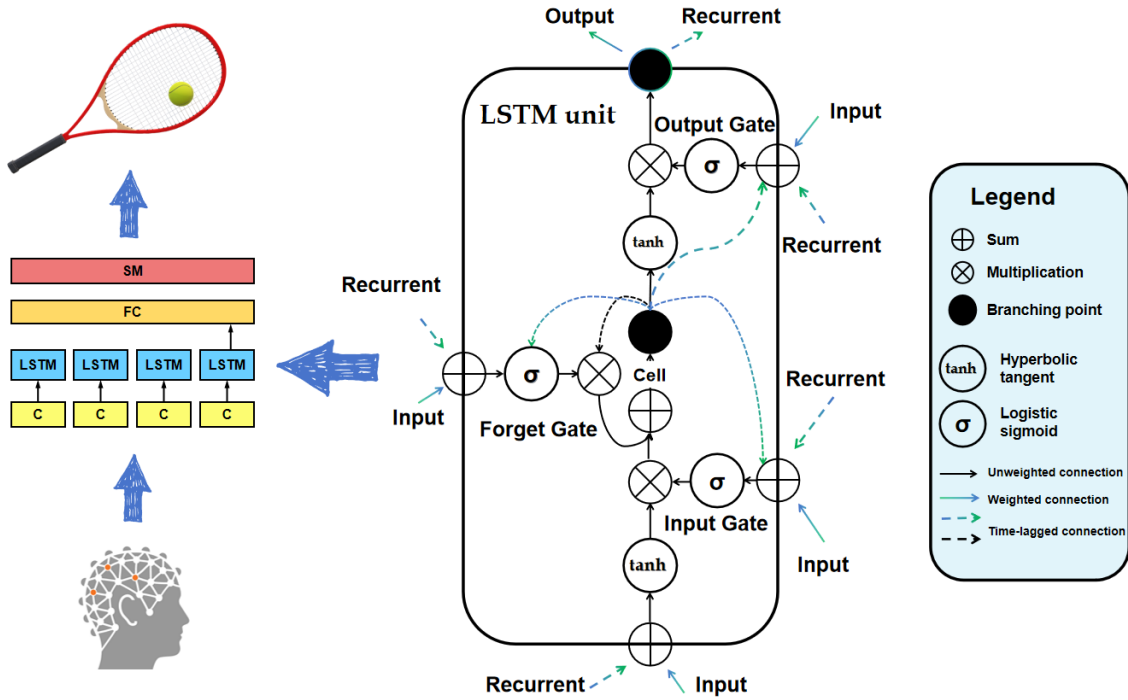


Figure 4: The Structure Diagram of LSTM

We employ LSTM to simulate players' behavior during a match. In this simulation, the Cell state in LSTM represents the player's current match status, the Input Gate simulates the player's reception of current changes on the field, the Forget Gate simulates the player's forgetting of previous match conditions, and the Output Gate simulates the actions taken by the player based on the current match status. Additionally, we construct a sequence of dimensionally reduced point data from a match, which serves as the input to the model and is denoted as x_t . Specifically:

- Input Gate determine what new information will be added to the unit state, similar to how an player adjusts his or her tactics and techniques in a game based on the current round and the opponent's strategy. We denote the input gate as I_t . The expression of I_t is:

$$I_t = \sigma(x_t \cdot k_1 + h_{t-1} \cdot k_2 + b_1) \quad (4)$$

- Forget Gate determines which information in the unit state should be discarded, similar to how a player ignores or forgets the favorable and unfavorable factors of previous rounds in a game in order to focus on the current challenge. In the current match, information that is further away from the present is more likely to be discarded by the oblivion gate, and their impact on the current situation gradually diminishes. This is similar to the human brain's forgetting curve. We denote the forget gate as F_t . The expression of F_t is:

$$F_t = \sigma(x_t \cdot k_3 + h_{t-1} \cdot k_4 + b_2) \quad (5)$$

- Output Gate determine which parts of the unit's state will be used to generate the current output, similar to how a player decides their next move based on the current round and previous experience. We denote the output gate as O_t . The expression of O_t is:

$$O_t = \sigma(x_t \cdot k_5 + h_{t-1} \cdot k_6 + b_3) \quad (6)$$

During the training process, the Cell state is updated based on the Input Gate and Forget Gate, after which the Output Gate forms the output based on the value of the Cell state.

3.3 Training and Predicting Results

We utilize the dataset provided by MCM, allocating 90% of the data for model training and the remaining 10% for testing. After iterating through 1000 training epochs, the model's loss function values are presented in Figure 5. The results indicate that our model achieves an accuracy rate of 94.52% on unseen data, demonstrating its robust generalization capabilities to accurately predict players' scores in real match scenarios.

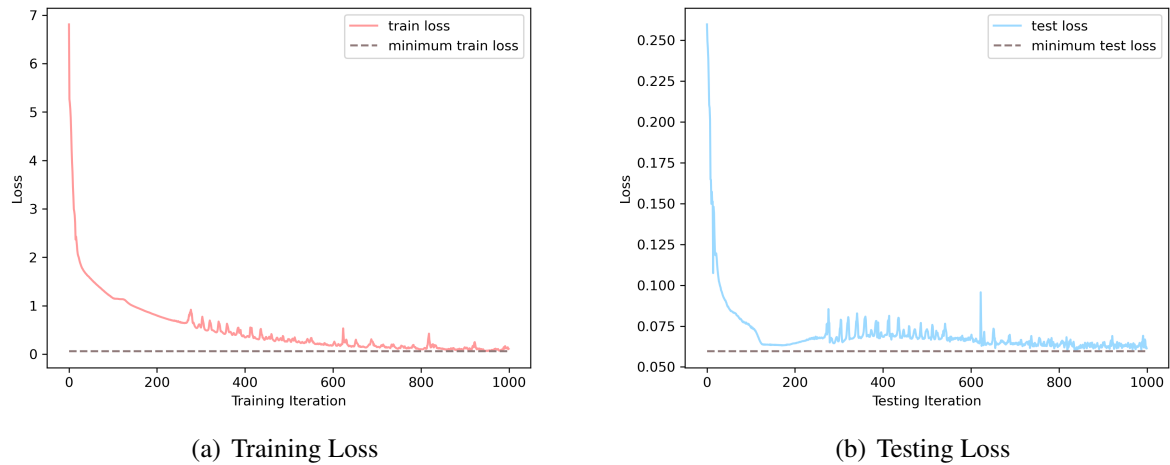


Figure 5: The Loss Curve for MM-LSTM

Based on this, we visualized the match process between Carlos Alcaraz and Novak Djokovic (whose data was not used for training) as shown in Figure 6. The X-axis represents the flow of the match, with the blue section representing Carlos Alcaraz and the green section representing Novak Djokovic. We

used the area of the bar chart to represent the performance score of the player at the current moment, with both players' performance scores ranging between 0 and 1, summing up to 1. The player with the higher performance score at each moment is predicted to be the winner of the game.

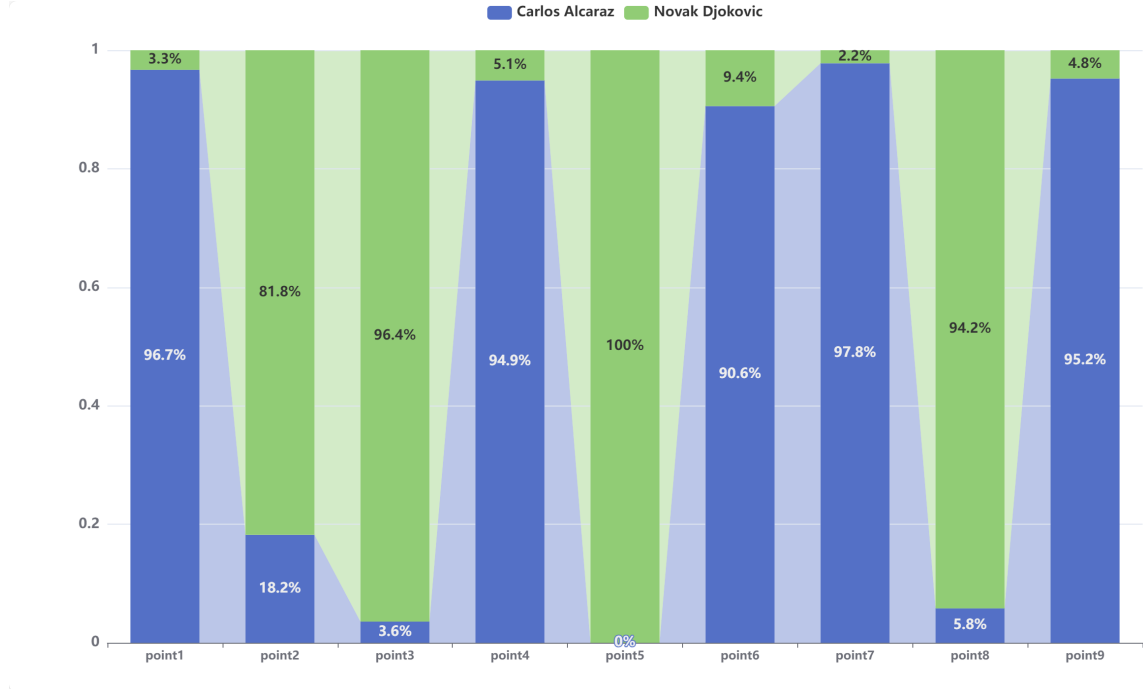


Figure 6: Match Flow Visualization: Carlos Alcaraz vs. Novak Djokovic

4 Problem 2: Momentum Modeling and Evaluation of Long Short-Term Memory Networks (MM-LSTM)

4.1 Stochastic Probability Based Winner Prediction

In order to assess whether momentum plays any role in the match, we first develop a stochastic probability-based prediction model of winning percentage. We choose three empirical metrics to parameterize this stochastic probability model:

- P_{base} : the win rate value of a player base, set to a fixed value of 0.5.
- P_{server} : the portion of a player's win rate that is increased over the base win rate as a server.
- m_i : the strength value of $player_i$.
- $P_{factors}$: the probability change value caused by factors.

The computation of P_{server} and m_i is described below in turn:

For P_{server} , we analyzed the win rates of players as servers based on the data set of Wimbledon 2023 Gentlemen's singles matches after the second round provided by MCM. It was found that the

probability of winning as a server decreases over the different sets of a match, suggesting that the effect of the serve on a player's win decreases as the match progresses. Also based on the analysis of the paper by researchers at the University of Amsterdam [2], we integrate both and decide to only consider the impact of the serving position in the first two games.

We have done a statistical analysis of the probability of a server winning in the data set, and the base value of the increase in the probability of a player winning by serving is 0.173119165. Based on this, we modeled the P_{server} formulation as follows:

$$R_{server} = 0.173119165 \quad (7)$$

$$P_{server} = \begin{cases} 2 \cdot R_{server}, & \text{set_no} = 1 \\ R_{server}, & \text{set_no} = 2 \\ 0, & \text{set_no} > 2 \end{cases} \quad (8)$$

Similarly, for other factors, we also use similar methods for analysis, and ultimately use the following formula to calculate:

$$\sum P_{factors} = P_{server} + \sum P_{others} \quad (9)$$

For m_i , we take the ratio of the final scores and other factors of two players in a game as the relative ratio of their strengths, based on which we calculate the strength intensity of each player as the value of m_i . The specific steps are as follows:

1. Initialize the player list, set two attributes for each player: relative_strength, strength_ratio. relative_strength is a list of the relative strength of the players, set its initial value to 100; strength_ratio Indicates the relative ratio of strength between the player as player1 and the corresponding player2.
2. Iterate through Wimbledon_featured_matches.csv, get the player1, player2, p1_point_won and p2_point_won at the end of each match. For each player1, the value of p1_point_won divided by p2_point_won is recorded in strength_ratio as its strength relative to player2.
3. Iterate through the list of players and get the relative_strength of the current player, recorded as player1_strength. Calculate the player2_strength of the corresponding player2, whose value is equal to player1_strength * ratio. After that, add the calculated play2_strength to the relative_strength list of player2.
4. Iterate through the list of players, and if the player's relative_strength has more than one value, take the average of the values and use it as the new relative_strength.
5. Scales the relative_strength of each player to a value between 0 and 100.

Based on the obtained P_{server} and m_i above, we model the probability of player1 winning under stochastic probability as follows:

$$P_1 = \frac{(P_{base} + \sum P_{factors}) \cdot m_1}{(P_{base} + \sum P_{factors}) \cdot m_1 + (P_{base} - \sum P_{factors}) \cdot m_2} \quad (10)$$

4.2 An LSTM Model is Equivalent to a Momentum-Based Model

To illustrate the correlation between our LSTM model and momentum, we simplify the modeled LSTM model into a well-formed momentum model based on the following equations.

Our LSTM model is formulated as follows, where I_t , F_t , O_t , C_t represent the input gate, forget gate, output gate, Cell state mentioned in Section 3.2, respectively:

$$H_t = O_t \cdot \tanh(C_t) = O_t \cdot \tanh \left(F_t \cdot C_{t-1} + I_t \cdot \tilde{C}_t \right) \quad (11)$$

Based on the formula for input gate, forget gate, output gate mentioned in 3.2, we can expand the formula 11:

$$H_t = \sigma(x_t \cdot k_5 + h_{t-1} \cdot k_6 + b_3) \cdot \tanh(\sigma(x_t \cdot k_3 + h_{t-1} \cdot k_4 + b_2) \cdot C_{t-1} + \sigma(x_t \cdot k_1 + h_{t-1} \cdot k_2 + b_1) \cdot \tanh(x_t \cdot k_7 + h_{t-1} \cdot k_8 + b_4)) \quad (12)$$

Simplify the formula 12:

$$f_i(x_t) = x_t \cdot k_{a_i(t)} + h_{t-1} \cdot k_{b_i(t)} + b_{c_i(t)} \quad (13)$$

$$H_t = \sigma(f_1(x_t)) \cdot \tanh(\sigma(f_2(x_t)) \cdot C_{t-1} + \sigma(f_3(x_t)) \cdot \tanh(f_4(x_t))) \quad (14)$$

We extract a part of the formula 14, denoted P_n :

$$\begin{aligned} P_n &= \sigma(f_2(x_t)) \cdot C_{t-1} + \sigma(f_3(x_t)) \cdot \tanh(f_4(x_t)) \\ &= F_2(x_t)C_{t-1} + F_3(X_t) \cdot \tanh(f_4(X_t)) \end{aligned} \quad (15)$$

Then we simply P_n :

$$\begin{cases} A = F_2(x_t)C_{t-1} + F_3(X_t) \\ B = F_3(x_t) \\ P_n = A + B \cdot \tanh(f_4(x_t)) \\ \quad = A + \alpha(p) \end{cases} \quad (16)$$

Thus, we can divide the formula in 14 into two parts, F_1 , P_n . F_1 denotes the function of the race at the current point in time, and P_n denotes the temporal part, which includes the content of the cell state in the LSTM, i.e., the effect of the previous race situation on the athlete.

$$F_i(x_t) = \sigma(f_i(x_t)) \quad (17)$$

$$H_t = F_1(x_t) \cdot \tanh(P_n) \quad (18)$$

$$P_n = A + \alpha(p) \quad (19)$$

The $\alpha(p)$ in the timing part P_n represents the effect of momentum on the player during the match. This momentum consists of two components: one is the hard power of the player, and the other is the soft power of the player (e.g., scoring, the effect of the opponent on the player's mood during the match, etc.).

At this point, we have modeled the LSTM model in Section 3.2 into a momentum-based model, which we call MM-LSTM.

4.3 Comparison of Stochastic Probability Model and Momentum-Based Model

In order to compare the random probability model and the momentum-based model, we calculated the accuracy of these two models for point win/loss prediction on a test match separately. The mean accuracy of all match predictions is presented in Table 1, and the results for individual match accuracies are illustrated in Figure 7.

Table 1: Mean Prediction Accuracy of Models

Model	Mean Accuracy
Stochastic Probability Model	67.776%
Momentum-Based LSTM Model	94.229%

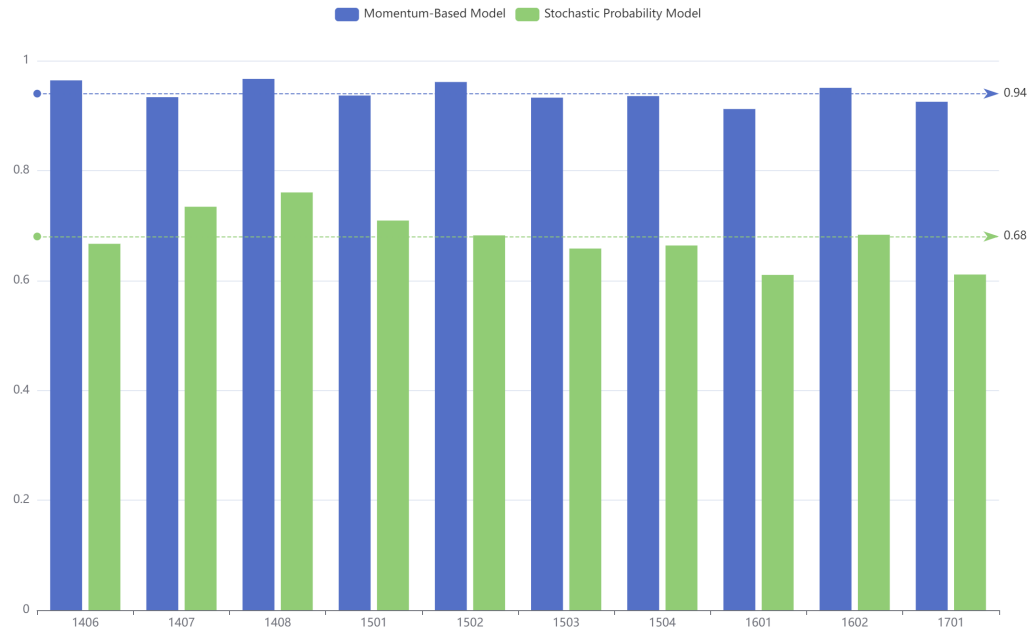


Figure 7: Prediction Accuracy of Stochastic Probability Model and Momentum-Based Model

As shown in Table 1, our momentum-based model outperforms the random probability model with about 25% accuracy. Based on this, we conclude that momentum plays a role in the game process and can be used to predict the winners and losers of the players.

5 Problem 3: Momentum Shift Predictor Based on MM-LSTM

5.1 Model Structure of the Momentum Shift Predictor

Based on the MM-LSTM model in Section 4.2, we formulate a time-series model of the change in player scores that predicts when the situation in a match shifts from favoring one player to favoring another based on its gradient. The formula for our model is as follows:

$$K_{relative}(n, m) = \frac{1}{n - m} \cdot \sum_{i=m+1}^n [P_i + 0.5] = \frac{1}{n - m} \cdot \sum_{i=m+1}^n [A + \alpha(p) + 0.5] \quad (20)$$

Firstly, let's consider the score situation between two athletes, both gradually increasing over time. We assume that the functions describing these two evolving conditions are denoted as f_1 and f_2 . Therefore, we can establish a simple mapping relationship as follows:

$$\begin{cases} y_1 = f_1(t) \\ y_2 = f_2(t) \end{cases} \quad (21)$$

In this case, we take the difference between the scores of the two athletes:

$$y_1 - y_2 = f_1(t) - f_2(t) \quad (22)$$

The gradient of their score function curves can be expressed as:

$$\begin{cases} k_1 = \frac{\delta y_1}{\delta t} \\ k_2 = \frac{\delta y_2}{\delta t} \end{cases} \quad (23)$$

At this point, based on our difference in scores, we can obtain a difference function whose gradient can be expressed as:

$$K_{relative}(n, m) = \frac{\delta(y_n - y_m)}{\delta t} = \frac{\delta y_n}{\delta t} - \frac{\delta y_m}{\delta t} = k_n - k_m \quad (24)$$

And since the score for the n th round is determined based on the score from the $(n-1)$ th round, the decision to score in each round relies on approximating the probability obtained from the MM-LSTM model. The probability results are rounded to obtain a binary outcome, indicating either victory (1) or failure (0). The scores are then updated based on this outcome, linking the MM-LSTM probability model with the scoring function. In another perspective, this can be viewed as a recursive formula. On this basis, we can also derive a scoring probability formula over a certain period:

$$y_n = y_{n-1} + [P_n + 0.5] \implies y_n = y_m + \sum_{i=m+1}^n [P_i + 0.5] \quad (25)$$

The preceding formulas have successfully established a unified connection between probability and scores. Moving forward, let's further explore the relationship between momentum and gradient.

The following formula describes the relationship between the formula for relative gradient difference and momentum. The relative gradient difference is a metric (refer to Formula 24) indicating score changes relative to time (or points). Here, $n-m$ represents the variation in scores over a time period of $n-m$. In general, we typically set $n-m$ to 3 (as we opted for interpolation rather than fitting). In this scenario, the score difference during this period reflects the variation in scores – the cumulative sum of whether each match results in a score. The probability P , predicted by the MM-LSTM model, approximates whether each match results in a score. Rounding P yields a binary outcome: victory (1) or failure (0). In the case of victory, points are added, causing the gradient to increase; in the event

of failure, no points are added, maintaining the gradient. (Considering the relatively modest quantity of gradient, i.e., the increment in victories, over short time periods, plotting a graph with large swings may not be conducive for observation, hence the choice of a relatively large yet manageable value of 3.)

Moreover, in light of Formula 19, where we interpret probability as momentum, we have intricately linked the relative gradient difference to momentum. Consequently, the relative gradient difference has become a key metric for analyzing changes in momentum.

$$K_{relative}(n, m) = \frac{1}{n - m} \cdot \sum_{i=m+1}^n [P_i + 0.5] = \frac{1}{n - m} \cdot \sum_{i=m+1}^n [A + \alpha(p) + 0.5] \quad (26)$$

The following formula represents the count of correct predictions, where P_i denotes the probability predicted at position i , representing the success probability for play 1. Applying the nested rounding function, when the prediction is successful (victory), it is rounded to 0, and when it fails (loss), it is rounded to 1. Adding 1 is done to convert the dimensional indicator of successful predictions to correspond with the dimensional indicator of *point_victor*. This allows for the assessment of the number of incorrect predictions. Substituting this into the frequency formula reflects the probability situation:

$$N_{false} = \sum_{i=1}^n |1 + [P_i + 0.5] - Point_victor_i| \quad (27)$$

The following formula is an inductive formula for calculating the final prediction accuracy:

$$\eta_{accuracy} = \frac{N_{true}}{N_{total}} = \frac{N_{true}}{N_{true} + N_{false}} = 1 - \frac{N_{false}}{N_{total}} \quad (28)$$

5.2 Factor Contribution Analysis

To test which factors contributed most to the model, we used a gradient importance analysis. The gradient represents the direction in which the function grows fastest at a given point, and its magnitude quantifies this maximum growth rate. For the model built in Section 5.1, we calculated the partial derivative of each input with respect to the output, and then calculated the absolute value of that partial derivative and normalized it. The final normalized value obtained represents the degree of contribution of the inputs to the output. The contribution degree of some of the input indicators is shown in Table 2.

Table 2: Gradient Importance Analysis

Factor	Value	Factor	Value	Factor	Value
point_no	0.0117	p1_double_fault	0.0283	p1_winner	0.1415
server	0.0100	p2_double_fault	0.0203	p2_winner	0.1223
serve_no	0.0107	p1_unf_err	0.1662	p1_double_fault	0.0283
serve_width	0.0666	p2_unf_err	0.1894	p2_double_fault	0.0203
return_depth	0.0386	p1_net_pt	0.0085	p1_unf_err	0.1662
p1_ace	0.0135	p2_net_pt	0.1064	p2_unf_err	0.1894

From the table, we can see that `p2_unf_err` contributes the most to the output with 0.1894, `p1_unf_err` contributes the second most to the output with 0.1662, and the rest of the metrics with top 8 contributions are: `p1_winner`, `p2_winner`, `p2_net_pt`, `serve_width`, `return_depth`, `p1_double_fault`. we visualized these data as Figure 8.

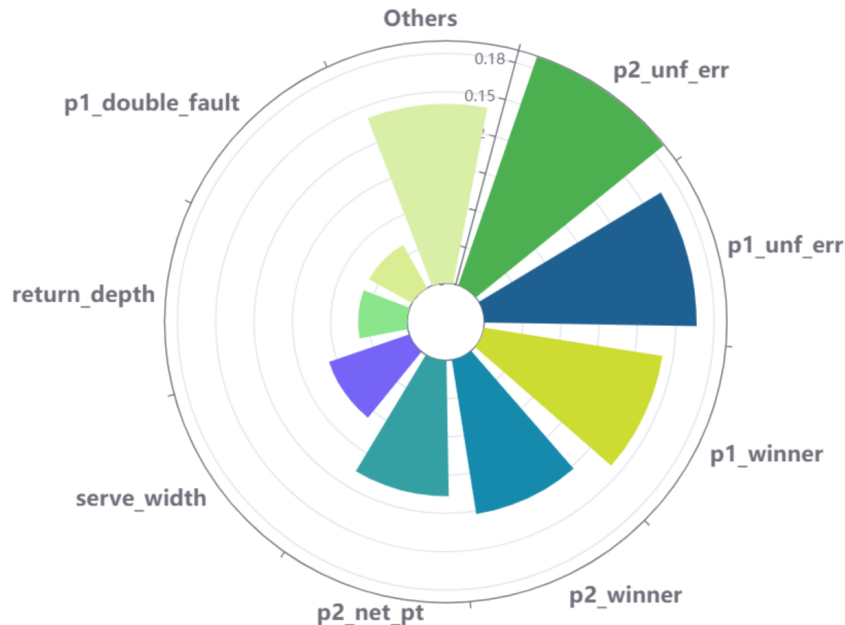


Figure 8: Factors' Contribution Based on Gradient Importance Analysis

5.3 Match recommendations based on momentum changes

Here, we first analyze the quantification of momentum. Regarding the relative strength factors in this model, they are known quantities that allow us to assess the player's strength by comparing their historical performances with other players. Simultaneously, we can extract data from the opponent's match records, such as the timing and proportion of serving faults as the first server, as well as the increase in the number of points won over time. This information helps us understand when the player is more active and demonstrates stronger abilities during the match.

The MM-LSTM model provides the swing in the probability of winning from the beginning to the end, offering insights into the player's changing momentum. Here, the factors unrelated to the match time are analogized as "m", while those related to the match time are analogized as "v". The momentum "p" is influenced by changes in "m" of the same units, causing changes in "v" and directly affecting changes in "p". Therefore, both factors need to be considered in assessing the player's performance.

The figure above displays the relative momentum changes between the two players in two separate matches, with 0 as the starting point. It is evident that the trend of relative momentum changes is unpredictable. For instance, in the match represented by the blue line, Jini initially had the advantage but later allowed the opponent to come from behind. This swing is also reflected in the scoring situation.

Based on the correlation analysis between momentum and other factors, the following recommendations can be derived:

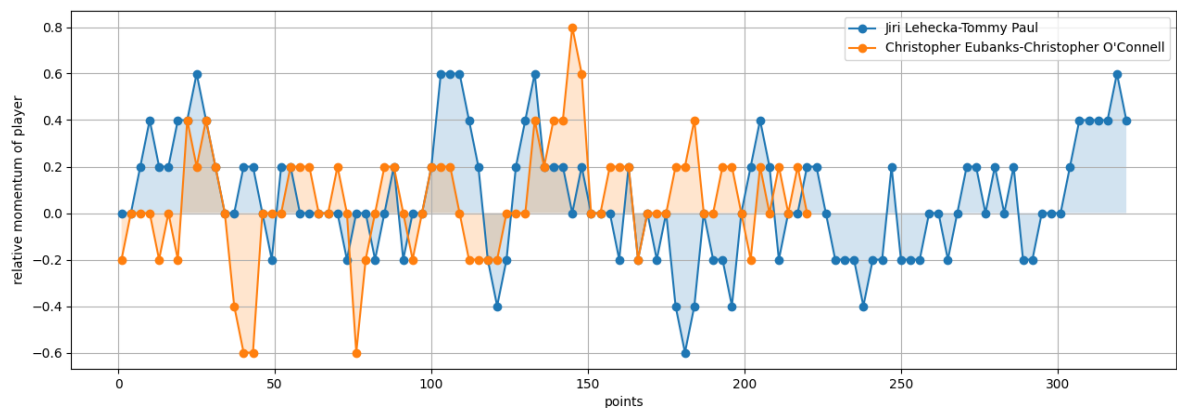


Figure 9: The Momentum Indicator Changes in Two Matches

Psychological Considerations:

- As the saying goes, "make the best effort in the first push, weaken in the second, and exhaust in the third", it can be fair to conclude that, except for the first set, serving has no impact on winning that set.[3] For the first set, serving opportunities should be seized, and caution should be exercised when facing the opponent's serve. Do not become overconfident or careless due to an early advantage, as it may lead to failure in subsequent matches.
- Maintaining composure is crucial when facing uncontrollable outcomes with decisive consequences. The player's mentality directly determines the strength of their current momentum in the game. Unstoppable winning serves and unforced errors can be decisive, with a 100% impact on the outcome of the current exchange. Players should maintain a stable mindset, as the probability of such serves is very small, around 5%, and subject to uncertainty over time. Therefore, there's no need to overly focus on such situations; learn to keep your composure. Additionally, after an unstoppable serve, the calculated probability of winning in the next five games increases to 60%, highlighting the significant role of decisive factors in determining momentum.
- Stay humble and composed. In the data, when the current score is in favor of the player, the average probability of winning the point is 26.81%, indicating that a lead may likely lead to failure. Be cautious in facing break points; winners facing break points tend to score more points, while losers struggle to maintain their level. Breaking and then holding serve is often more challenging than sailing smoothly with the wind.[4]

Non-psychological Considerations:

- Maintain flexibility on the court and avoid being confined to a specific area. According to data analysis, if a player is positioned at the net, their probability of scoring is 69.88%, indicating a significant advantage in playing at the net. Additionally, when calculating the win rate with a running distance exceeding the average, it is 48.75%, slightly lower than the average. This could be attributed to the physical exhaustion caused by increased running distance. However, in the statistics, it was found that the running distance in victorious matches tends to be slightly above

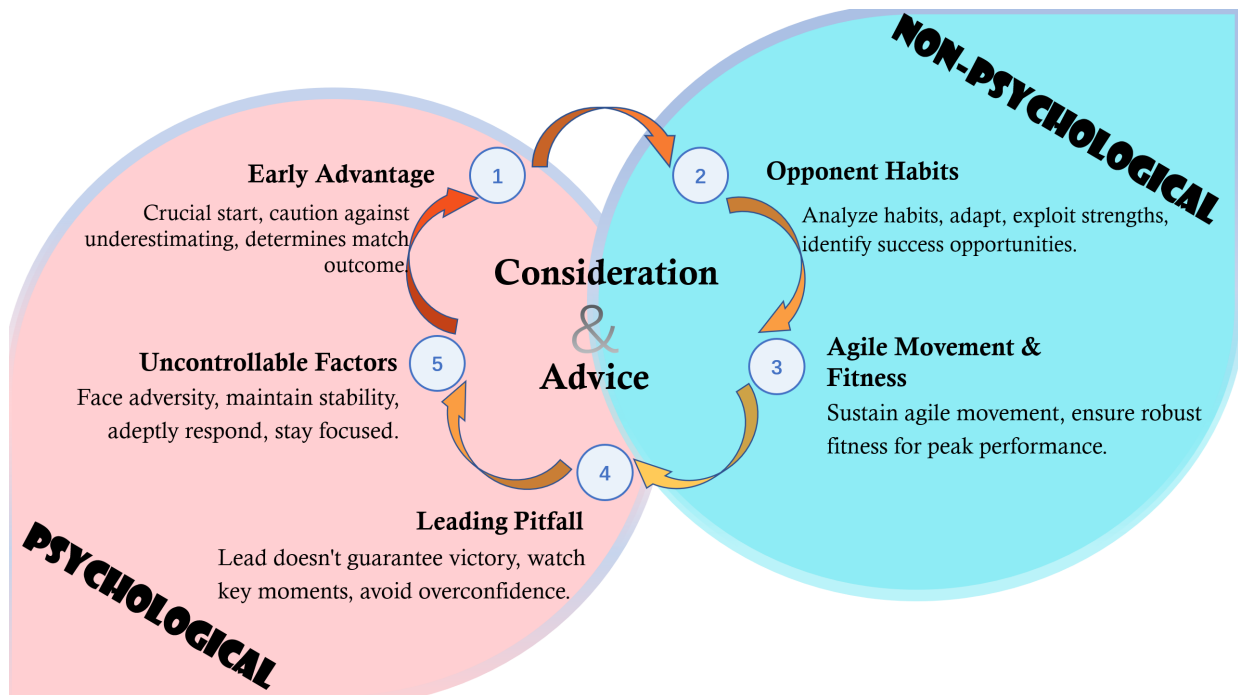


Figure 10: Suggestions for Athletes

the average. This suggests that while physical exhaustion has a minimal impact on victory, agile positioning and an active physique play a crucial role.

- Carefully study the opponent's past match videos and data, paying attention to their strengths and weaknesses. Understand the opponent's playing style to formulate better tactics, including their preference for forehand or backhand, hitting power, speed, running distance, etc. According to the data, most players tend to prefer serving in the "w" direction, particularly favoring "CTL" near the sidelines, and their return shots usually have considerable depth. However, this is a general statistical observation for all players, and individual analysis should be conducted for specific opponents based on their historical data.

In conclusion, we believe that the most crucial aspect is to learn to control your own momentum. The main reason for the swings in momentum fundamentally depends on the changes in a player's strength over time on the court. By mastering and controlling these changes, you can achieve autonomy and make your momentum appear random while being controlled by yourself. This allows for the use of deceptive tactics to mislead opponents or to make continuous and aggressive progress, aiming for a decisive victory.

6 Problem4: Validation and Expansion of MM-LSTM

6.1 Prediction of Momentum Swings

Based on the MM-LSTM model established earlier, we used a match as the test set and obtained the corresponding output results:

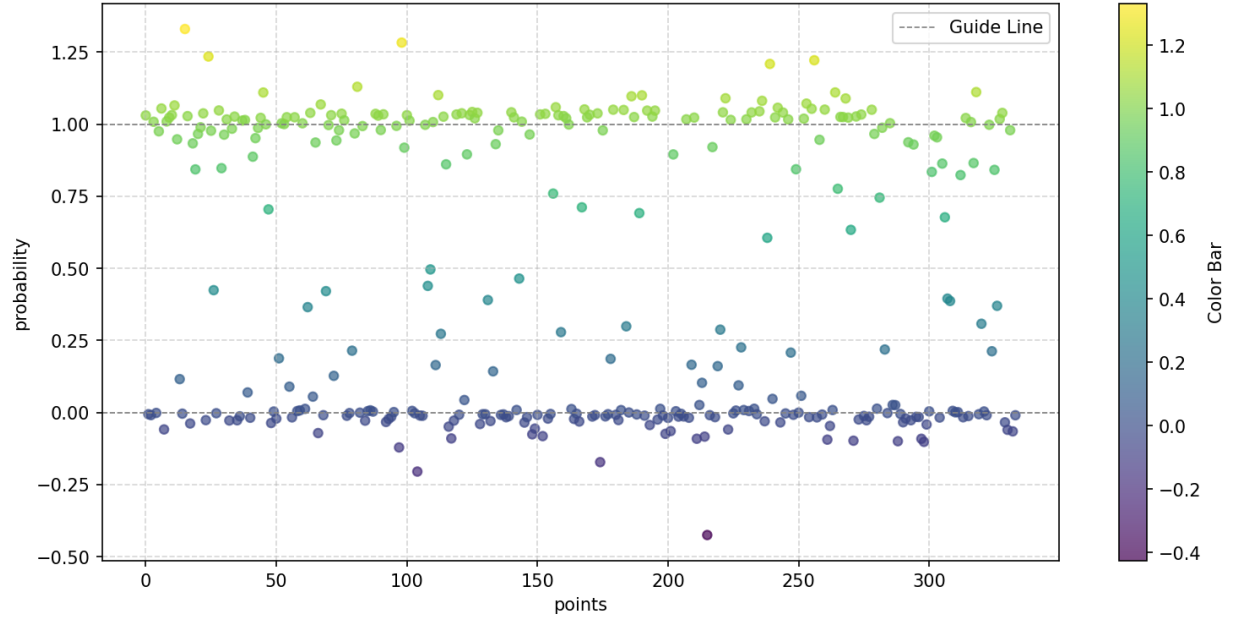


Figure 11: Scatter Plot of Probability Values Predicted Using MM-LSTM Model

We will calculate the probability value using the Equation 27 and Equation 28 obtained from 5.1 to obtain $\eta_{accuracy} = 90.71\%$, the fitting effect is good. We will use Equation 26 to calculate the predicted relative slope curve of the fitted data, and plot the curve with the real data in the following figure:

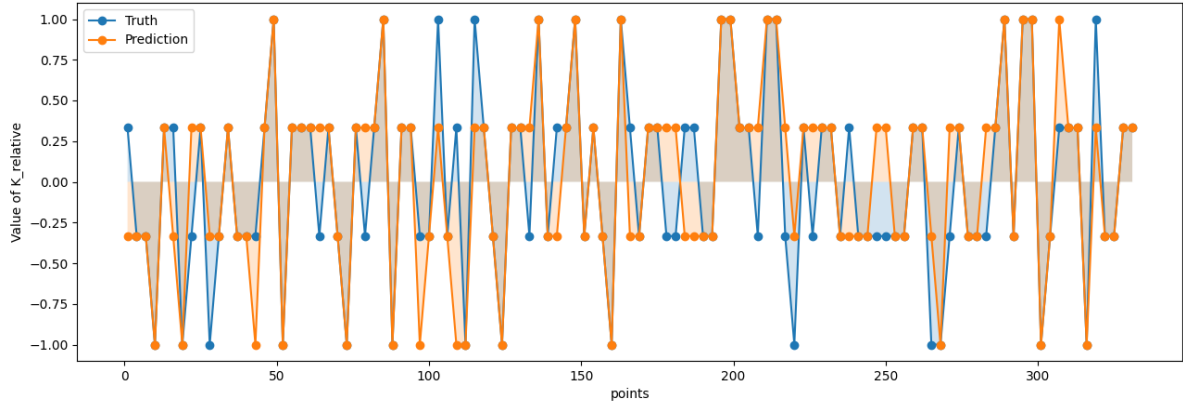


Figure 12: The Fitting Effect of Momentum True Value and Predicted Value

In the above figures, we can observe that the predicted momentum fluctuation curve largely overlaps with the actual momentum fluctuation curve. The calculated coefficient of determination R^2 is 61.51%, indicating a moderate level of fit. The discrepancy between correlation and prediction accuracy could be attributed to the relationship between the fitting coefficient and predicted values, which involves the slope and function values. Additionally, the chosen time period is not a small subset but rather a segment, contributing to the observed difference. In the calculations, the best fitting performance is

achieved when the number of data points in the time period is 3. We believe that too short a time span leads to excessive fluctuation in the graph, resulting in poor fitting performance. On the other hand, an overly long time span tends to average out the swings, making it challenging to capture the nuances.

6.2 Future Model Influence Factors Analysis

Based on the analysis of the above model prediction results, although it has reached 90.71% belonging to the acceptable range of the predictive model; however, the model influence factor mainly focuses on the athletes' performance in the competition and its quantitative indexes, and it does not take into account many factors other than non-athletic performance factors. Therefore the assessment and quantification of subjective and objective factors of non-athletic performance is a direction that can be further optimized in the future model.

6.2.1 Objective Factors

The above model mainly analyzes the data of Wimbledon, where "serve is king", so the proportion of serve-related factors in the prediction model is high. However, in addition to Wimbledon, there are also the Australian Open, the US Open and the French Open among the four major tennis tournaments. Each tournament has its own unique characteristics, especially the venue factor is particularly prominent, such as Table 3 is the analysis of the characteristics of different venues.

Table 3: Tennis Court Characteristics and Associated Tournaments

Court Type	Ball Speed	Return Height	Return Speed	Delegate Competition
Clay Courts	1	Very High	Medium	French Open
Hard Courts	2	Stable	Fast	Australian Open & U.S. Open
Grass Courts	4	High	Very Fast	Wimbledon Championships
Carpet Courts	3	Medium	Medium	ATP Indoor Race

Non-standardized court surface criteria are detrimental to the generality of the prediction model; the strong correlation between serve and rally speeds on grass at Wimbledon led players to generally adopt aggressive attacking strategies and serve to win; conversely, French Open players were more conservative, resulting in the longest average court time of the French Open matches. To enhance the predictive model, the training set can be extended to various court conditions. Ultimately, including court conditions as an influencing factor will improve the overall predictive accuracy of future models.

6.2.2 Subjective Factors

The subjective factors of the game are difficult to quantify, in which the psychological quality and the strength of the comparison is mutual influence, affecting the trend of the game. The level of the opponent indirectly affects the psychological state of the player, when encountering stronger opponents, may be nervous, resulting in malfunctioning; or there will be a "strong is strong" phenomenon can be over-performed. Correspondingly, when encountering weaker opponents, different psychological fluctuations will also occur, but the degree to which the psychological fluctuations affect the performance

of different players varies. Therefore, it is very important for players to develop their self-regulation ability, anti-stress ability, and the ability to adapt to the venue from the experience of the tournament.

6.3 Testing the generalizability of the model on table tennis matches

To test the generalizability of our model across other sports, we collected data from the WTT 2024 table tennis matches. We trained and tested the model on a dataset consisting of 1515 table tennis matches. We partitioned 80% of the data for training and reserved the remaining 20% for testing. After training for 1000 epochs, our model achieved a prediction accuracy of 75.574% on the test set, which did not reach the prediction accuracy achieved in tennis.

To analyze the reasons for the unsatisfactory performance of the model, we compared the collected data from the WTT 2024 matches with the data provided by MCM on the Wimbledon 2023 Gentlemen's singles matches after the second round. We believe that the suboptimal prediction results in table tennis matches are due to the following two factors:

1. The quantity of data collected for table tennis matches is relatively small. The tennis dataset comprises 7,285 data entries, whereas the table tennis dataset contains only 1,515 data entries, amounting to just 20.8% of the tennis data in terms of entry count.
2. The table tennis dataset lacks a variety of metrics compared to the tennis dataset. The table tennis data includes only the match ID, player names, and the scores of each player, without various in-game state metrics (such as "p1_double_fault" and "p2_double_fault"). Consequently, the total count of metrics in the table tennis data sums up to 6,060. In contrast, the tennis data, when counted by the number of metrics, totals 335,110. In summary, the sum of the table tennis metrics accounts for only 1.8% of the total count of metrics in the tennis dataset.

In conclusion, we believe that the MM-LSTM model we designed exhibits a certain degree of generality. However, its performance is constrained by the limited quantity of data. If more match data can be collected to allow the model to learn a greater variety of match-related information, there is substantial potential for improvement in model performance.

7 Model Analysis

7.1 Strengths and Weaknesses

Strengths.

1. **Close to real-world scenarios:** In problem 1, we utilized the three gate mechanisms and Cell state of LSTM to simulate the human processes of memory formation and forgetting, closely approximating real-life scenarios.
2. **Temporal memory capability:** The LSTM model captures and remembers long-term dependencies through its Cell state, enabling it to effectively process long sequences of data and form a memory of the data throughout the competition process.

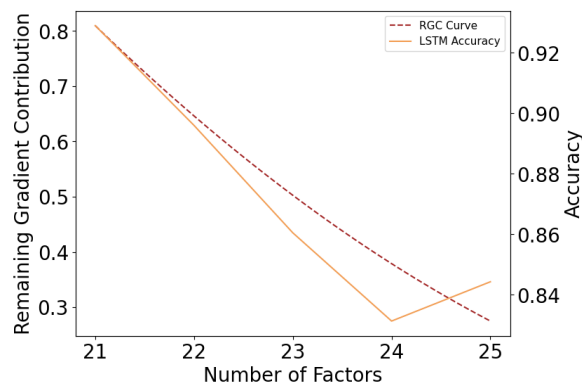
3. **Momentum-based model representation:** In problem 2, we modeled the LSTM as a mathematical model based on momentum, providing a more formal analysis of the intrinsic relationship between the LSTM model and the momentum model.
4. **Strong generalization capability:** We performed dimensionality reduction on the data from the competition, extracting some mutually independent factors to be used as inputs for the model, thereby enhancing its generalization capabilities when faced with new data.
5. **Objective results:** The results of the model are quantifiable and can be described by mathematical representations so that speculative thinking is not required.

Weaknesses.

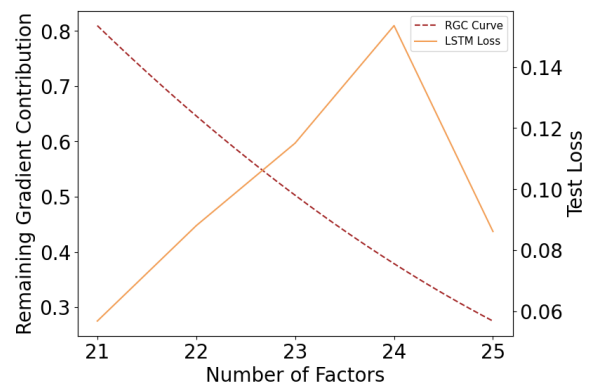
1. **The momentum modeling is not comprehensive:** Due to the influence of many non-intuitive factors in the competition process, even though our LSTM model considers many factors in the game, it still cannot accurately model the momentum of players throughout the competition.
2. **Insufficient prediction accuracy:** To further improve accuracy, we need to use more game data to enable our deep learning model LSTM to learn more information about the games.

7.2 Sensitivity Analysis

To validate the model's sensitivity, we simulated scenarios of missing data by selectively removing portions of the input data. Specifically, we sequentially excluded several variables that were identified to have the largest gradient contributions in Section 5.2. The model's accuracy decreased by only 2% when the variable with the greatest gradient contribution was removed. When five variables were excluded, the model still maintained an accuracy of 84%. These results indicate that our model can adapt to partial data missingness, demonstrating strong robustness and resilience. The related experimental analysis results are presented in Figure 13.



(a) The accuracy of LSTM changing with the number of factors



(b) The loss of LSTM changing with the number of factors

Figure 13: Figures for Sensitivity Analysis

8 Memorandum

To: The Global Professional Tennis Coach Association (GPTCA)

From: Team #2416613

Subject: The Role of Momentum in Tennis Movement

Date: February 5, 2024

Dear Officer of GPTCA:

From the model established by our team based on match data, we have found that Momentum plays a crucial role in tennis matches. Evaluating the Momentum indicator through model parameters in multiple matches for the same player allows for a high probability prediction of the score trend in subsequent matches. Furthermore, after neutralizing the skill gap between opposing players, the Momentum between different players in the same match shows a positive correlation with their performance.

Analyzing the predictive effects and understanding the changes in momentum, we have several important findings and suggestions that we hope will help improve the accuracy of your decisions.

1. Place special emphasis on the initial momentum shifts of both the opponent and one's own team during the match. Through the analysis presented earlier, it was observed that the momentum at the beginning of the game has a significant impact on the overall match. However, it is equally important to acknowledge the potential threat posed by momentum shifts resulting from underestimating the opponent.
2. In terms of model predictions, the undeniable improvement in accuracy of the MM-LSTM model compared to the fitted traditional models serves as compelling evidence for the powerful influence of momentum.
3. For certain decisive factors, such as unstoppable winning serves or serving errors, their impact on momentum is significant and can influence players' attempts to build on success or stage a comeback during the match.

Additionally, here are recommendations for players dealing with unexpected and decisive events in the flow of a tennis match:

1. It's essential to break free from negative thinking early on and mentally fortify oneself like a warrior. Pay no mind to the opponent and strive to exhibit strength and resilience.
2. Choosing the right moment to ignite one's inner passion undoubtedly proves more beneficial in the competition. Avoid expending energy too early and learn to manage time and energy wisely.
3. During practice, keep score, so you can forget about the score during the match. In data analysis, it is evident that previous historical scores have a significant impact on the current situation.

In conclusion, in a match, robust physique, a stable and healthy mindset, appropriate strategic planning, and exceptional technical skills are all indispensable.

**Yours Sincerely,
Team #2416613**

References

- [1] Briki W, Den Hartigh R J R, Markman K D, et al. How psychological momentum changes in athletes during a sport competition[J]. *Psychology of Sport and Exercise*, 2013, 14(3): 389-396.
- [2] Klaassen F, Magnus J R. On the probability of winning a tennis match[J]. *Medicine and Science in Tennis*, 2003, 8(3): 10-11.
- [3] Magnus, Jan R., and Franc JGM Klaassen. "On the advantage of serving first in a tennis set: four years at Wimbledon." *Journal of the Royal Statistical Society: Series D (The Statistician)* 48.2 (1999): 247-256.
- [4] Meffert, Dominik, et al. "Tennis serve performances at break points: Approaching practice patterns for coaching." *European journal of sport science* 18.8 (2018): 1151-1157.

Appendices

Core Code of LSTM

```
1  import torch
2  from torch import nn
3  class LSTMModel(nn.Module):
4      def __init__(self, input_size, hidden_size, num_layers,
5          output_size):
6          super(LSTMModel, self).__init__()
7          self.hidden_size = hidden_size
8          self.num_layers = num_layers
9          self.lstm = nn.LSTM(input_size, hidden_size, num_layers,
10             batch_first=True)
11          self.fc = nn.Linear(hidden_size, output_size)
12  def forward(self, x):
13      h0 = torch.zeros(self.num_layers, x.size(0), self.
14          hidden_size).to(x.device)
15      c0 = torch.zeros(self.num_layers, x.size(0), self.
16          hidden_size).to(x.device)
17      h0 = h0.squeeze(0)
18      c0 = c0.squeeze(0)
19      out, _ = self.lstm(x, (h0, c0))
20      out = self.fc(out[:, -1, :])
21      return out
```

Report On use of AI

1. OpenAI ChatGPT (Feb 6, 2024 version, ChatGPT-3.5)

Query1: Could you please explain the exact interpolation method and formula?

Output: Certainly! Interpolation is a technique used to estimate values between two known values or data points. There are various interpolation methods, and the choice of method depends on the nature of the data and the desired level of accuracy. I'll explain two commonly used interpolation methods: linear interpolation and Lagrange interpolation.

1. Linear Interpolation: Linear interpolation is a simple method that assumes a linear relationship between two adjacent data points. Given two points (x_1, y_1) and (x_2, y_2) , the linear interpolation formula for estimating the value y at a point x between x_1 and x_2 is:

$$y = y_1 + \frac{(x - x_1) \cdot (y_2 - y_1)}{x_2 - x_1}$$

2. Lagrange Interpolation: Lagrange interpolation uses a polynomial to fit the data points. For $n + 1$ data points $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$, the Lagrange interpolation polynomial is given by:

$$P(x) = \sum_{i=0}^n y_i \cdot L_i(x)$$

where $L_i(x)$ is the Lagrange basis polynomial:

$$L_i(x) = \prod_{j=0, j \neq i}^n \frac{x - x_j}{x_i - x_j}$$

The Lagrange polynomial passes through all the given data points and can be used to estimate the function values at points between the given data.

These formulas provide a simplified explanation of linear and Lagrange interpolation. In practice, for more complex data, other interpolation methods like spline interpolation or polynomial interpolation of higher degree might be used, depending on the specific requirements and characteristics of the dataset.

2. OpenAI ChatGPT (Feb 4, 2024 version, ChatGPT-3.5)

Query1: Gradient calculation formula

Output: In mathematics and computation, the gradient is typically a vector that represents the rate of change or slope of a function at a specific point. For a scalar function $f(x)$, its gradient $\nabla f(x)$ can be computed through partial derivatives. If $f(x)$ is a multivariable function, the gradient is a vector where each component corresponds to the partial derivative with respect to the respective variable.

The general formula for gradient computation is as follows:

For a scalar function $f(x_1, x_2, \dots, x_n)$, its gradient ∇f is a vector containing partial derivatives:

$$\nabla f = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n} \right)$$

Here, each $\frac{\partial f}{\partial x_i}$ represents the partial derivative with respect to the variable x_i .

If you have a specific function $f(x_1, x_2, \dots, x_n)$, you can use this formula to compute its gradient.