Problem Chosen	2023 MCM/ICM Summary Sheet	Team Control Number
C		2309601

A Comprehensive Study on Wordle Coverage

Summary

Twitter provided a user feedback channel to collect almost a year's worth of feedback data from Wordle, including information on Date, Word, Number in hard mode and more. We applied mathematical formulas and machine learning models to rationalize the requests made by the New York Times.

Firstly, we looked at the data trend and fitted the data before and after it using two segmentation functions. It was determined a Fourier polynomial was used for the first segment, and an exponential function was used for the second segment, with **the final segmentation function (Q1 Segmentation Function Y Model)** shown in Equation (11). **Method 1 infers the interval for the Number of reported results on March 1, 2023 is [1320,2542].** The second idea is constructing the temporal relationship dataset to obtain the LSTM model. The relative error of **the final time-series model Q1 LSTM Model** is only 5% and **method 2 infers the interval for the Number of reported results on March 1, 2023 is [1313, 1454].** The second sub-question is to calculate the correlation coefficient of a certain attribute of a word to PSRTW (the percentage of scores reported that were played in Hard Mode) as an indication of relevance. By plotting the correlation heat map (Figure 10), it was found the attribute had a low correlation with PSRTW, i.e., the attribute we selected had almost no effect on PSRTW.

Then, we quantified the words in terms of letter frequencies. The time series is considered to be an equivariant sequence. A deep learning approach was chosen to establish the link between the input as a data series and the word and the output as seven percentages using BP neural network, which was optimized using genetic algorithm. The relative error of the **Q2 GA-BP Model** is shown in Table 5. **We predict the correlation of the word "EERIE" on 1 March 2023 will be relevant in [0,4,27,33,22,7,3] (%)** on March 1, 2023. Our level of confidence depends on the percentage of the difference between the mean and 1 of the relative errors other than 1 try and 7 or more tries (X). This means **we are 63% confident that the results are accurate.**

Next, we chose 3 tries, 4 tries and 5 tries percentage as the scale of difficulty. **The model Q3 K-means Model** was used to classify all words into ABCD classes from hard to easy by calculating the Euclidean distance between the center of the cluster and the origin. **The difficulty of the Q3 K-means Model for predicting "EERIE" is B**. In addition, we trained a one-hot coding decision tree and a random forest classifier (**Q3 DT RF Model**) for 11 common lexical items with a degree of difficulty. **The results of both classifiers were: B.** The predictions of both methods were consistent.

Finally, the number of reported results will likely be related to weekdays or weekends. We found no correlation between the Number of reported results and weekends at all times. The Number of days of the week was found to be more negatively correlated with scores from 150 days after the statistics (when it had stabilized).

Keywords: temporal model; BP neural network; K-means clustering; machine learning

1 Introduction 4
1.1 Problem Background 4
1.2 Restatement of the Problem 4
1.3 Literature Review 4
1.4 Our Work 5
2 Assumptions 5
3 Notations 5
4 Task1: Mapping the relationship between date and Number of reported results 6
4.1 Problem Analysis 6
4.2 The Establishment of Model 7
4.2.1 Q1 Segmentation Function Y Model 7
4.2.2 Q1 LSTM Model 8
4.2.3 Attribute and Percentage Correlation Analysis 8
4.3 The Solution of Model 8
4.3.1 Q1 Segmentation Function Y Model 8
4.3.2 Q1 LSTM Model 10
4.3.3 Attribute and Percentage Correlation Calculation and Analysis 12
5 Task2: Create a relational mapping of dates and word pairs to percentages 13
5.1 Problem Analysis 13
5.2 The Establishment of Model 13
5.2.1 Q2 GA-BP Model 13
5.3 The Solution of Model 14
5.3.1 Q2 GA-BP Model 14
6 Task3: Establishing a mapping of word difficulty to word attributes 17
6.1 Problem Analysis 17
6.2 The Establishment of Model 18
6.2.1 Q3 K-means Model 18
6.3 The Solution of Model 18
6.3.1 Q3 K-means Model 18
6.3.2 Q3 DT RF Model 20
6.3.3 Analysis of results 20
7 Task4: Digging for other interesting information 20

8 Error Analysis 21
9 Model Evaluation and Further Discussion 21
9.1 Strengths 21
9.2 Weaknesses 22
9.3 Further Discussion 22
10 Memorandum 23
References 25

1 Introduction

1.1 Problem Background

Wordle, a traditional word game, is loved for its unique game mechanics and interface design and has received widespread attention worldwide [1]. The characteristics of text information and interaction have prompted more and more players to try Wordle. However, the number of daily game passers fluctuates in different magnitudes due to the limitations of word characteristics. To analyze the relevant factors affecting the number of players and to study the development trend of this game, it is necessary to establish a suitable mathematical model to predict the development trend of this game.

1.2 Restatement of the Problem

Considering the background information and restricted conditions identified in the problem statement, we need to solve the following problems:

- **Problem 1**: Develop a mathematical model to explain the quantitative changes in reported outcomes and use the model to obtain a prediction interval for the reported outcomes on March 1, 2023, while analyzing the effect of word attributes on the percentage of participants in the difficulty model.
- **Problem 2**: Develop a model that can predict the relevant percentage of future dates, analyze the uncertainty between the model and the prediction results, and give specific examples for the word "EERIE."
- **Problem 3**: Develop a model that can classify words according to their difficulty, identify the attributes of the given the word associated with each classification, determine the difficulty of the word "EERIE" in this model, and discuss the accuracy of the classification model.
- **Problem 4**: List and describe some other exciting dataset features.

1.3 Literature Review

With the popularity and application of Internet technology, online games are increasingly becoming an essential component of people's daily life, showing considerable growth potential[2]. Currently, some methods can be better applied to predict the growth trend of game player numbers [3,4]. Africa et al. [5] developed a survival integration model to predict player churn using the model and combined it with experiments to demonstrate the method's effectiveness in improving the accuracy of traditional analysis (e.g., Cox regression). Vafeiadis et al. [6] summarized several machine learning methods applied to customer churn prediction, including artificial neural networks, SVM, Etc., to validate and evaluate the applicability of each type of method to the problem in question.

1.4 Our Work

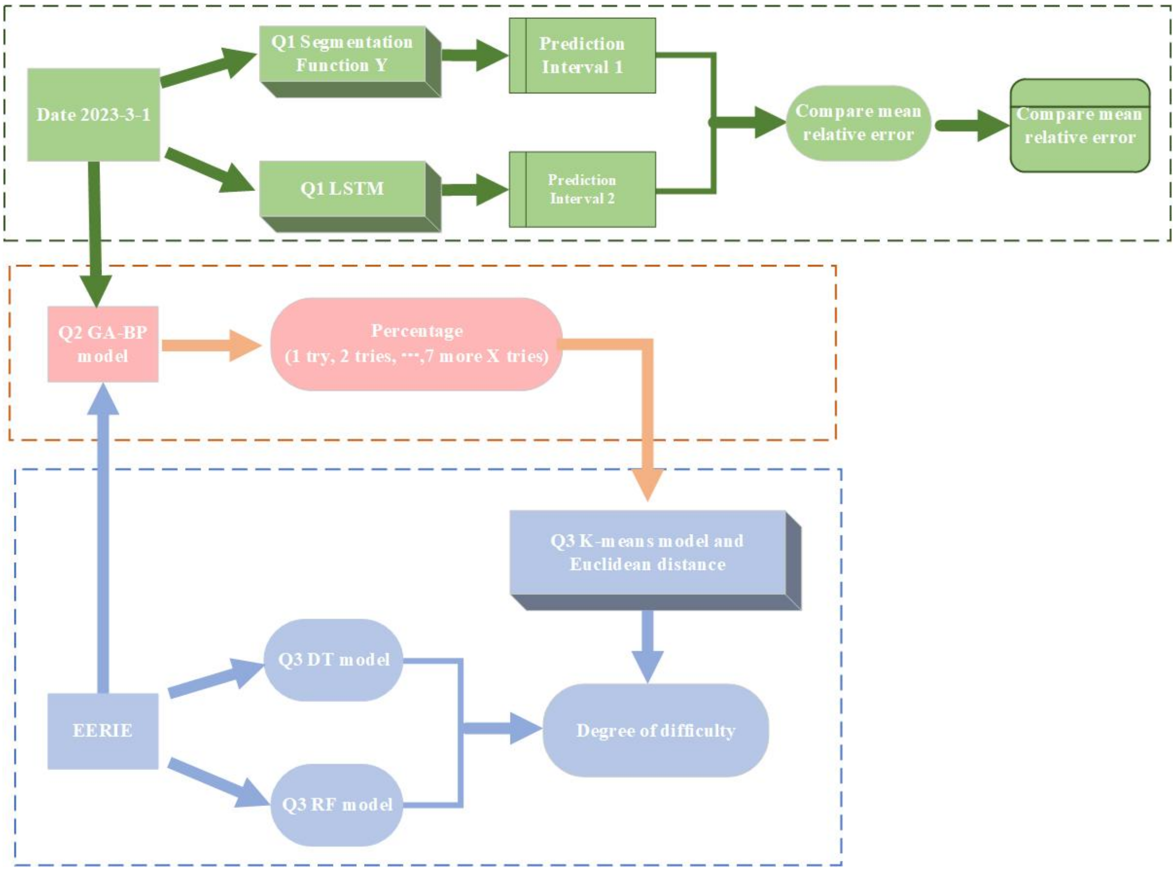

Date 2023-3-1
Q1 Segmentation Function Y
Prediction Interval 1
Q1 LSTM
Prediction Interval 2
Compare mean relative error
Compare mean relative error
Q2 GA-BP model
Percentage (1 try, 2 tries, ···,7 more X tries)
Q3 K-means model and Euclidean distance
Q3 DT model
EERIE
Degree of difficulty
Q3 RF model

Figure 1: Overall structure of our work

2 Assumptions

In order to facilitate the establishment of the optimal investment strategy model, we make the following assumptions and simplifications according to the actual situation and classical theory.

Assume that the average relative error indicates how good any one model is.

Assume there is no conflict between the quantification of words by frequency of letter occurrence in question 2 and the existence of a relationship between filling letters in the game's rules.

Assume that the percentage of 3 4 5tries in question 3 has the most excellent effect on the degree of difficulty.

Assume that the "Number of reported results" in question 4 plateaus after 150 days from January 7, 2022. There are no more dramatic ups and downs.

3 Notations

The key mathematical notations used in this paper are listed in Table 1.

Table 1: Notations used in this paper

Symbol	Description	Unit
x	Date	
y	Number of reported results	
$y^{'}$	Predicted number of reports results	
W	Error (absolute value of the subtraction of actual and predicted values)	
$w^{'}$	Relative Error	
θ	Average relative error	
epoch	Number of training sessions	
Neu_num	Number of neurons in LSTM layer	
n_1	Number of neurons in the input layer	
n_2	Number of hidden layer neurons	
$X_i(i=1,...,n)$	The *i-th* object	
$C_k\ (k=1,...,n)$	The kth clustering center	
$X_{it}\ (1 \le t \le m)$	The tth attribute of the *i-th* object	
C_{jt}	*The t-th* attribute of the *j-th clustering* center	
S_k	The *k-th* class cluster	
$C_l\ (1 \leqslant l \leqslant k)$	The center of a cluster	
$\|S\|_l$	Number of objects in the *l-th* class cluster	
$X_i\ (1 \leqslant i \leqslant \|S_l\ )\|$	The *i-th* object in the *l-th* class cluster	

Threshold date: The date when the trend of the Number of reported results changes with the date (rising then falling) turns, set to February 4, 2023, i.e., 29 days after the statistic date.

SQx (Sub question x): The x-th subproblem in the process of analysis of a problem.

PSRTW (the percentage of scores reported that were played in Hard Mode): this property of a word is defined as the Number in hard mode divided by the Number of reported results.

4 Task1: Mapping the relationship between date and Number of reported results

4.1 Problem Analysis

Question 1 can be divided into two sub-questions as follows：

SQ1: Develop a model to describe the relationship between "Number of reported results" and "Data" and predict a possible interval for "Number of reported results" on March 1, 2023. we reported the results" on March 1, 2023.

SQ2: Study whether specific properties of words affect the percentage of scores reported that were played in Hard Mode, and explain why.

SQ1 Analysis: This question is about finding the correspondence between variables and time. We planned to use two methods to solve the problem and compare the results produced by the two methods. Method 1 first needs to visualize the data (see Figure 1 for the trend of the data), observe the trend of the data with the date, and then try to fit it with a suitable mathematical formula, and then launch the prediction space of "Number of reported results" for the specified date. Method 2 uses the known data " Number of reported results," constructs the time-series relationship data set by differencing adjacent terms, trains the LSTM time-series model conforming to SQ1, and performs inference.

SQ2 analysis: The focus is on calculating the correlation coefficient of a particular word attribute to PSRTW (the percentage of scores reported that were played in Hard Mode) to illustrate the correlation. The attributes of the words need to be specified or defined. We selected 11 common lexical properties of words and the frequency of words (related to the frequency of letters in the document). We observed the correlation between the selected property and PSRTW by plotting the correlation heat map.

4.2 The Establishment of Model

4.2.1 Q1 Segmentation Function Y Model

Based on the problem description, we first visualize the trend of "Number of reported results" over time, and the results are shown in Figure 2.

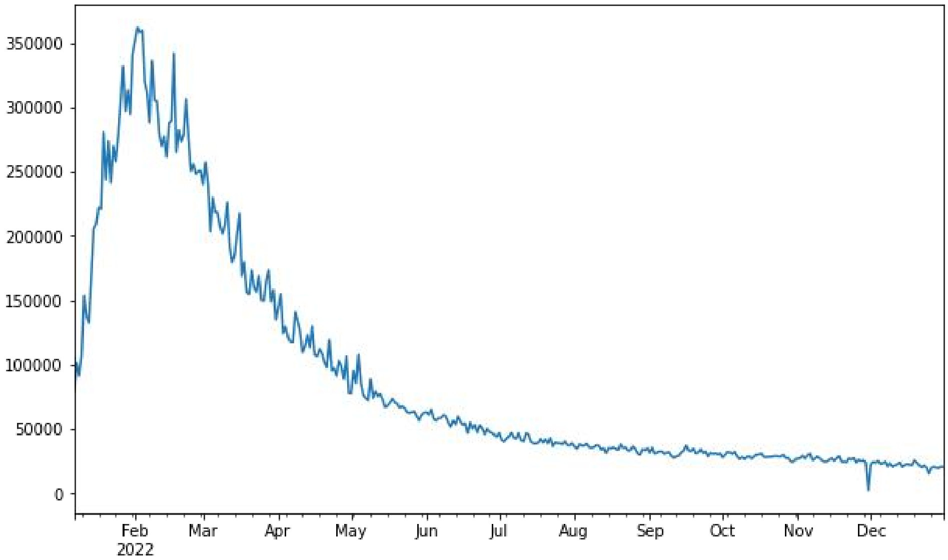


Figure 2: Number of reported results as a function of date

It can be noticed that the number of participants in this game increases sharply with the development of time, reaching the maximum number of participants in this game on the threshold day and decreasing gradually thereafter. In order to objectively describe the above situation, we consider a segmentation function to quantify it. The whole process can be divided into two stages: an increasing and a decreasing period of "Number of reported results".

4.2.1.1 Growth period modeling

Let the value of "Number of reported results" be y and the value of "Date" be x. To facilitate the calculation, let x=1 when the time is January 7, 2022, x=2 when the time is January 8, 2022, and so on. When the time is January 8, 2022, x=2, and so on. Figure 3 shows the scatter plot for this interval, and a Fourier fit is made to the scatter plot. Let the "growth period" function be.

$$y = a_0 + a_1 * \cos(x * w) + b_1 * \sin(x * w) \tag{1}$$

4.2.1.2 Downturn modeling

Figure 4 shows a scatter plot of the data from February 5, 2022 to December 31, 2022. Analysis of the data reveals that the scatter plot follows the trend of the exponential function. An exponential function (exponential) was used to fit the scatter plot for this period. Let the function be

$$y = a * \exp(b * x) \tag{2}$$

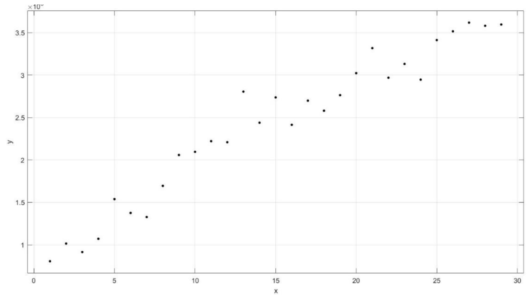


Figure 3: Scatterplot over the growth period

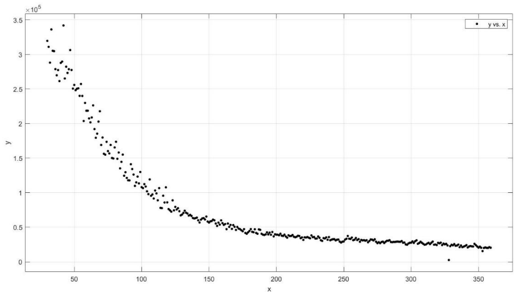


Figure 4: Scatterplot over the decline period

4.2.2 Q1 LSTM Model

Long Short Term Memory (LSTM) model is essentially a specific form of Recurrent Neural Network (RNN). It is an "end-to-end" model that uses the original data as model input, automatically learns the data features, and outputs predictions [7]. The LSTM model solves the problem of RNN short-term memory by adding gates to the RNN model, allowing the recurrent neural network to really use the long-range temporal information effectively. temporal information. The specific structure is shown in the figure below.

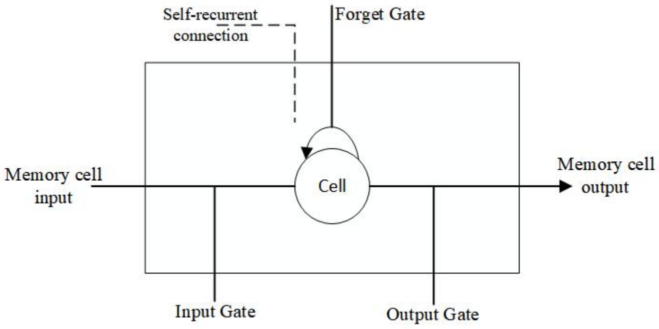


Figure 5: LSTM concept diagram

4.2.3 Attribute and Percentage Correlation Analysis

First, the PSRTW of all words is calculated, then the lexical properties of all words are found, and one-hot encoding is performed for 11 common lexical properties (1 for a word belonging to that lexical property and 0 for the opposite). In this way, an 11-dimensional numerical feature of the word can be obtained. In addition, word frequency can also be defined as a word attribute. The word frequency is the sum of the number of occurrences of the 5 letters within the word in the document. Finally, a correlation coefficient matrix with PSRTW and correlation percentages (1 try,2 tries...) is calculated for these 12 features. The correlation coefficients are used to determine the relevance.

4.3 The Solution of Model

4.3.1 Q1 Segmentation Function Y Model

4.3.1.1 Growth period model solving

The data were brought into equation (1) for Fourier fitting. The results of the confidence interval of the function parameters within 95% were obtained as

$$a_0 = -1.932\times10^{11}\ (-9.321\times10^{17}, 9.321\times10^{17}) \tag{3}$$

$$a_1 = 1.932\times10^{11}\ (-9.321\times10^{17}, 9.321\times10^{17}) \tag{4}$$

$$b_1 = -3.546\times10^{8}\ (-8.555\times10^{14}, 8.555\times10^{14}) \tag{5}$$

$$w = -4.421\times10^{-5}\ (-106.6, 106.6) \tag{6}$$

Bringing the parameters into equation (1), the functional expression of the growth period can be obtained as:

$$\begin{aligned} Y &= (-1.932\times10^{11}) + (1.932\times10^{11}) * \cos(x * (-4.421\times10^{-5})) \\ &+ (-3.546\times10^{8}) * \sin(x * (-4.421\times10^{-5})) \end{aligned} \tag{7}$$

The fitting results are shown in Figure 6.

In order to measure the fit of the model and to show that the model is the best, we calculated the "R-Squared" and "Adjusted R-Squared" for several models. They were used to evaluate the models.

The evaluation metrics of the different models are shown in Table 2. In the first half, the R-square and Adjusted R-Squared values of the Fourier fitting model are closer to 1 than the other models and have the best fitting effect.

Table 2: R-squared and Adjusted R-Squared for each model in the growth period

Function	R-square	Adjusted R-square
Fourier	0.9584	0.9535
Gauss	0.9471	0.9432
Polynomial	0.9398	0.9352

4.3.1.2 Down period model solving

The data were brought into equation (2) for the exponential function fit. The results of the confidence interval of the function parameters within 95% were obtained as

$$a = 4.673\times10^{5}\ (4.528\times10^{5}, 4.818\times10^{5}) \tag{8}$$

$$b = -0.0131\ (-0.0135, -0.0127) \tag{9}$$

That is, the functional expression for the growth period can be obtained as

$$y = (4.673\times10^{5}) * \exp((-0.0131) * x) \tag{10}$$

The fitting results are shown in Figure 7, and the values of R-squared and Adjusted R-Squared for different models are shown in Table 3. It can be seen that the values of R-squared and Adjusted R-Squared for the exponential function fitting model in the latter half are closer to 1 compared to the other models, indicating that the fitting effect best.

Table 3: R-squared and Adjusted R-Squared for each model in the decline period

Function	R-square	Adjusted R-square
Exponential	0.9610	0.9609
Power	0.9527	0.9520
Fourier	0.9251	0.9248

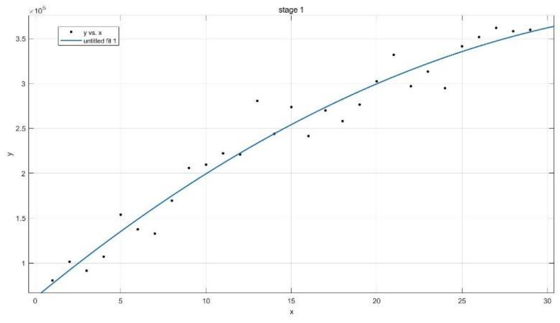


Figure 6: Growth period Fourier fit

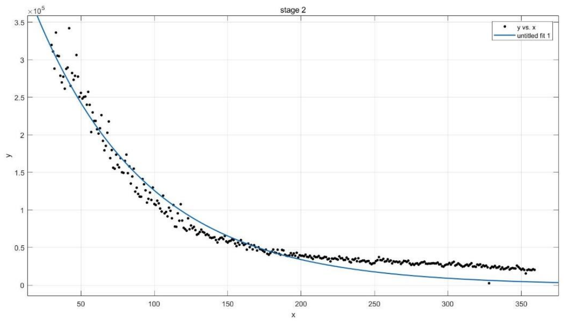


Figure 7: Exponential fit for the decline period

4.3.1.3 Integrated solution results and prediction intervals

In summary, the segmentation function created can be obtained as

$$y=\begin{cases}(-1.932\times10^{11})+(1.932\times10^{11})*\cos(x*(-4.421\times10^{-5}))+(-3.546\times10^{8})*\sin(x*(-4.421\times10^{-5}))\ \mathrm{x}\geq1,\mathrm{x}\leq29\\(4.673\times10^{5})*\exp((-0.0131)*x)\ \mathrm{x}\leq30\end{cases} \tag{11}$$

Let the actual "Number of reported result" value be y, the predicted "Number of reported result" value be y' , the error be w, the relative error be w' , and the average relative error be θ , so, the average relative error of

$$w=\left|y'-y\right| \tag{12}$$

$$w'=\frac{w}{y} \tag{13}$$

$$\theta=\frac{\sum_{1}^{359}w'}{359} \tag{14}$$

The average relative error is calculated to be θ=0.316449587. This error is very small and within the acceptable range.

From the above analysis, it can be seen that when the time is March 1, 2023, x=419, and brought into equation (11), we can get y=1931.1, that is, the number of reported results on March 1, 2023 is 1931. In addition, the relative error of the model θ=0.316449587, so the prediction interval of the number of reported results can be calculated as

$$Y_1=y*(1+\theta) \tag{15}$$

$$Y_2=y*(1-\theta) \tag{16}$$

Bringing in the values of y and θ, the calculation can be obtained as Y_1=2542.19, and Y_2 =1320. The prediction interval that yields the number of reported results on March 1, 2023 is [1320,2542].

4.3.2 Q1 LSTM Model

4.3.2.1 Model instantiation

"Number of reported results" is a set of time series, which implies much information about time. We will follow three steps to adapt the LSTM model to SQ2.

1. Preprocessing: The sequence of the "Number of reported results" is different in constructing a supervised learning dataset. The current difference is divided into the label of

the previous difference and regularized to [-1,1] for easy LSTM training.

2. Training: Training to obtain the Q1 LSTM Model.

3. Forecast: The forecast requires an initial differential input, which we choose as -824=20380-21204. This is the difference between "manly" on December 31, 2022, and "molar" on December 30, 2022. This is the difference between the word "manly" on December 31, 2022, and "molar" on December 30, 2022. After inputting into the LSTM, the "Number of reported results" for January 1, 2023, is output. This value is recorded, and the difference is made with the "Number of reported results" (20380) of the word "manly" on December 31, 2022. And then input it into the LSTM. The value of "Number of reported results" on March 1, 2023, is obtained.

The processing flow for SQ2 is shown in Figure 8.

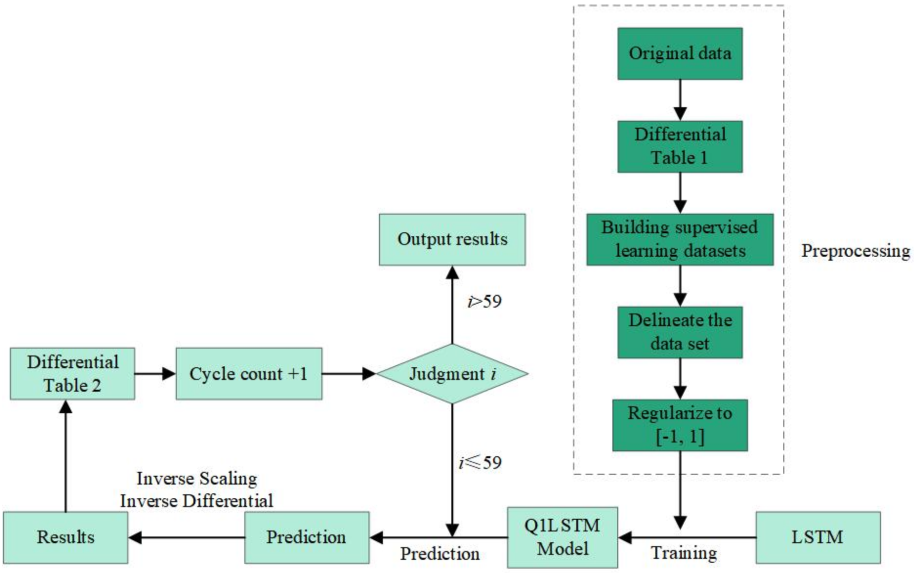


Figure 8: How to generate a Q1 LSTM Model

Figure 8 is illustrated as follows.

Raw data: The "Number of reported results" sequence is in the attachment.

Difference table 1: stores the current date minus the previous day in the original data. Since January 8, 2022, the first difference is 101503-80630=20873, and so on, for 358 items.

Difference table 2: Calculates the difference between the current and previous items according to the Results table and stores the different sequences of the predicted results with 60 items. Difference table 2 has the first difference value [1203, -824], which is used to start the program loop.

Supervised learning dataset: a two-dimensional array constructed by dividing the antecedent differences into inputs and the posterior differences into outputs. For example, [[20873, -10026], [-10026, 15657]].

4.3.2.2 Analysis of results

We use the average relative error θ of the LSTM model on the test set (see Eq. 12 13 14) to describe the model's accuracy. The hyperparameters that make the average relative error θ the smallest are selected through many experimental comparisons, and the appropriate prediction interval is calculated from this (see Equation 15 16). The experimental data are shown in Table 4.

Table 4: Comparison of LSTM hyperparameters and mean relative error *θ*

batch_size	epoch	Neu_num	θ	Forecast interval
1	2	2	0.0581	[4876,5478]
1	2	3	0.0516	[10052,11146]
1	3	3	0.0608	[9384, 10598]
1	3	4	0.0512	[1313, 1454]
1	4	4	0.0563	[1812,2028]

It can be seen that the LSTM *θ* trained with the number of training times (epoch) = 3 and the number of LSTM layer neurons (Neu_num) = 4 is the smallest at 5.12%. It is much smaller than the 31.64% of the Q1 Segmentation Function Y Model. So we choose the prediction interval [1313, 1454] as the final answer.

Figure 9 shows the performance of this model on the training set, and it is very intuitive to find that the fit is much stronger than that of the Q1 Segmentation Function Y Model.

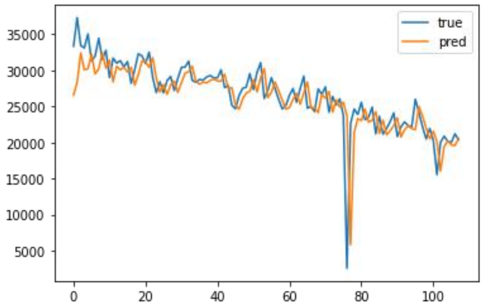
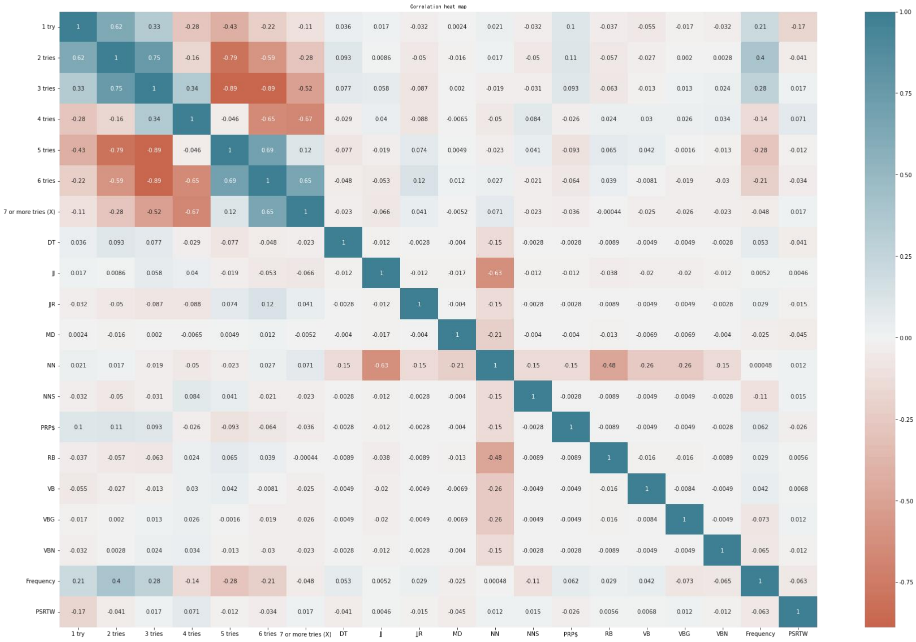


Figure 9: The fit of Q1 LSTM Model on the test set

4.3.3 Attribute and Percentage Correlation Calculation and Analysis

The matrix of correlation coefficients with PSRTW and correlation percentages (1 try, 2 tries...) was calculated for the 12 features, as shown in Figure 10.



	1 try	2 tries	3 tries	4 tries	5 tries	6 tries	7 or more tries (X)	DT	JJ	JJR	MD	NN	NNS	PRP$	RB	VB	VBG	VBN	Frequency	PSRTW
1 try	1	0.62	0.33	-0.28	-0.43	-0.22	-0.11	0.036	0.017	-0.032	0.0024	0.021	-0.032	0.1	-0.037	-0.055	-0.017	-0.032	0.21	-0.17
2 tries	0.62	1	0.75	-0.16	-0.79	-0.59	-0.28	0.093	0.0086	-0.05	-0.016	0.017	-0.05	0.11	-0.057	-0.027	0.002	0.0028	0.4	-0.041
3 tries	0.33	0.75	1	0.34	-0.89	-0.89	-0.52	0.077	0.058	-0.087	0.002	-0.019	-0.031	0.093	-0.063	-0.013	0.013	0.024	0.28	0.017
4 tries	-0.28	-0.16	0.34	1	-0.046	-0.65	-0.67	-0.029	0.04	-0.088	-0.0065	-0.05	0.084	-0.026	0.024	0.03	0.026	0.034	-0.14	0.071
5 tries	-0.43	-0.79	-0.89	-0.046	1	0.69	0.12	-0.077	-0.019	0.074	0.0049	-0.023	0.041	-0.093	0.065	0.042	-0.0016	-0.013	-0.28	-0.012
6 tries	-0.22	-0.59	-0.89	-0.65	0.69	1	0.65	-0.048	-0.053	0.12	0.012	0.027	-0.021	-0.064	0.039	-0.0081	-0.019	-0.03	-0.21	-0.034
7 or more tries (X)	-0.11	-0.28	-0.52	-0.67	0.12	0.65	1	-0.023	-0.066	0.041	-0.0052	0.071	-0.023	-0.036	-0.00044	-0.025	-0.026	-0.023	-0.048	0.017
DT	0.036	0.093	0.077	-0.029	-0.077	-0.048	-0.023	1	-0.012	-0.0028	-0.004	-0.15	-0.0028	-0.0028	-0.0089	-0.0049	-0.0049	-0.0028	0.053	-0.041
JJ	0.017	0.0086	0.058	0.04	-0.019	-0.053	-0.066	-0.012	1	-0.012	-0.017	-0.63	-0.012	-0.012	-0.038	-0.02	-0.02	-0.012	0.0052	0.0046
JJR	-0.032	-0.05	-0.087	-0.088	0.074	0.12	0.041	-0.0028	-0.012	1	-0.004	-0.15	-0.0028	-0.0028	-0.0089	-0.0049	-0.0049	-0.0028	0.029	-0.015
MD	0.0024	-0.016	0.002	-0.0065	0.0049	0.012	-0.0052	-0.004	-0.017	-0.004	1	-0.21	-0.004	-0.004	-0.013	-0.0069	-0.0069	-0.004	-0.025	-0.045
NN	0.021	0.017	-0.019	-0.05	-0.023	0.027	0.071	-0.15	-0.63	-0.15	-0.21	1	-0.15	-0.15	-0.48	-0.26	-0.26	-0.15	0.00048	0.012
NNS	-0.032	-0.05	-0.031	0.084	0.041	-0.021	-0.023	-0.0028	-0.012	-0.0028	-0.004	-0.15	1	-0.0028	-0.0089	-0.0049	-0.0049	-0.0028	-0.11	0.015
PRP$	0.1	0.11	0.093	-0.026	-0.093	-0.064	-0.036	-0.0028	-0.012	-0.0028	-0.004	-0.15	-0.0028	1	-0.0089	-0.0049	-0.0049	-0.0028	0.062	-0.026
RB	-0.037	-0.057	-0.063	0.024	0.065	0.039	-0.00044	-0.0089	-0.038	-0.0089	-0.013	-0.48	-0.0089	-0.0089	1	-0.016	-0.016	-0.0089	0.029	0.0056
VB	-0.055	-0.027	-0.013	0.03	0.042	-0.0081	-0.025	-0.0049	-0.02	-0.0049	-0.0069	-0.26	-0.0049	-0.0049	-0.016	1	-0.0084	-0.0049	0.042	0.0068
VBG	-0.017	0.002	0.013	0.026	-0.0016	-0.019	-0.026	-0.0049	-0.02	-0.0049	-0.0069	-0.26	-0.0049	-0.0049	-0.016	-0.0084	1	-0.0049	-0.073	0.012
VBN	-0.032	0.0028	0.024	0.034	-0.013	-0.03	-0.023	-0.0028	-0.012	-0.0028	-0.004	-0.15	-0.0028	-0.0028	-0.0089	-0.0049	-0.0049	1	-0.065	-0.012
Frequency	0.21	0.4	0.28	-0.14	-0.28	-0.21	-0.048	0.053	0.0052	0.029	-0.025	0.00048	-0.11	0.062	0.029	0.042	-0.073	-0.065	1	-0.063
PSRTW	-0.17	-0.041	0.017	0.071	-0.012	-0.034	0.017	-0.041	0.0046	-0.015	-0.045	0.012	0.015	-0.026	0.0056	0.0068	0.012	-0.012	-0.063	1

Figure 10 Correlation matrix

Looking at this matrix, we can see that the correlation coefficients of the word attributes, i.e., the 12 numeric features, with the PSRTW and the correlation percentages (1 try, 2 tries...) are very small and rarely exceed 0.1. This indicates that our chosen attributes have almost no influence on the PSRTW and the correlation percentages (1 try, 2 tries ...).

5 Task2: Create a relational mapping of dates and word pairs to percentages

5.1 Problem Analysis

Since the game is played with continuous alphabetic input, it occurred to us that we could split the words into five letters and quantify the words using the frequency of occurrence of the letters in familiar words (Figure 11) as weights [8]. The higher the frequency of occurrence, the higher the weights, with a maximum of 26 for e and a minimum of 1 for *z*. The time series is treated as a monotonically increasing series of equal differences with a difference of 1. We choose a deep learning method to train seven models and use BP neural network to establish the connection between the input as a date sequence and word letters and the output as seven percentages. Since the initial weights of the BP neural network are randomly assigned, we use a genetic algorithm to optimize it to reduce the training time and improve the model's accuracy.

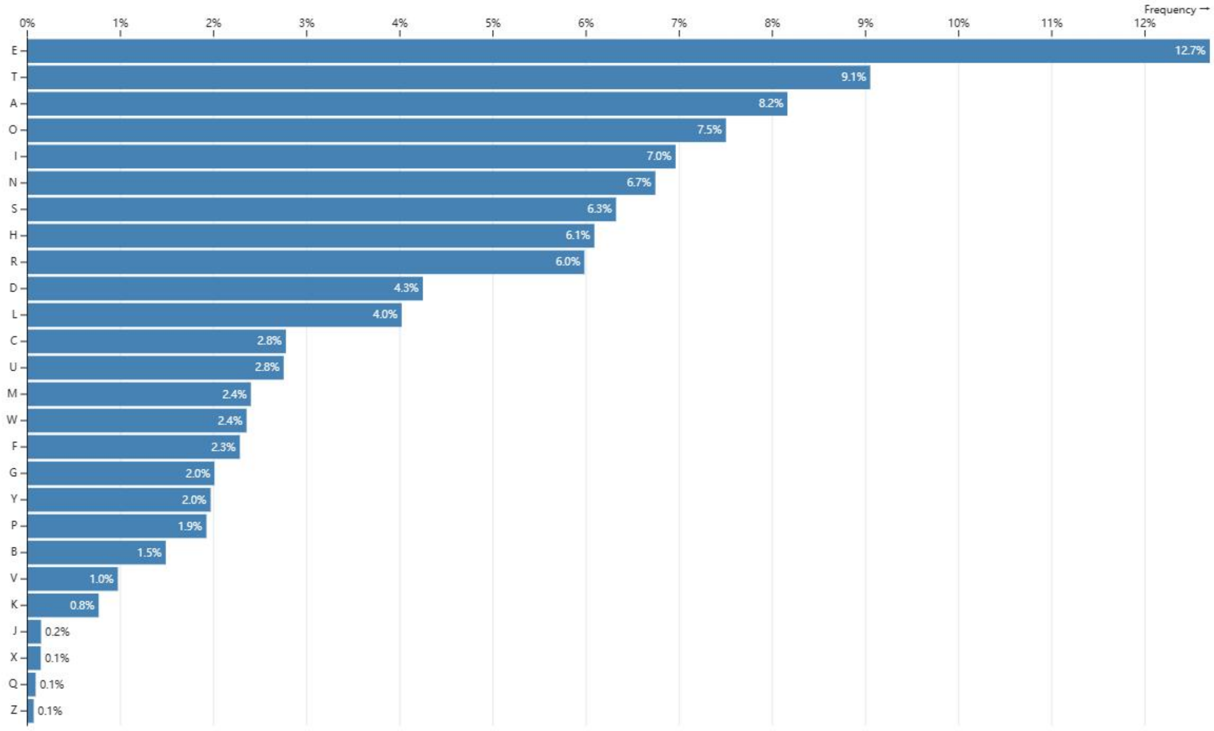


Figure 11 Relative frequency of letter use in English

5.2 The Establishment of Model

5.2.1 Q2 GA-BP Model

A neural network can be understood as an exceptionally complex function that is impossible to apply a generalized formula [9]. The neural network structure and the weight threshold determine it. The genetic algorithm is introduced to obtain the optimal initial weight threshold of the neural network. It is assumed that all solutions corresponding to the weight threshold are populations, and the individuals in the population are a set of weight thresholds. The work of the genetic algorithm is to let the individuals in this population reproduce; in the process of reproduction, on the one hand, the chromosomal crossover will occur, and new individuals will be created [10]. On the other hand, there is a probability that genetic mutations will occur and new individuals will be created. Then, by calculating the fitness of individuals, eliminating the poor-quality individuals and keeping the excellent quality individuals according to the selection operator, and allowing the reproduction process to continue, it is possible to evolve the best or superior individuals.

The overall flow of the algorithm is shown in Figure 12.

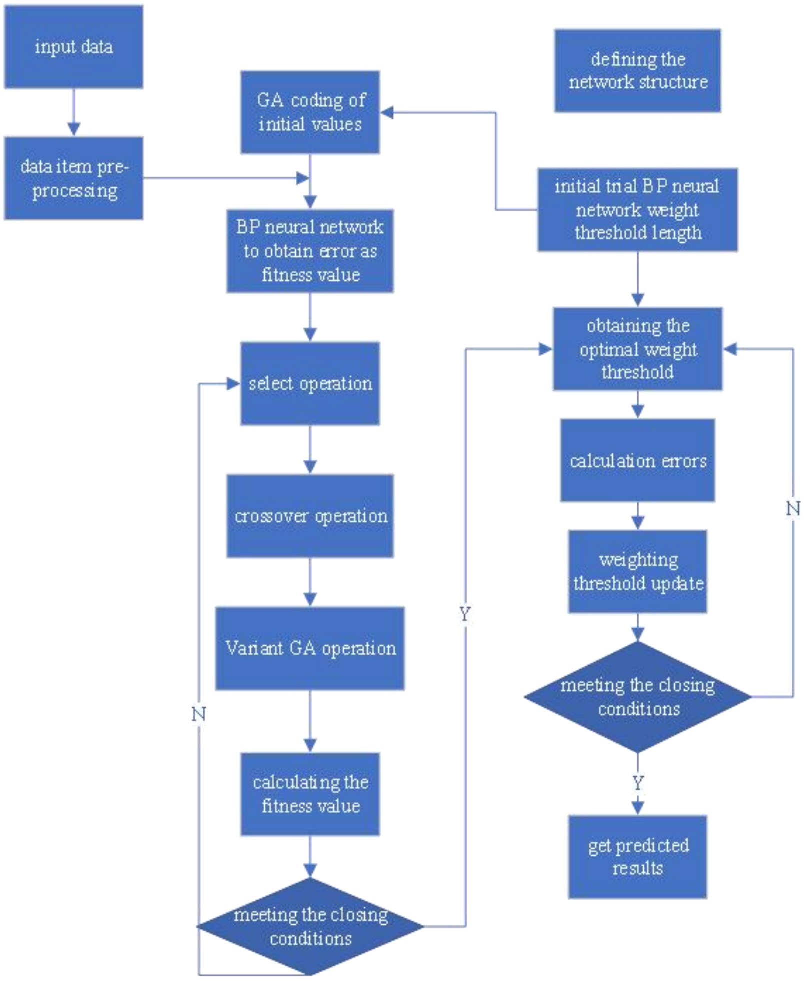

input data
data item pre-processing
GA coding of initial values
BP neural network to obtain error as fitness value
select operation
crossover operation
Variant GA operation
calculating the fitness value
meeting the closing conditions
N
Y
defining the network structure
initial trial BP neural network weight threshold length
obtaining the optimal weight threshold
calculation errors
weighting threshold update
meeting the closing conditions
N
Y
get predicted results

Figure 12: GABP algorithm flow chart

5.3 The Solution of Model

5.3.1 Q2 GA-BP Model

5.3.1.1 Model instantiation

1. Determination of chromosome length

We choose a three-layer neural network. The chromosome lengths are now calculated according to Figure 13.

n_1 and n_2 represent the number of neurons in the input and hidden layers. Where the sample has six input parameters and one output parameter, so here $n_1 = 6$. Since the number of neurons in the hidden layer and the number of neurons in the input layer have the following approximate formula.

$$n_2 = 2 * n_1 + 1 \tag{17}$$

Therefore, n_2 =13. BP neural network structure: 6-13-1 i.e., the number of nodes in the input layer, hidden layer, and output layer are 6, 13, and 1, respectively.

Total number of weights: 6*13+13*1=91 Total number of thresholds: 13+1=14

Number of parameters to be optimized by the genetic algorithm (chromosome length): 91 + 14 = 105

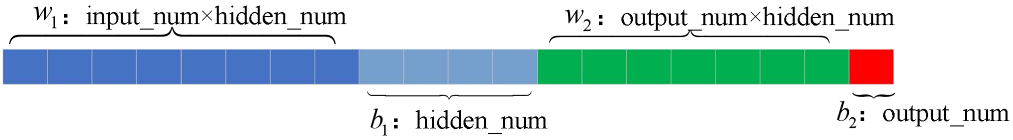


Figure 13: BP neural network parameters (chromosome form)

The genetic algorithm process is briefly described as follows.

(1) Import the data and normalize the training data with the test data. Determine the BP neural network structure, which is 6-13-1, so the total number of parameters to be optimized is 6*13+13*1+13+1=105, the chromosome length.

(2) Initialize all individuals of a population, i.e., encode the chromosomes of each individual. A set of chromosomes of population size is generated, and the chromosome storage structure is a 105-dimensional vector of real numbers. Their fitnesses are calculated separately. The average fitness of the initial population and the optimal chromosome are recorded.

(3) Start evolution, selection on the parent, crossover from parent to offspring, and mutation operation. Decoding calculates the fitness of each individual of the offspring, and the individual chromosome with the smallest fitness of the offspring is compared with the fitness of the optimal chromosome of the parent, and if it is smaller than the parent, it is replaced as the optimal chromosome, and if it is larger, it remains unchanged. The average fitness and optimal chromosome of the generation are also recorded.

(4) Continue the evolution until the preset number of evolutionary generations is reached, and the optimal chromosome is output. That is the optimal initial weight threshold combination for the BP network.

(5) The optimal weight threshold is assigned to the BP network, and the input and output data are trained. The error calculation is performed, and the weights are updated for a finite number of cycles. The maximum number of cycles is set to 200, the learning rate is 0.1, and the expected error is 0.00001. The training ends when the maximum number of cycles is reached, or the error is less than the expected error. At this point, the model is the optimal BP model.

5.3.1.2 Analysis of results

We trained seven models (Q2GA-BP Model) twice to predict seven percentages of "EERIE" on March 1, 2023. The average relative error θ is poor (Inf, probably too discrete to fit), but the rest of the model's errors are within the acceptable range. Finally, we choose the prediction result of the first training model as the final answer, i.e., the percentage of the relevance of the word "EERIE" on March 1, 2023, is [0,4,27,33,22,7,3] (%). Figure 14 shows the results of the second training of the Q2GA-BP Model for the five percentages.

Table 5 : Average relative error *θ* and prediction results of Q2 GA-BP Model

	1 try	2 ties	3 ties	4 ties	5 ties	6 ties	7 or more tries (X)
1st θ	Inf	0.5952	0.3134	0.1346	0.2673	0.6369	Inf
Percentage of 1st "EERIE"	0	4	27	33	22	7	3
2nd θ	Inf	0.6716	0.3217	0.1372	0.2852	0.7033	Inf
Percentage of 2nd "EERIE"	0	6	21	33	23	8	3

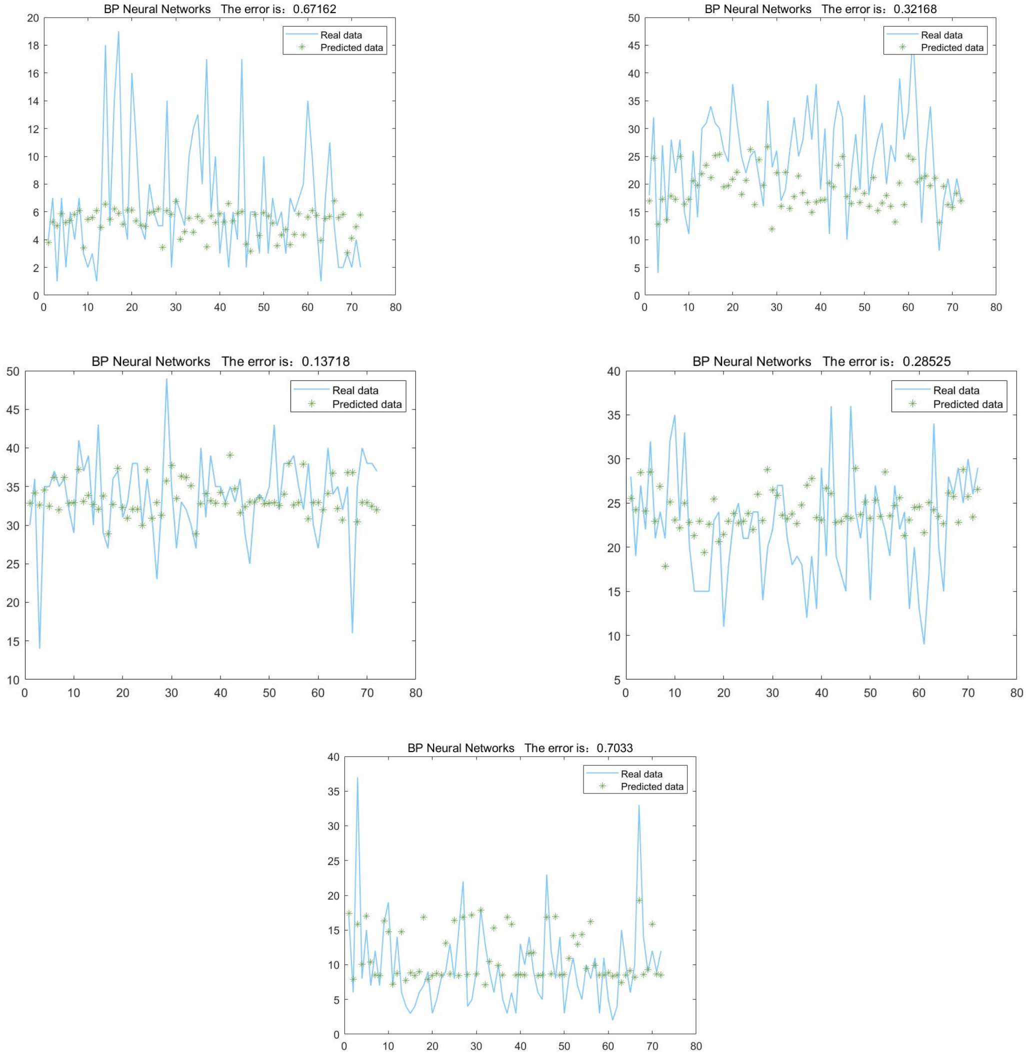


Figure 14: Fit of the 2nd training Q2GA-BP Model for 7 percentages

6 Task3: Establishing a mapping of word difficulty to word attributes

6.1 Problem Analysis

An intuitive understanding of the difficulty of a word is that the more attempts to guess the word, the more difficult it is. Seven attempt correlation percentages are an excellent evaluation scale. Also, considering that this question requires a difficulty prediction for "EERIE," the percentage of attempts for this word is given by question 2. There was an error difference in the predicted percentages in question 2, so we chose 3 tires, 4 tries and 5 tries, which had relatively small errors, as the final evaluation scale. We perform K-means clustering of all words with category four based on these three metrics to obtain the model Q3 K-means Model. This model will classify all words into four categories. However, the clustering results do not indicate the difficulty level of the words but only the existence of different categories. The clustering results have to be transformed into classification results by adding specific semantic information to the clustering results. So, we first find the clustering centers of these four categories and specify the difficulty level of each category by calculating the Euclidean distance between them and the origin (0,0,0) (a larger Euclidean distance indicates a larger percentage of more attempts and a more difficult word). The resulting semantic information classifies all words into four categories: A-difficult, B-more difficult, C-average, and D-easy. The training process of the Q3 K-means Model and the Q3 DT RF Model is shown in Figure 15, where the three percentages of the predicted "EERIE" are used as input for the Q3 K-means Model. The difficulty of "EERIE" (Figure 16). In order to discuss the accuracy of the method, we propose another method for mutual validation of the above results. For the word attributes, 11 common lexical properties were selected (DT, JJ, Etc., lexical properties only have a low correlation with the relevant percentage and should have a high correlation with the degree of difficulty). A decision tree and random forest classifier (Q3 DT RF Model) were trained for word lexicality and difficulty (Figure 15). The prediction process for "EERIE" with known word properties is shown in (Figure 16), and finally, the prediction results of the three models are compared.

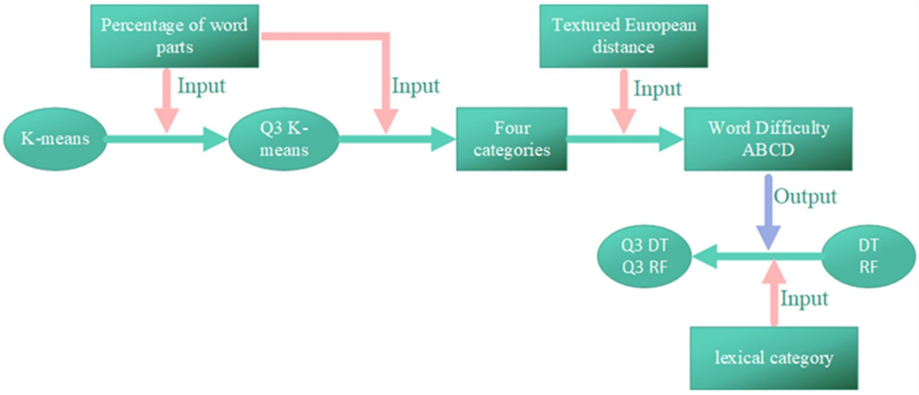


Figure 15: Training process of Q3 K-means Model and Q3 DT RF Model

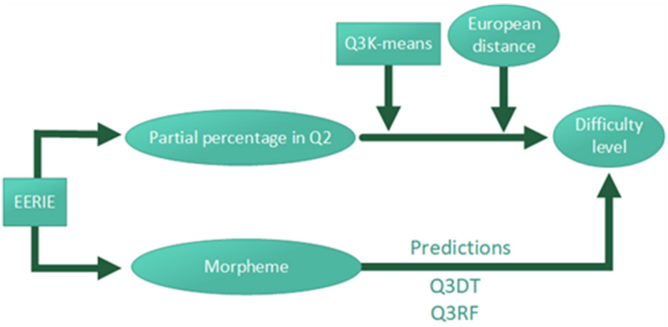


Figure 16: Prediction process of "EERIE" by Q3 K-means Model and Q3 DT RF Model

6.2 The Establishment of Model

6.2.1 Q3 K-means Model

The K-means clustering (K-means) algorithm is a representative of unsupervised clustering algorithms, whose primary function is automatically grouping similar samples into a class [11]. Assume that a given data sample, X, contains n objects $X = \{X_1 , X_2 , X_3 , ..., X_n \}$, where each object has attributes of m dimensions. The K-means algorithm aims to cluster the n objects into specified k class clusters based on the similarity between objects, where each object belongs to and only belongs to a class cluster with the smallest distance to the center of the class cluster. For K-means, it is first necessary to initialize k *cluster* centers $\{C_1, C_2 , C_3 , ..., C_k \}$, $1 < k \leq n,$ and then by calculating the Euclidean distance from each object to each cluster center, as shown in the following equation

$$dis(X_i, C_j) = \sqrt{\sum_{t=1}^{m}(X_{it} - C_{jt})^2} \tag{18}$$

In the above equation, X_i denotes the i-th object $1 \leq t \leq n$, C_j denotes the *j-th* clustering center with $1 \leq j \leq k$, X_{it} *denotes the t-th* attribute of the *i-th* object with $1 \leq t \leq m$, and C_{jt} denotes the *t-th* attribute of the j-th clustering center.

The distance of each object to each cluster center is compared in turn, and the object is assigned to the class cluster of the nearest cluster center to obtain k class clusters $\{S_1 , S_2 , S_3 , ..., S_k \}$.

The K-means algorithm defines the prototype of class clusters in terms of centers, and the class cluster center is the mean value of all objects within the class cluster in each dimension, which is calculated as follows

$$C_t = \frac{\sum_{X_i \in S_l} X_i}{|S_l|} \tag{19}$$

where C_l denotes the center of a cluster, $1 \leq l \leq k$,$|S_l|$ denotes the number of objects in the 1st class cluster, X_i denotes the *i-th* object in the 1st class cluster, $1 \leq i \leq |S_l|$.

6.3 The Solution of Model

6.3.1 Q3 K-means Model

6.3.1.1 Model instantiation

As described in the Task 3 problem analysis, we need to extract the 3 tires, 4tries, and 5tries of each word as the criteria for evaluating the word difficulty and perform K-means clustering with category four based on these criteria. Figure 17 shows the clustering results, where the four clustering centers are represented by large and unique shapes, with small dots representing each word and its *x, y, and z* coordinates representing the percentages of 3 tires, 4tries, and 5tries, respectively, and the colors representing the different categories. Due to the inherent depth ambiguity of the 3-dimensional graph, the clustering results could be more intuitive. Therefore, the clustering results are shown in Figure 18 for the three percentage fields between the two, which can be interpreted as the projection of the 3-dimensional graph in the *x, y, and z* plane.

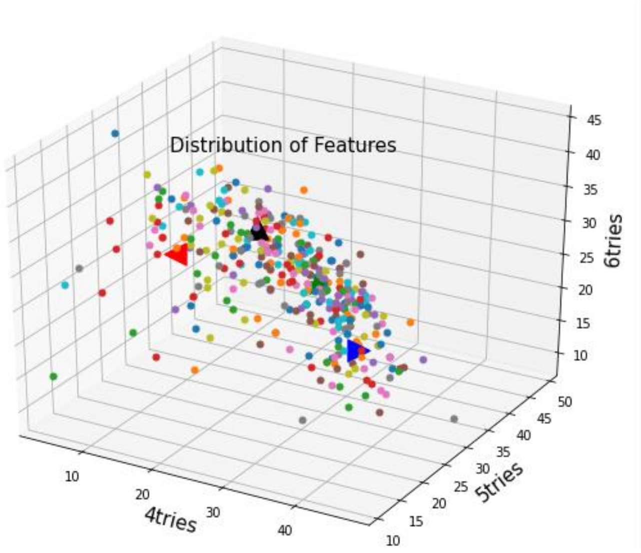


Figure 17: Q3 K-means Model clustering results 3-dimensional graph

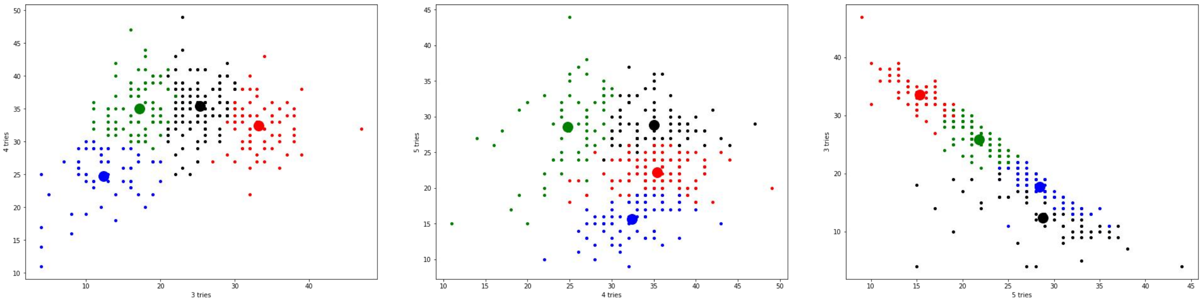


Figure 18: Q3 K-means Model clustering results 2-dimensional graph

The coordinates of the clustering centers, the distance from the origin, and the degree of difficulty are shown in Table 6. degree of difficulty: A-difficult, B-difficult, C-average, D-simple.

Table 6: Clustering Center Information

	x	y	z	Euclidean distance from the origin	Difficulty level of the class
Center 1	17.309	34.907	28.762	48.429	C
Center 2	33.342	32.328	15.589	48.988	A
Center 3	12.406	24.745	28.576	39.785	D
Center 4	25.492	35.5	22.092	48.971	B

From the results of the second question, the percentages of 3 tires, 4 tries and 5 tries for "EERIE" are [27,33,32]. The result of this data into the Q3 K-means Model is that "EERIE" belongs to category "B", which means "more difficult".

6.3.2 Q3 DT RF Model

6.3.2.1 Model instantiation

To study the properties of words, we choose 11 common lexical properties (DT, JJ, Etc.) as features and encode words and lexical properties as one-hot. In addition, we can derive the difficulty level of each word from the Q3 K-means Model. Next, the problem is understood as a discrete data classification problem. The decision tree nodes are selected for classification based on the Gini index, and the random forest contains ten decision trees [12,13]. The correspondence between the word difficulty and difficulty can be found by training the decision tree and the random forest classifier (Q3 DT RF Model).

The lexical one-hot code for "EERIE" is [0,0,0,0,0,1,0,0,0,0,0,0,0]. The result of the Q3 DT RF Model is "B", which means "harder".

6.3.3 Analysis of results

For both methods, the final result is "B," i.e., "harder" for all three classifiers. This illustrates both the accuracy of our prediction in question 2 and the success of our prediction in question 3. This shows that our model predicts difficulty based on lexical properties more accurately.

7 Task4: Digging for other interesting information

We have a wild guess about the game: The number of reported results is most likely related to weekdays or weekends because people tend to have more time to focus on real life rather than electronics on weekends.

We analyzed the mean values. It was found that overall, i.e., all times, number of reported results did not correlate with weekends (Figure 19). Furthermore, 150 days after the count (when it has stabilized), there is a large negative correlation between the day of the week and the score, and the score on weekends (Saturday and Sunday) is significantly smaller than the score on weekdays (Figure 20). One possible reason for this is that after a period of time, people's enthusiasm gradually recedes, and they prefer to use Wordle to pass the time on weekdays.

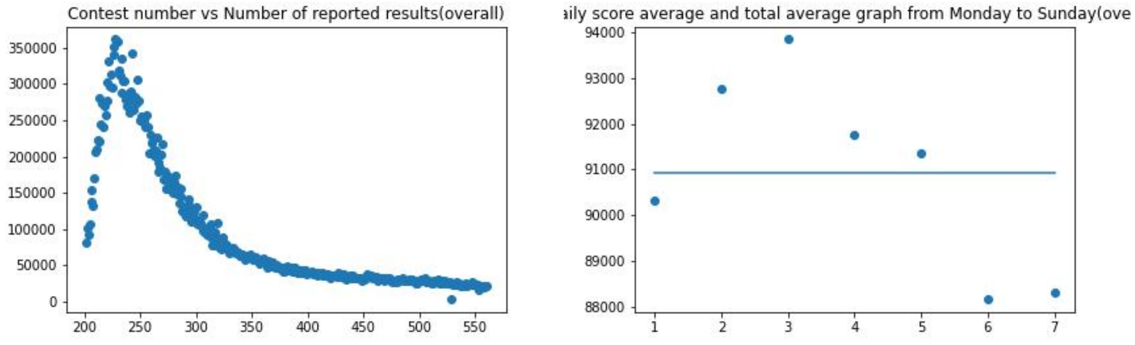


Figure 19: Score-Number distribution (overall) and mean and total mean of scores for each day from Monday to Sunday (overall)

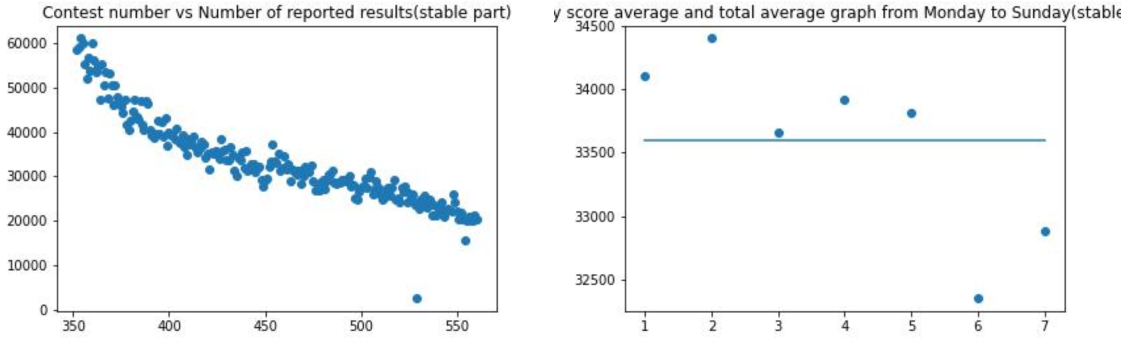


Figure 20: Score-number distribution (stable part) and mean and total mean of daily scores from Monday to Sunday (stable part)

8 Error Analysis

Our solution does not consider the rules of the game and considers it a crossword puzzle. Problem 2 considers only the possibility of human selection of letters and using this probability as a weight is not reasonable. There are correlations between letters in Wordle games, which can be experienced from the square color rule. The quantification process of words does not reflect the rules of the game and is bound to make the final prediction results have a significant error. In addition, the selection of word attributes in question three should not be restricted to word properties only. More attention should be paid to specific features of the data set, such as word frequency, word frequency, Etc. Using 11-dimensional data to describe word properties as an attribute would exaggerate the effect of word properties on the difficulty level. In order to avoid excessive errors, instead of one-hot coding, we should calculate the weighted values and add other attributes as input to get a more comprehensive view of the influence of attributes on difficulty.

9 Model Evaluation and Further Discussion

Our model is simple in structure and easy to implement, but accordingly, many things could be improved with its use. We will discuss and analyze the possible risks encountered next.

9.1 Strengths

Task 1: Q1 Segmentation Function Y Model has deterministic mathematical expressions that make it scalable and perfect for future problems such as derivatives and integrals. Q1 LSTM Model is extremely good at handling time series. The introduction of the gate mechanism allows them to correlate well with recent and early series data.

Task 2: Genetic algorithms are commonly used to optimize BP neural networks for the selection of initial weight thresholds. These speeds up the fitting speed of the BP network and thus facilitates the generation of a more generalized model. Words are quantifiable based on statistical principles, with the frequency of letter occurrences as the weights. It is consistent with human behavior toward word guessing and is a better and more innovative quantization scheme for words with fixed length.

Task 3: A comparison experiment with two schemes can verify the clustering effect of the Q3 K-means Model on the one hand and the prediction accuracy of the Q2 GA-BP Model on the other hand.

9.2 Weaknesses

-No cleaning was performed on the given data. It is noted that the sum of the correlation percentages has exceeded 100% in some cases, but in practice, this data is not excluded considering the sparse data volume. It is also not normalized in use, which directly affects the performance of the trained model.

-Single evaluation metric for models. Machine learning models have multiple evaluation metrics not limited to mean relative error. The average relative error is not universal. It is biased to use it for some specific models. It is not easy to describe the goodness of the model.

-**The models chosen are too classical solidified**. Most of the models cited in this paper are classical models. Some models are too simple, and some are too complicated. Simple models often do not get more expected results, and complex models take more time.

9.3 Further Discussion

Before data processing, make sure to pre-process data, such as eliminating invalid values, extending valid data, Etc. For the trained model, multiple evaluation metrics should be adopted to evaluate the model in an overall style and multivariate manner. We should pay attention to the more popular models in recent years to select model categories. Most of them are based on traditional models and optimized on this basis. They tend to have better performance in dealing with the same problem. In short, one should take a more comprehensive view of the problem to achieve twice the result with half the effort.

10 Memorandum

To: New York Times Editorial Board
From: MCM Team # 2309601
Subject：Trend analysis on the digital characteristics of wordle reports
Date：February 20, 2023

Dear Sir:

We are a team participating in the 2023 US Student Mathematical Modelling Competition. In response to your question, we have investigated it using sound In response to your question, we have investigated it using sound mathematical formulations and machine learning models in conjunction with the available wordle feedback data, and with the help of rational analysis, The following is a summary of our results on this task.

We have adopted two solutions to address the problem of the forecast interval for the results reported on 1 March 2023. First, to better identify the patterns of change in the data, we visualized the data in the table with the help of a tool. First, to better identify the patterns of change in the data, we visualized the data in the table with the help of a tool. We found that the data showed a general trend of increasing and then decreasing, so we found the cut-off point between the two patterns of change as the threshold date, based on which we created a segmentation function and In addition, we also received better results with the help of the time-series model, and the relative error of the final time-series model. In addition, we also received better results with the help of the time-series model, and the relative error of the final time-series model (Q1LSTM Model) was only 5%, so we used [1313,1454] as the last prediction interval.

We first considered the quantification of words for predicting the Percent of try times for a particular day and word in the future. Considering that the words are of the same length, we break down individual words into separate letters and use the frequency of occurrence of the letters in common vocabulary. words are of the same length, we break down individual words into separate letters and use the frequency of occurrence of the letters in common vocabulary Based on this, a BP neural network was used to establish the connection between the input as date and word and the Based on this, a BP neural network was used to establish the connection between the input as date and word and the output as seven percentages, and the model was optimized with the help of a genetic algorithm, and the relative error of the model (Q2GA-BP Model) was The correlation percentages for "EERIE" on March 1, 2023, were 0%, 4%, 27%, 33%, 22%, 7%, and 3%. Based on this result, there is roughly a 63% probability that our prediction is accurate.

In the third question, we prioritized evaluation indicators that measure the difficulty of words. We consider that the difficulty of successfully guessing a word is related to the number of guesses, the higher the number of guesses, the greater the difficulty. For this reason, we used three tries, four tries, and five tries from Problem 2 as the evaluation scale, taking into account the error. Next, the words were classified using the K-means clustering Next, the words were classified using the K-means clustering method, and a Q3K-means

Model was built to rate the difficulty of the words based on the classification with the help of Euclidean distances (the difficulty levels were divided into A-difficult, B-difficult, C-average, and D-simple). Finally, the difficulty of the word "EERIE" was determined to be B-difficult using the Q3K-means Model. In addition, both the decision tree and the random In addition, both the decision tree and the random forest classifier (Q3DT Q3RF Model) were used to predict the word "EERIE" with a definite lexical identity, and the results of both were: B-difficult. results of the Q3K-means Model are consistent with the prediction results of the Q3RF Model, further validating the accuracy of the findings.

Concerning the characterization of the given dataset, we guessed, based on a priori knowledge, that the number of reported results for that day should be correlated with both weekdays and non-weekdays. However, the mean analysis showed that the number of reported results was not directly correlated with whether it was a weekday, but further observation of the score trends over the following 150 days revealed that a day of the week was negatively correlated The above analysis suggests a significant difference in the allocation of play time between weekdays and non-workdays, with weekdays scoring significantly higher than non-workdays. The above analysis suggests a significant difference in the allocation of play time between weekdays and non-weekdays, with more people using wordle for leisure and relaxation on weekdays. The reason for this difference may be that people's perception of wordle has changed over time from a hot trivia game to a gadget to spend time between work.

The above is a summary of our solutions and results to the issues raised by your society. Thank you for taking the time to review them, and please feel free to Thank you for taking the time to review them, and please feel free to contact us if you need more information about our findings.

Sincerely,

Team # 2309601

References

[1] "wordle-The New York Times." The New York Times, 2022. Accessed December 13, 2022 at http://www.nytimes.com/games/ wordle/index.html.

[2] Chuang-Chun Liu. Understanding player behavior in online games: The role of gender [J]. Technological Forecasting & Social Change, 2016:265-274.

[3] Jeon JiHoon, et al. Extracting gamers' cognitive psychological features and improving performance of churn prediction from mobile games[C]// Computational Intelligence & Games. IEEE, 2017.

[4] Kim Seungwook, et al. Churn prediction of mobile and online casual games using play log data [J]. PloS one, 2017, 12(7):e0180735.

[5] Periáñez África, et al. Churn Prediction in Mobile Social Games: Towards a Complete Assessment Using Survival Ensembles[C]//2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA). IEEE, 2016.

[6] Vafeiadis T, Diamantaras K I , Sarigiannidis G , et al. A comparison of machine learning techniques for customer churn prediction [J]. Simulation Modelling Practice & Theory, 2015, 55:1-9.

[7] Zha W, Liu Y, Wan Y, et al. Forecasting monthly gas field production based on the CNN-LSTM model [J]. Energy, 2022: 124889.

[8] Pande H. Mathematical modeling of the frequencies of letters for their occurrence in corpora, words (types) and in the initial positions of words of corpora [J]. Glottotheory, 2021, 12(1): 57-69.

[9] Han J X, Ma M Y, Wang K. Product modeling design based on genetic algorithm and BP neural network [J]. Neural Computing and Applications, 2021, 33: 4111-4117.

[10] Katoch S, Chauhan S S, Kumar V. A review on genetic algorithm: past, present, and future [J]. Multimedia Tools and Applications, 2021, 80: 8091-8126.

[11] Ghazal T M, Hussain M Z, Said R A, et al. Performances of K-Means Clustering Algorithm with Different Distance Metrics [J]. Intelligent Automation & Soft Computing, 2021, 30(2).

[12] Yariyan P, Janizadeh S, Van Phong T, et al. Improvement of best first decision trees using bagging and dagging ensembles for flood probability mapping[J]. Water Resources Management, 2020, 34: 3037-3053.

[13] Speiser J L, Miller M E, Tooze J, et al. A comparison of random forest variable selection methods for classification prediction modeling [J]. Expert systems with applications, 2019, 134: 93-101.