Research on Wordle Puzzle Game based on trend prediction model based on data index

Xuanyun Liu[1, *], Yiyang Huang[2, #], Kuntao Huang[3, #]

[1]College of science, University of New South Wales, Sydney, Australia, 2052

[2]College of Electrical Engineering and Automation, Fuzhou University, Fuzhou, China, 350108

[3]Faculty of Business, The Hong Kong Polytechnic University, Kowloon, China, 999077

*Corresponding author: z5432542unsw@gmail.com

#These authors contributed equally.

Abstract. World Puzzle is a word game that has exploded in popularity over the past year. For this kind of game phenomenon, this paper studies. First, the LSTM algorithm is used to describe the trend of their number and to fit a high-precision time model. At the same time, considering the rigor, this paper also establishes the ARIMA time series analysis model to predict the data again. On this basis, the data accuracy of the two models is compared, the model is selected, and five indexes such as the number of words and the number of meanings are defined. Second, Pearson correlation coefficient was used to investigate the relationship between the effects of these indicators on the percentage of reported scores in difficulty mode. Finally, BP iterative neural network was used to calculate the correlation between the indicators and the percentage of players attempting to solve puzzles each time, and GBDT machine learning method was used to quantify the relationship between the word difficulty attribute and the reported result distribution, so as to establish the correlation model between the word attribute and the percentage of passing rate. The results show that LSTM model is more suitable than ARIMA model to describe the trend of the number of reports over time. Meanwhile, when the word "EERIE" was inserted into the model, the average number of correct guesses was 5.399.

Keywords: Wordle Charades, LSTM, BP Neural Network, GBDT.

1. Introduction

Wordle is currently a daily scrabble game offered by the New York Times to the public, in which players try to solve a puzzle by guessing a five-letter word in six or fewer attempts, receiving feedback for each guess. [1] The game can be divided into regular and hard modes, depending on the amount of feedback they give. Since the words given every day have different tables in some features, this paper makes a mathematical model for the different features of each word and the final overall score of the competition, so as to predict the future data and thus predict the future trend of this game.[2]

Based on this background, this paper establishes a model of the number of reports changing over time, considers the influence of word attributes on future results, and predicts the future results. At the same time, it predicts the result of word guessing on a future day, predicts the expected answer score of specific words, and studies the word "EERIE" as an example to determine the average number of correct guesses, which provides a directional guide for the future improvement and upgrading of the game.

2. Data source

This paper uses question C from the 2023 MCM/ICM contest (available at: The Mathematical Contest in Modeling (MCM) and The Interdisciplinary Contest in Modeling (ICM) - COMAP) through analysis and research, the data obtained are analyzed by descriptive statistics, and the rule between user customs clearance data and word difficulty is found.

3. Time-Report Volume Change Model

3.1. Data Analysis

Based on the data given, the number of reports from 7 January to 31 December in 2022 , looking at the structure reveals a steeper growth curve in the number of reports initially, before slowly falling back and stabilising at around 20,000.

3.2. Predict Data by Long Short-Term Memory Model

We develop a model that can account for the change in the number of reports and predict the number of reports as of 1 March 2023. While time series forecasting models are usually used, this question considers the use of a LSTM model to describe the change in the number of reports over time.

LSTM (Long Short-Term Memory) At the core of LSTM are three multiplicative units: the input gate, the output gate and the forget gate. the LSTM combines short-term and long-term memory through gating to fit the data. There are 359 sets of date and number of reports in the observed data, we control the first 200 as the training set and the last 159 as the test set, these 359 data are the input gates i_t. and 100 time units were predicted backwards as output gates o_t.

Definitions f_t, c_t, h_t as the forget gate, cell state, hidden state. Details are shown in Figure 1.

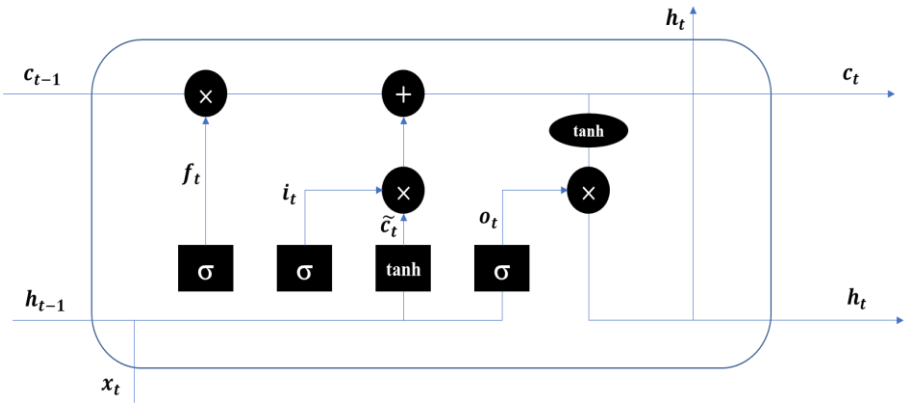


Figure 1: Schematic Diagram of LSTM

3.2.1 LSTM-Step

Step1: Deciding on the time-to-report number of messages to discard. The current input x_t(reported data) and the previous neuron information h_{t-1} (last set of reported data) are first read and the forgetting gate f_t determines the discarded information.

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{1}$$

Step 2: Determine the new information stored in the cell state.

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \tag{2}$$

$$\tilde{c_t} = tanh(W_c[h_{t-1}, x_t] + b_c) \tag{3}$$

Step 3: Update the state by multiplying the old state and$f_t$discarding the useless information and determining the new candidate value $i_t \cdot \tilde{c_t}$.

$$c_t = f_t \cdot c_{t-1} + i_t \cdot \tilde{c_t} \tag{4}$$

Step 4: Determine the output.

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = o_t \cdot \tanh(c_t) \tag{6}$$

3.2.1 LSTM-Processing

In MATLAB, set for 400 iterations of training and 191 single-step predictions (191 is the prediction value used for validation and 100 is the value predicted going forward. The gradient threshold was set to 1. An initial learning rate of 0.01 was specified, which was reduced after 125 training rounds by multiplying by a factor of 0.2. Details are shown in Figure 2.

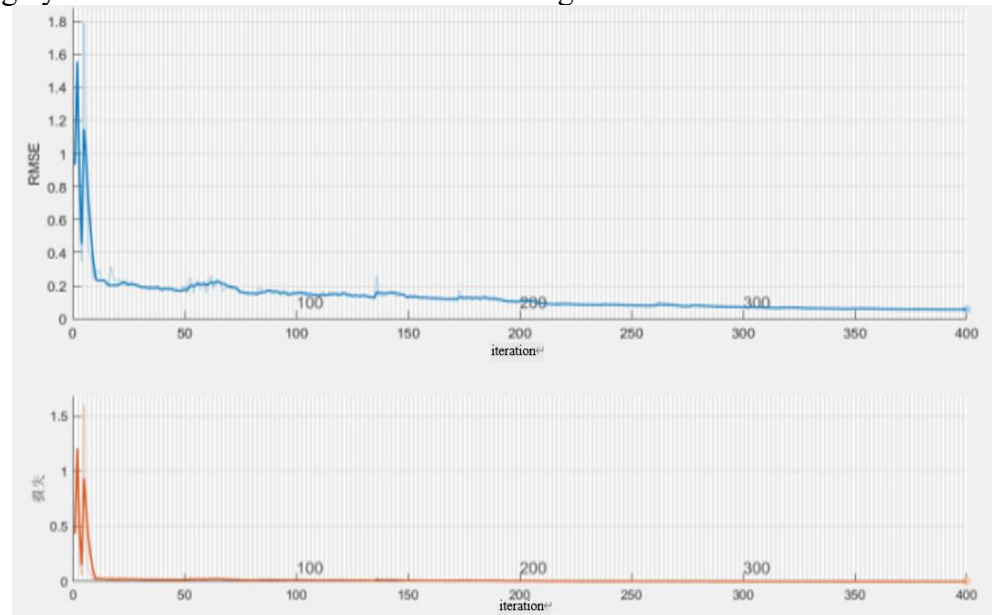


Figure 2: Training Progress

The first 200 units are the sample set, the next 159 sub-units test the set and finally the 100-unit prediction model is performed. Details are shown in Figure 3.

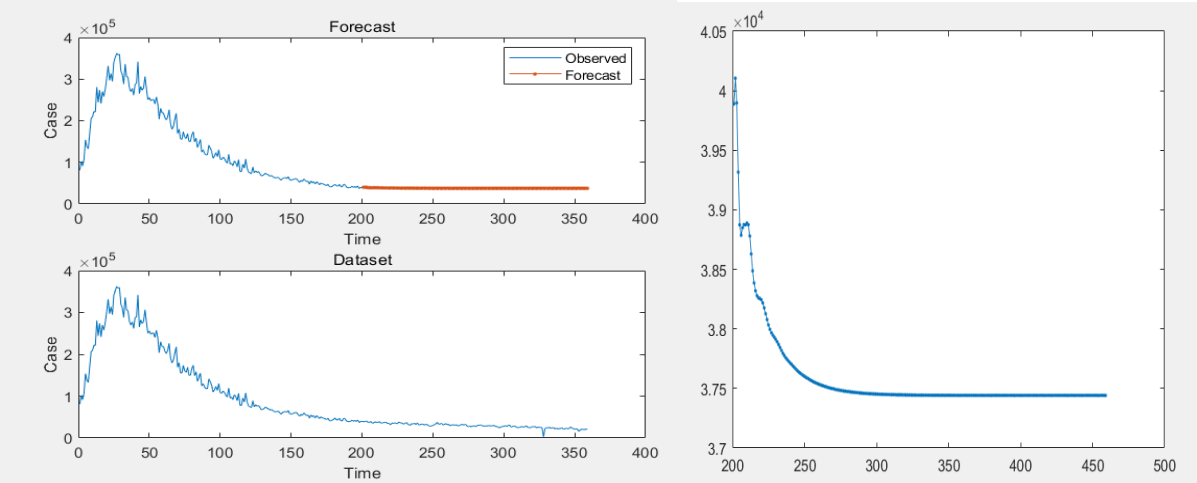


Figure 3: Prediction Chart

The final projections were obtained as Table 1.

Table 1: LSTM Model Prediction Results

Jan 10th	38888.648
Jan 20th	38244.75
Jan 30th	37912.234
Feb 10th	37701.875
Feb 20th	37593.828
Mar 1st	37535.359

The report number for Mar 1st is therefore predicted to be around 37535.359, with a margin of error of around 19%.

3.3. Accuracy Comparison and Interval Prediction by ARIMA Model

We considered numerous time series models, and for accuracy considerations built up an ARIMA time series analysis model to forecast the future, comparing the data and accuracy of both, and choosing the optimal model to describe the amount of change reported over time.[3,4]

The ARIMA flow chart is shown below. The data is first smoothed and processed, then the model is identified, the next step is parameter estimation, and after this series of operations is completed, the model is tested and then the model can be applied to the data for time series estimation. Details are shown in Figure 4 and 5.

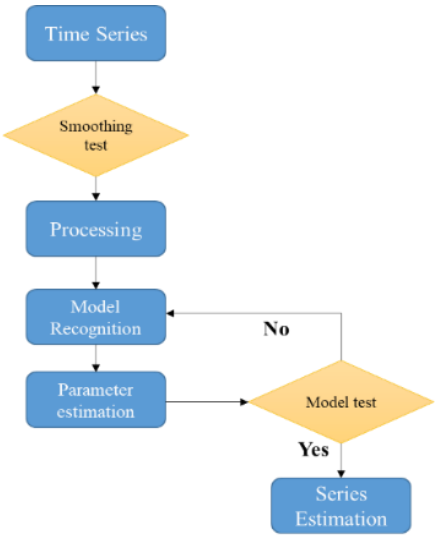


Figure 4: ARIMA Step

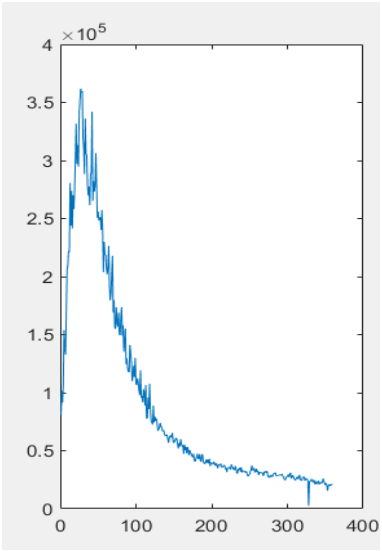


Figure 5: Second Order Differential Processing

(1) Autoregressive modeling (AR)

$$y_t = \mu + \sum_{t=1}^{p} \gamma_i y_{t-i} + \varepsilon_t \tag{7}$$

(2) Moving average model (MA) building

$$y_t = \mu + \varepsilon_t + \sum_{i=1}^{q} \theta_i \varepsilon_{i-t} \tag{8}$$

(3) The MA and AR formulas are as follows

$$\mathrm{ACF(k)} = \mathrm{P_k} = \frac{\mathrm{Cov(y_t, y_{t-k})}}{\mathrm{Var(y_t)}} \tag{9}$$

$$x_t = \emptyset_1 x_{t-1} + \emptyset_2 x_{t-2} + \ldots + \emptyset_k x_{t-k} + \vartheta_t \tag{10}$$

Set y_t is the current value, $\mu$is the constant term, p、q is the order, γ_i is the autocorrelation coefficient and ε_t is the error,The calculated results are shown in Table 6 and 7.

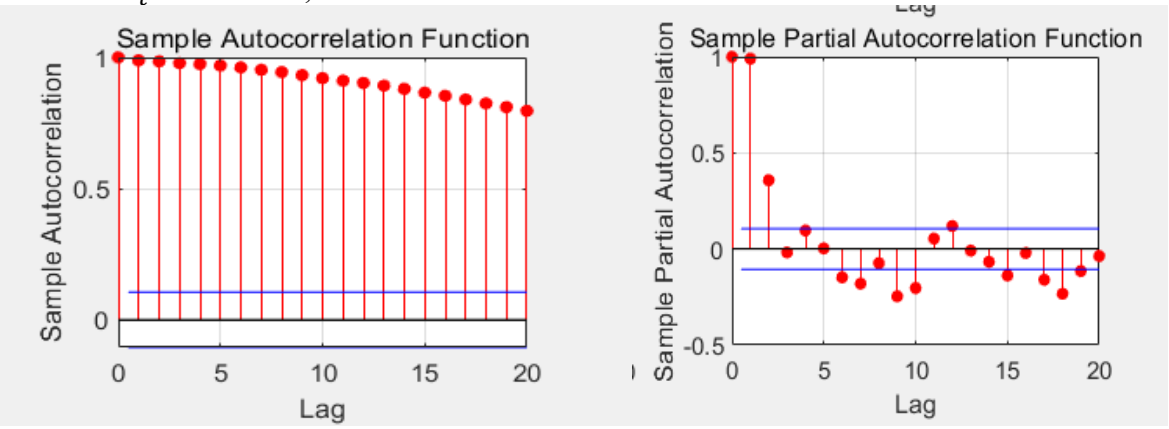


Figure 6: AR Function

Figure7: MA Function

i,

The results are shown above. The final ARIMA time series calculation results are shown in Table 2, which are compared with the lstm model to obtain:

Table 2: Comparison of LSTM and ARIMA prediction results

Jan 5th	38873.883	21187.5047
Jan 25th	38032.625	23284.70925
Feb 15th	37638.586	25305.76907
Feb 25th	37557.602	26206.61024
Mar 1st	37535.359	26556.36572

Result: The data for Mar 1st under the ARIMA model is 26,556 reports with an error rate of 29%.

3.4. Results and Model Evaluation

We compared the error rates of the models, with the LSTM model having an error rate of 17% and the ARIMA model having an error rate of 29%, so the LSTM model was chosen to explain the time-data volume variation , and used to predict the number of results reported on 1 March 2023 to be within an interval of around 37535.

Because the ARIMA model needs to be based on smooth serial data, the data we examined for this question was not smooth and the sample data set in the case was large with 359 sets of time-volume information. This is therefore the main reason for the poor accuracy.

Compared to ARIMA, the LSTM model, which has a memory function and is suitable for large data sets, is more efficient in the application of time series forecasting, has the advantages of being seasonal and has no delay lag. Therefore, in terms of the output of the results, the LSTM model predicts data more closely to the actual.

4. Model Research on Word Attributes and Percentages of Scores in Hard Mode

4.1. Definition of Word Attributes

From the analysis of the previous problem, the reported number of predictions is related to the public's mastery of the words, and the number of attempts will be lower in case of common words. To determine which attributes are correlated with the number of attempts, we create five metrics describing the words and later performed a correlation analysis. The definition criteria are shown below:

Length: It is determined by the number of letters. In general, the more letters, the more attempts.

The Number of Word Meaning: A word often has more than one meaning, which can easily influence the choice of the players. Use python to crawl Collins dictionary and find the number of word meanings.

Quantity Level: Players may tend to choose more common words. We use python to crawl the search volume of words in Collins dictionary. According to the number of searches, 0~500 is Level 1, 500~1000 is Level 2, >1000 is Level 3.

Details are shown in Figure 8.

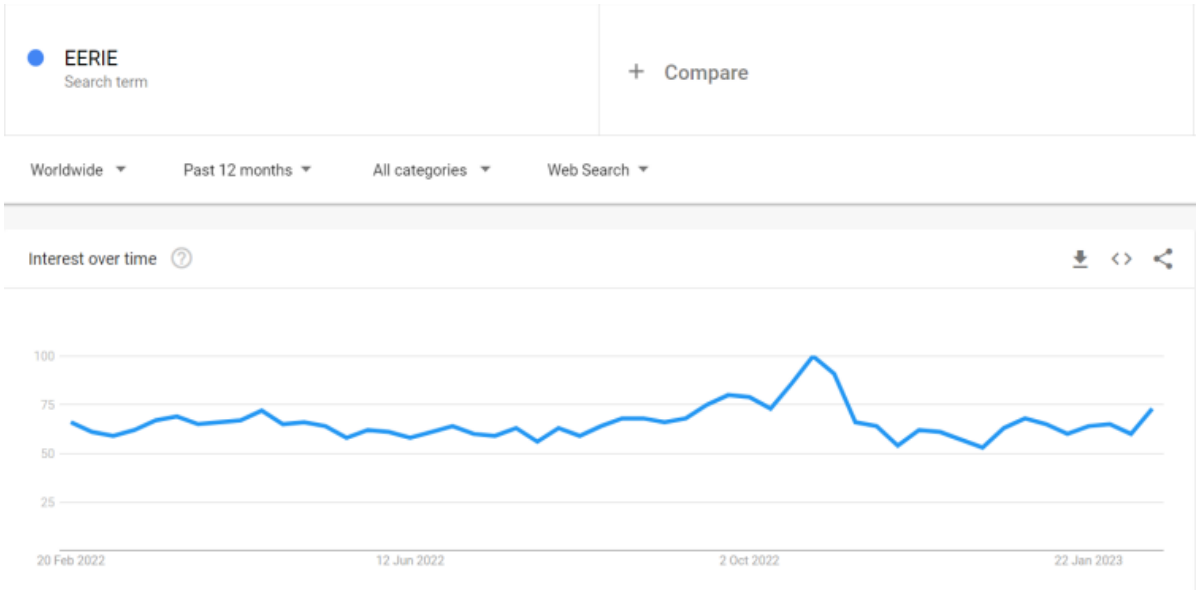


Figure 8: Search Term of EERIE from Google Trends

Part of Speech: It is divided into verbs, nouns, adverbs, and prepositions, and the most common meaning of the word is selected for assignment, with 1 for verbs, 2 for nouns, 3 for adverbs, and 4 for others.

The Number of Letter Repeats: If a player selects the letter in yellow tile, it is generally difficult to select that letter again. Therefore, if the letter is repeated, it will affect the player's judgment to some extent. Details are shown in Figure 9.

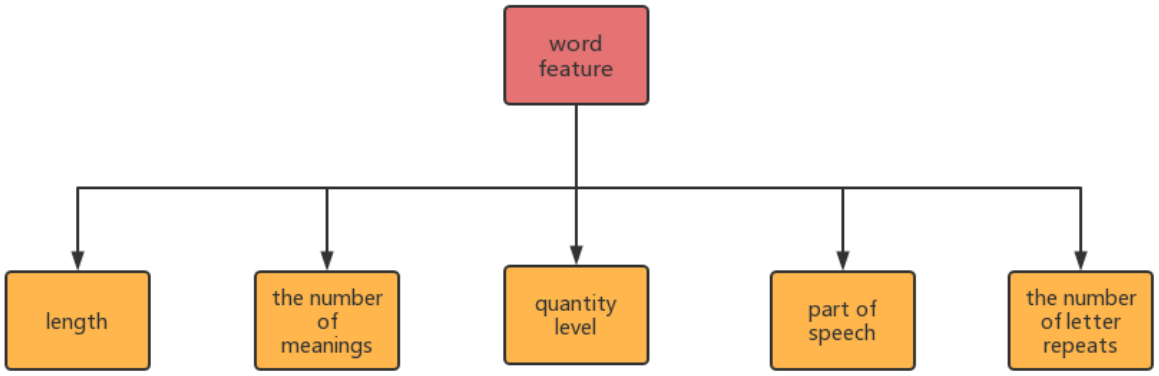


Figure 9: Word Attributes

4.2. Correlation Analysis on Word Attributes

We used the Pearson correlation test to eliminate useless variables and thus streamline the number of independent variables in the fitted equation. The formula of Pearson correlation coefficient is as follows.

$$r_{pearson} = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i-\bar{Y})^2}} \tag{11}$$

where, x_i is the independent variable, corresponding in turn to Length, The Number of Word Meaning, Quantity Level, Part of Speech, The Number of Letter Repeats. y_i is the dependent variable, corresponding to Hard Mode Completion Rate. The results are shown in the following figure 10.

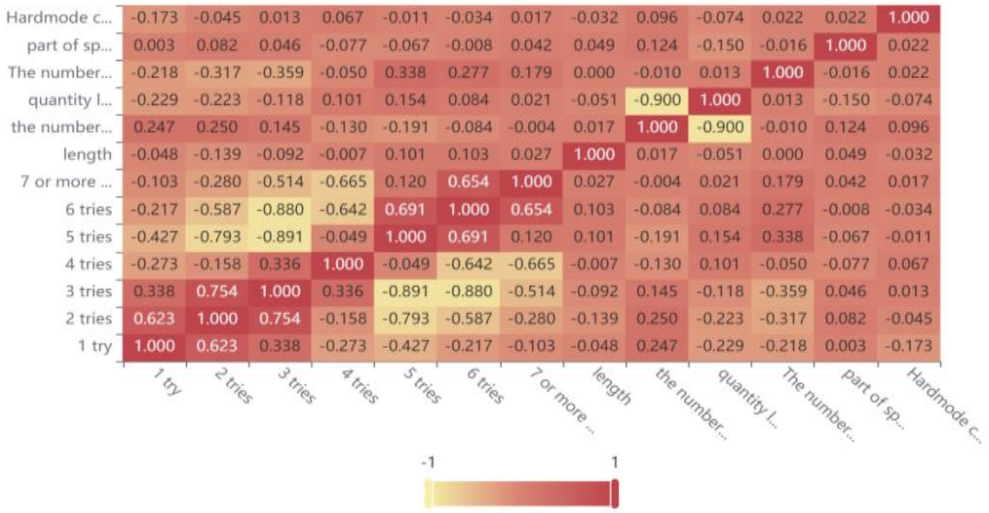


Figure 10: Pearson Correlation Coefficient

The Number of Word Meaning and Quantity Level are attributes that significantly affect Hard Mode Completion Rate. The correlation between the two and the completion rate was 0.096 and -0.074, respectively. This indicates that the more meaningful or more remote a word is, the more likely it is to be guessed. Differently, the correlation between Part of Speech or The Number of Letter Repeats and the completion rate was 0.022, which indicates that these two attributes do not have a significant relationship with the completion rate.

The model is tested: The significance P-value is 0.729, which presents significance and the null hypothesis can be rejected. In terms of variable co-linearity, VIF is all less than 10, so the model has no problem of multiple co-linearity. Therefore, it shows that the word attributes can have an impact on Hard Mode Completion Rate. The degree of fit of the true and predicted values is shown in Figure 11.

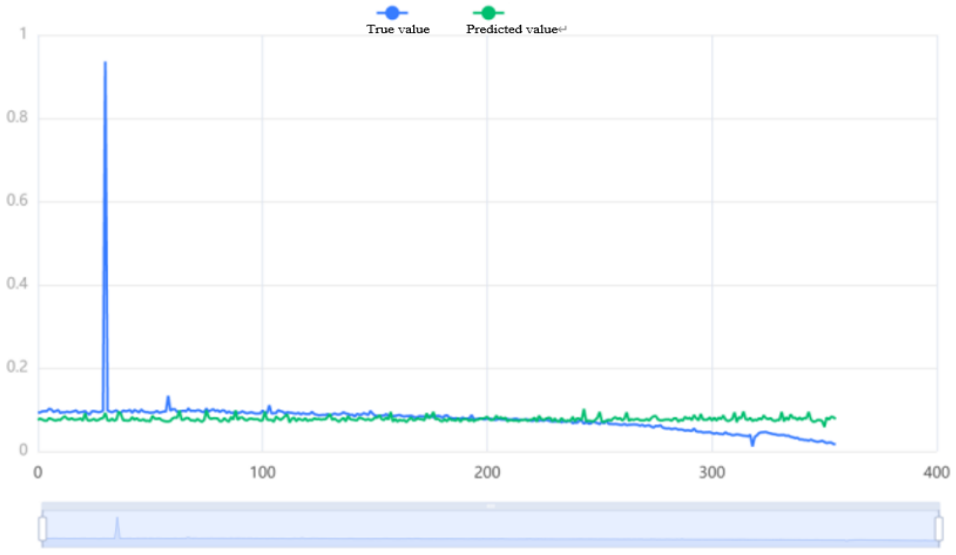


Figure 11: The Effect of Word Attributes on Hard Mode Completion Rate

5. Prediction of the Distribution of the Report Results

5.1. Explore Data Correlation by Back Propagation Neural Network

To achieve data prediction, BP (Back Propagation) Neural Network was used to find the correlation between the difficulty of a word and the distribution of reported results by combining the five word attributes established in Problem 1 to determine the relationship between a multiple word indicator and the percentage of reports.[5,6]

5.1.1 Data Pre-processing

Based on attachments, the times the result was inferred for each word was calculated and set to Avenge. The details are shown in Table 3 (partial data).

Table 3: Number of Speculations Required for Words (Partial)

Word	Length	Meanings of Words	Quantity Level	Part of Speech	The Number of Letter Repeats	Avenge
manly	5.00	4.00	3.00	adj	1.00	4.38
molar	5.00	8.00	2.00	n	1.00	4.18
havoc	5.00	4.00	3.00	n	1.00	4.40

5.1.2 Model Establishment

Considering the lack of large amount of data in this problem, it may lead to the BP neural network learning to converge too slowly and cannot guarantee convergence to the global minimum point. Therefore, we use genetic algorithm to optimize the neural network. Details are shown in Figure 12.

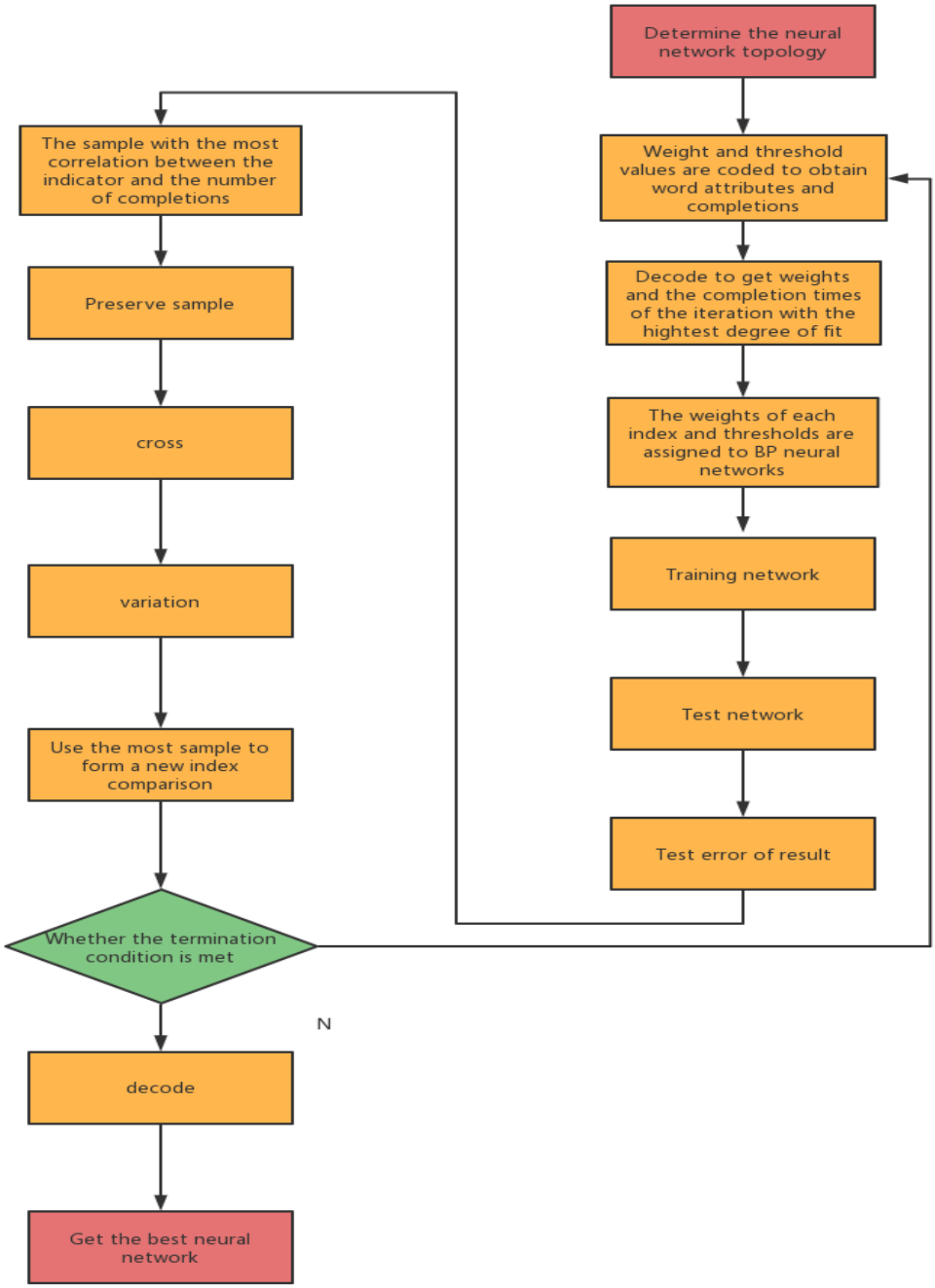


Figure 12: BP Neural Network based on Genetic Algorithm

The weights of each indicator in the hide and output layers and the number of completions with the highest fit when this iteration is performed are adjusted as follows:

$$\omega_{ij} = \omega_{ij} - \mu_1 * \frac{\partial E(\omega,b)}{\partial \omega_{ij}} = \omega_{ij} - \mu_1 * \vartheta_{ij} * x_i \tag{12}$$

$$b_j = b_j - \mu_2 * \frac{\partial E(\omega,b)}{\partial b_j} = b_j - \mu_2 * \vartheta_{ij} \tag{13}$$

Similar adjustments occur in the input and hide layers, as follows:

$$\omega_{ki} = \omega_{ki} - \mu_1 * \frac{\partial E(\omega,b)}{\partial \omega_{ki}} = \omega_{ki} - \mu_1 * \vartheta_{ki} * x_k \tag{14}$$

$$b_i = b_i - \mu_2 * \frac{\partial E(\omega,b)}{\partial b_i} = b_i - \mu_2 * \vartheta_{ki} \tag{15}$$

5.1.3 Results in BP Model

Using Matlab, the training set is set to 200, the test set is set to 159, and the number of iterations is 1000. The five attributes of the word are fitted to the neurons with the average number of passes, and the gradient of the error of iterative genetic decreases, gradually increasing the accuracy. Details are shown in Figure 13.

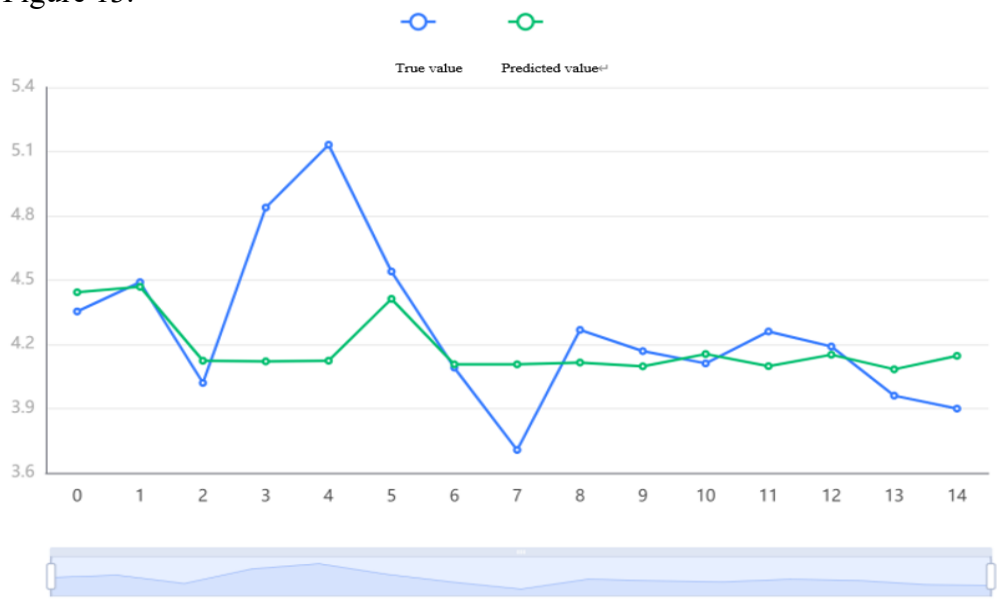


Figure 13: Relationship between True Value and Predicted Value

o consider the fitting results, mean square error and mean absolute error are introduced in this paper. The results are shown in the following Table 4.

Table 4: MSE and MAE of the training and test sets

	MSE	MAE	R^2
Training Set	0.139	0.183	0.767
Test Set	0.159	0.281	0.603

MSE (Mean Square Error): The mean square error for the five indicators and the average number of passes is 0.139 and 0.159, which is an acceptable error.

MAE (Mean Absolute Error): It reflects the actual situation of the error of the predicted values. The mean square error of the five indicators and the average number of passes is 0.183 and 0.281.

5.2. GBDT Model Regression Prediction

From the above analysis, the correlation between the difficulty word attributes and the distribution of the reported results is high. Therefore, the Gradient Boosting Decision Tree model was used to

quantify the relationship between the two since obtaining the fitted relationship, and the prediction model was successfully built. [7,8]

GBDT is an iterative decision tree algorithm that can use the negative gradients of various differentiable loss functions as residual fits and is a Boosting algorithm with a regression tree as the basis function.

Step1: Initialize $f_0(x) = argmin_\gamma \sum_{i=1}^{N} L(y_i, \gamma)$

Step2: For m =1 to M:

For i =1, 2,…,N compute

$$r_{im} = -\left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f=f_{m-1}} \tag{16}$$

Fit a regression tree to the targets r_{im} giving terminal regions $R_{jm}, j = 1,2,\dots,J_m$

For $j = 1,2,\dots,J_m$ compute

$$\gamma_{jm} = arg\min_{\gamma} \sum_{x_i \epsilon R_{jm}} L(y_i, f_{m-1}(x_i) + \gamma) \ ] \tag{17}$$

Update

$$f_m(x) = f_{m-1}(x) + \sum_{j=1}^{J_m} \gamma_{jm} I(x \epsilon R_{jm}) \tag{18}$$

Step3: Output $\hat{f}(x) = f_M(x)$

5.2.1 Regression Fitting

Based on the fitted relationships obtained by the BP neural network algorithm, [9,10]regression fits were first performed with the T1,T2,T3,T4, T5, T6, and T7 distributions to obtain the following Table 5.

Table 5: Attribute Importance Table

Attribute	Attribute Importance
The Number of Word Meaning	43.10%
length	5.80%
quantity level	4.10%
part of speech	21.60%
The number of letter repeats	25.40%

The table shows that The Number of Meanings of the Words and the Quantity Level are the two most important factors under the distribution of Completion Rates, while the Length and Part of Speech have almost no effect on the results.

The details are shown in Table 6.

Table 6: Accuracy Table

	T1	T2	T3	T4	T5	T6	T7 or more
Error Degree	29%	10%	19%	31%	15%	29%	57%
R^2	0.58	0.69	0.56	0.472	0.66	0.58	0.34

5.2.2 Distribution Forecast

Data collected for EERIE: the word has only the meaning of fear, part of speech is adjective (replaced by the number 2), the number of searches is less than 500 (level 1), the word length is 5 and 3 letters of E are repeated. Details are shown in Figure 14.

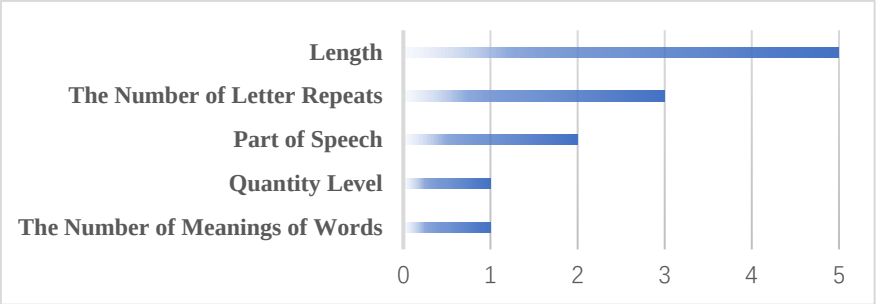


Figure 14: EERIE Word Attributes

The data is substituted into the GBDT machine learning prediction model to obtain the prediction probability. The details are shown in Table 7.

Table 7: GBDT prediction probability table

	T1	T2	T3	T4	T5	T6	T7
Parameter	0.0000084	1.39847	4.08274	14.000912	24.486961	36.99813	17.99746
Probability	0%	1.4%	4.1%	14.2%	24.6%	37.2%	18.1%

Therefore, the expected value of the word EERIE is 5.399 times guessed.

6. Conclusions

This paper models the different characteristics and final performance of each word in order to make predictions about future data. In the process, after validation analysis, we found that the LSTM (Long Short Term Memory) model can be used to simulate changes in the number of reports over time and to make predictions about future data. The forecast finds that the number of results reported on March 1, 2023, is in the range (26,556,37,535), with an error rate of 17%. Secondly, five definitions were set for word attributes (length, number of polysemous words, generality, part of speech, number of repeated letters), and Pearson correlation coefficient method was used to deduce the correlation between the scores reported by the difficulty model. At the same time, BP neural network was used to explore the correlation between word difficulty and result distribution, to develop a multi-word measure related to the percentage of reports, and GBDT machine learning method was used to quantify this relationship, specifically predicting "EERIE" data, and it was found that the average number of correct guesses was 5.399. Compared with ARMA, the traditional lexical analysis method based on time series, LSTM method is adopted in this paper, which is more accurate. Meanwhile, Pearson correlation coefficient method is used for correlation deduction, and the results obtained can provide reference for future game development and update Settings.

References

[1] Min Z. A study of the number of Wordle users and experience predictions[J]. Academic Journal of Mathematical Sciences,2023,4(2).

[2] Shi X,Guang J,Shao L. Wordle data analysis based on time series analysis model[J]. Academic Journal of Mathematical Sciences,2023,4(2).

[3] Ji-Bi-Cong & Zhang-Pin-I. (2022). Financial time series forecasting based on ARIMA-LSTM model. Statistics and Decision Making (11), 145-149.

[4] Yang, Yueheng & Zuo, Shuai. (2022). Processing of foundation pit monitoring data based on time series prediction model. Journal of Jiangsu Construction Vocational Technology College (02), 14-18.

[5] Wang M, Xu YH & Zhu XJ. (2022). Optimized BP neural network based on improved genetic algorithm for prediction model of diabetic complications. Computers and Modernization (11), 69-74.

[6] Wenchao F,Sheng C,Shuo S. Dynamic characteristics and real-time control of a particle-to-sCO2 moving bed heat exchanger assisted by BP neural network[J]. Energy,2022,256.

[7] CAI Juan, LAN Yaxun, Liu Yuan. Research on LAN intrusion Location and detection based on GBDT optimization algorithm [J/OL]. Computer measurement and control: 1-9 [2023-09-04].

[8] ZhA Wei, Dong Yanwu, Jiang Zhouhua et al. Based on vacuum consumable GBDT algorithm ingot manganese content prediction model [J]. China metallurgical, 2023 (7): 107-114.

[9] Yu Fangfang. Research on construction dynamic schedule control based on BP neural network prediction and fuzzy earned value method [D]. Dongbei University of Finance and Economics,2020.

[10] Li Ping, Zeng Lingke, Yue Anze et al. Design of BP Neural Network prediction System based on MATLAB [J]. Computer Applications and Software,2008(04):149-150+184.