Word Mining Research Based on Intelligent Algorithms

Ruilin Chu *

College of Automation, Southeast University, Nanjing, China, 211189

* Corresponding Author Email: 15689372668@163.com

Abstract. Wordle is a popular puzzle that The New York Times currently provides every day, and it has a high popularity. Among them, the number of results reported every day, the characteristics of words and other data have attracted widespread attention. This paper first used the ARIMA model to predict the number of daily reported outcomes and found that it was only accurate for the linear part of the data. Then, this paper used the LSTM neural network model to predict, and found that the LSTM model can predict the nonlinear part of the data well, which just makes up for the deficiency of the ARIMA model, and the predicted results are basically consistent with the original data. The data range of March 1st is [17586.36, 44379.83]. Further, this paper adopted the LSTM neural network model based on genetic algorithm optimization, which can solve the over-fitting problem that may occur in the LSTM neural network due to too few data sets. Finally, the SVM multi-classification model are used. According to the quantified word feature labels, the difficulty of words is divided into three categories: hard, medium, and easy. Using existing data tests, it's proved that the classification accuracy is very high.

Keywords: Time Series, LSTM, Tag Sort, Genetic Algorithm.

1. Introduction

1.1. Background

Wordle is a popular jigsaw puzzle game published daily by The New York Times. Players need to guess the English word composed of 5 random letters within 6 chances. After each guess, the game shows which of the letters the player guessed were correct, which were incorrect, and which correct letters were in the correct positions. The game has a normal mode and a hard mode. The hard mode requires that once the player finds the correct letters in the word (the stickers are yellow or green), they must use these letters in subsequent guesses.

The difficulty of each word is not the same, it depends on the characteristics of various aspects of the word, such as letters, pronunciation, frequency of use, etc. And these characteristics will significantly affect the player's guessing of words.

This paper has made statistics on the game result reports of many players from January 7, 2022 to December 31, 2022. The statistical files include the total number of daily game passers, the number of difficult mode passers, and the percentage of players who try. Statistics found that the results of the game are changing every day, which may be related to some properties of words. Therefore, predicting future word-guessing results based on existing data has wide application value for tasks such as reasoning, selection and sorting of words.

1.2. Research Status of the Problem

According to the research of Nadav Oved, the BiLSTM(Hochreiter and Schmidhuber)model are commonly used tool in time-series analysis. Both models make a prediction for the next time step (game/period) given performance metrics from the three previous time steps [1]. These models exhibit the predictive power of performance metrics alone and serve as baselines for comparison to text-based models. This model can incorporate textual features from participators' data with performance metrics from the previous three-time steps. These models help us quantify the marginal effect of adding textual features in predicting the direction of the deviation from the player's mean performance, over metric-based models.[2]

In natural language processing (NLP), several methods have achieved promising results in this task. Deep learning-based approaches have been widely used in this context, and deep neural networks (DNNs) have increased the ability to generate representations for data and learning models. The expected increase in size and complexity of DNN architectures introduces new challenges to designing and configuring models.[3]

To classify words, we can mark words through experiments involving tasks such as morphological syntax, and then make four marker generators based on three different markup syntaxes (hidden Markov model, memory-based, transformation rules, and maximum entropy) [4]. Simply by using more than one classifier, and voting between their outputs, we expect to eliminate the quirks, and hence errors, that are due to the bias of one particular learner. However, there is also a way to make better use of the differences: we can create an arbiter effect.

2. Basic Model

2.1. Time Series Forecasting Model

Time series prediction is based on historical data and uses scientific methods to predict the number of reported results in the future. Knowing the time series {X1, X2…XT}, find the number of reported result XT+1, XT+2…XT+m.The formula is shown in equation 1.

$$\hat{X}_{T+1}, \hat{X}_{T+2}, \hat{X}_{T+3}, \cdots, \hat{X}_{T+m} = f(X_1, X_2, X_3, \cdots, X_T) \tag{1}$$

Time series forecasting methods are divided into linear forecasting models and nonlinear forecasting models. Daily game situations contain many uncertain factors, which will be affected by various social factors, and these factors are also highly disordered, so it is difficult to predict with a single model. Traditional time series models can extract linear features, while neural network models have strong mapping properties for nonlinearities. Therefore, this paper use the linear time series forecasting algorithm ARIMA and the nonlinear forecasting LSTM neural network to predict the number of people who guessed the word correctly, and combine the results to obtain the prediction interval.

2.1.1 ARIMA Linear Prediction Model

ARIMA (Autoregressive Moving Average Model) is a statistical model widely used in time series analysis and forecasting. The model combines autoregressive (AR) and moving average (MA) to transform a non-stationary time series into a stationary time series. The core of the ARIMA model is the autocorrelation of time series. It assumes that observations in a time series are related to previous observations and uses these relationships to predict future observations. Let $\{X_t, t = 0, \pm1, \pm2, \cdots\}$ be an ARIMA (p, d, q) sequence.[5]

$$X_t - \varphi_1 X_{t-1} - \cdots - \varphi_p X_{t-p} = \varepsilon_t - \theta_1 \varepsilon_{t-1} - \cdots - \theta_q \varepsilon_{t-q} \tag{2}$$

Then its autocovariance function satisfies

$$\varphi(B)\gamma_k = \begin{cases} 0, & k > q \\ -\theta_q \sigma_\varepsilon^2, & k = q \end{cases} \tag{3}$$

From formula (3), taking $k = q+1, q+2, \cdots, q+p,$, we get the system of equations

$$\begin{cases} \gamma_{q+1} = \varphi_1 \gamma_q + \varphi_2 \gamma_{q-1} + \cdots + \varphi_p \gamma_{q+1-p} \\ \gamma_{q+2} = \varphi_1 \gamma_{q+1} + \varphi_2 \gamma_q + \cdots + \varphi_p \gamma_{q+2-p} \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \gamma_{q+p} = \varphi_1 \gamma_{q+p-1} + \varphi_2 \gamma_{q+p-2} + \cdots + \varphi_p \gamma_q \end{cases} \tag{4}$$

Write it in matrix form as

$$\begin{bmatrix}\gamma_{q+1}\\ \gamma_{q+2}\\ \vdots\\ \gamma_{q+p}\end{bmatrix}=\begin{bmatrix}\gamma_q & \gamma_{q-1} & \cdots & \gamma_{q+1-p}\\ \gamma_{q+1} & \gamma_q & \cdots & \gamma_{q+2-p}\\ \vdots & \vdots & & \vdots\\ \gamma_{q+p-1} & \gamma_{q+p-2} & \cdots & \gamma_q\end{bmatrix}\begin{bmatrix}\varphi_1\\ \varphi_2\\ \vdots\\ \varphi_p\end{bmatrix} \tag{5}$$

Solving this system of equations can be obtained from the regression coefficient vector $(\varphi_1, \varphi_2, \cdots, \varphi_p)^T$. Where ε_t is zero mean; σ_ε^2 is smooth white noise; p is autoregressive order; φi is autoregressive parameters;X_t is a stationary sequence after the d-order difference;q is the moving average order.

2.1.2 LSTM Neural Network Model

LSTM (Long Short-Term Memory) is a recurrent neural network (Recurrent Neural Network, RNN) model commonly used to process sequence data. The core of the LSTM model is a memory unit and three gates, which realize alignment by controlling information flow. Sequence processing and memory.[6]

The algorithm flow of LSTM can be divided into three steps: the forget gate is used to control whether the information is retained or not, and the decision on how much information to forget depends on the new input Xt and the ht-1 passed by the previous layer, which is determined by the α (Sigmoid function) function output. The formula is shown in (6) below.

$$f_t = \sigma(W_f \times [h_{t-1}, X_t + b_f]) \tag{6}$$

Where f_t is the information retained after the information input by the upper layer network passes through the forget gate; is its weight information.

The input gate controls the input of new information. First, the Sigmoid activation function is used to judge the choice of the input information value of the previous layer, and then the new activation function Tanh is used to form a new output and multiplied by the calculation result of the Sigmoid function. The resulting formula is as follows.

$$i_t = \sigma(W_i \times [h_{t-1}, X_t] + b_i) \tag{7}$$

$$\tilde{C}_t = \tanh(W_c \times [h_{t-1}, X_t] + b_c) \tag{8}$$

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \tag{9}$$

i_t is the information retained by the sigmoid function;$\tilde{C}_t$ is the added input information;C_{t-1} is the old unit cell information;C_t is the new cell state information. To output the value that needs to be output and the state of the unit, you first need to run the Sigmoid function to determine the specific part of the output unit state. The cell state is manipulated by the tanh function multiplying it by the sigmoid output. The formula for the output gate is as follows.

$$o_t = \sigma(W_o \times [h_{t-1}, X_t] + b_o) \tag{10}$$

$$h_t = o_t \times \tanh(C_t) \tag{11}$$

o_t is the output information about the cell state;h_t is the part of the information you want to output.

2.2. SVM Classification Model

The support vector machine (SVM) is a statistically based mathematical model that can be used in classification and regression problems. Like most machine learning classification models, support vector machines are built through training. For linearly inseparable problems, SVM can map the training data, that is, the label value of each word from the linearly inseparable feature space to a higher-dimensional space, so that linearly inseparable data can be classified through such a hyperplane.

The goal of the SVM training process this paper use is to find an optimal hyperplane, and the classification results generated by this hyperplane need to be the most robust and show super

generalization ability to new data. In the sample space, the partition hyperplane can be expressed by formula (13):

$$f(x) = W^T \emptyset(x) + b \tag{12}$$

Where w and b are model parameters, w indicates the weight and b represents a constant; $\emptyset(x)$ represents the feature vector after mapping x and maps the input data set to a high-dimensional space.[7]

In order to verify the accuracy of the SVM classification model, we need to manually classify the existing data sets. Here we use the weighted average method based on the report results, as follows:

$$Y = \sum_{i=1}^{7} i \cdot G_i \tag{13}$$

Gi in the formula represents the percentage value of the word i-th try.

3. Optimize: Multivariate Time Series Forecasting Model

3.1. LSTM Multivariate Time Series Forecasting

When building a univariate forecasting model, only time and the variable itself can be used for reference, and such variables often appear very thin. The multivariate time series means that in addition to time dependence, other factors jointly affect the occurrence of a certain result. The multivariate time series can understand and use the relationship between multiple variables, which helps to describe the dynamic behavior of the data and provide better forecasting results. The LSTM-based recurrent neural network can almost perfectly solve the problem of multiple input variables.[8]

The title requires predicting the relevant percentage of future date results, which requires time and the value of the word classification label as a variable, and the relevant percentage as an output. Suppose the data looks like this.

$$\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} & y_1 \\ x_{21} & x_{22} & \cdots & x_{2n} & y_2 \\ \vdots & \vdots & & \vdots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} & y_m \end{pmatrix} \tag{14}$$

To predict the value of y_m in period m, we can use all the information in period m-1.

$$(x_{m-1,1}, x_{m-1,2}, \cdots, x_{m-1,n}, y_{m-1}) \tag{15}$$

Establish a linear model between the target value y_m of the period m and all the information of the period $m-1$.

$$y_m = w_0 y_{m-1} + w_1 x_{m-1,1} + w_2 x_{m-1,2} + \cdots + w_n x_{m-1,n} + \epsilon \tag{16}$$

And then extended to the p-order lag, you can get better prediction results.

3.2. Genetic Algorithm Optimization

Genetic algorithm optimization is a computational method based on biological evolution theory for solving optimization problems. It is an evolutionary computing method that simulates the evolutionary process in nature, generates new solutions through operations such as selection, crossover, and mutation, and gradually optimizes the solutions.[9]

Because the genetic algorithm optimization parameters are the initial weights and thresholds of the LSTM neural network, as long as the network structure is known, the number of weights and thresholds is known. The genetic algorithm is introduced to optimize the best initial weights and thresholds.

The process of bringing the model of the paper to the Genetic Algorithm is as follows:

Population initialization: Individual coding uses binary coding, which is composed of four parts: input layer and hidden layer connection weight, hidden layer threshold, hidden layer and output layer

connection weight, output layer threshold, each weight and threshold using M bit binary coding, the ownership value and threshold code is connected as an individual code. The LSTM network structure this paper use is 15 - 31 - 3, so the number of weights and thresholds is listed in Table 1:

Table 1: Weights and Thresholds

α_1	β	α_2	γ
465	31	93	3

Where $\alpha_1$is the connection weight between the input layer and the hidden layer; β is the implied value threshold; α_2 is the connection weight between the hidden layer and the output layer; γ is the output layer threshold.

Fitness function: Select the norm of the error matrix of the predicted value and the expected value of the prediction sample as the output of the objective function. The fitness function adopts the ordered fitness allocation function:

$$\mathrm{FitnV} = \mathrm{ranking}(\mathrm{obj}) \tag{17}$$

Where obj is the output of the objective function.

Select the operator: Select operators using random traversal sampling (SUS).

Cross operator: The cross operator uses the simplest single-point cross operator.

Mutation operator: Mutation produces the number of mutated genes with a certain probability, and the mutated genes are selected by random method. If the code of the selected gene is 1, it becomes 0; Otherwise, it becomes 1. Table 2 sets the operating parameters.

Table 2: The Operating Parameters.

Population size	Maximum genetic number	Binary digits	Crossover probability	Mutation probability	Generation gap
40	50	10	0.7	0.01	0.95

This paper brings existing word labels and associated percentages into the model for training. At the same time, in order to prevent over-fitting due to the small data set, it is necessary to combine the genetic algorithm to optimize and reduce the error value. Specifically, the error is reduced from 0.84 to 0.78.

4. Data Pre-processing

All data used in this article are from Question C of MCM 2023.

First, the paper pre-processes the data to remove outliers. In the rules of the game, a given word consists of five letters, and therefore words and related data that do not meet the requirements in the statistics are removed. The function in MATLAB is used to remove outliers to remove data whose values are far from the overall trend. The abnormal data is shown in Table 3, and the correct words are indicated in parentheses.

Table 3: The Abnormal Data

Data	Word	Number of reported results
2022/4/29	tash (trash)	106652
2022/11/26	clen (clean)	26381
2022/11/30	study	2569
2022/12/16	rprobe (probe)	22853

5. Analysis of the Results

5.1. Comparative Analysis of the ARIMA Model and the LSTM Model

5.1.1 ARIMA Time Series Model

First, by using the ARIMA time series prediction model, this paper performed a forecast fit to the number of outcomes reported in 2022 and obtained the following results as Figure 1:

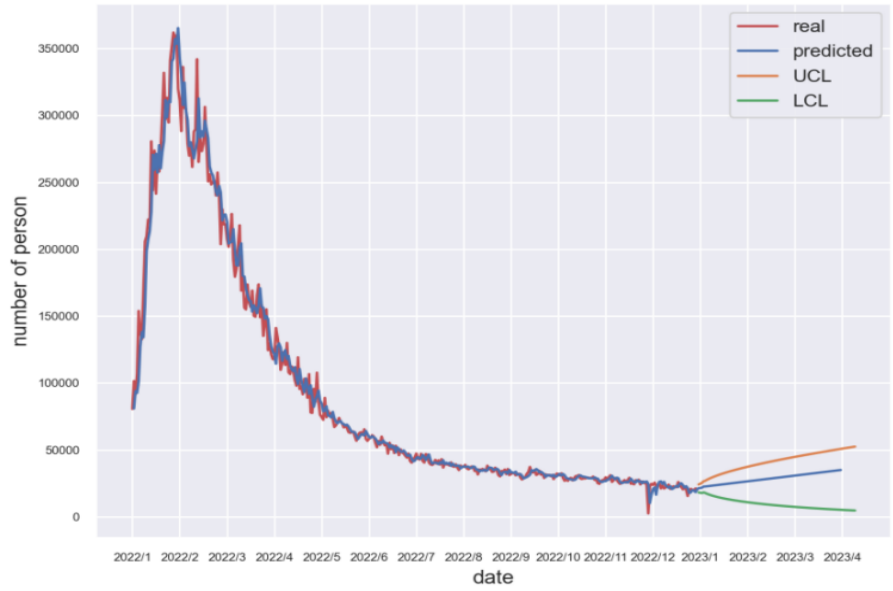

Figure 1: ARIMA Forecast Chart

From Figure 1, it's found that the ARIMA model has a high degree of fitting to the linear part of the sequence, and there is a certain error in the nonlinear part.

5.1.2 LSTM Neural Network Model

Further, this paper use LSTM neural network model to perform a forecast fit to the number of outcomes reported in 2022 and obtained the following results as Figure 2:

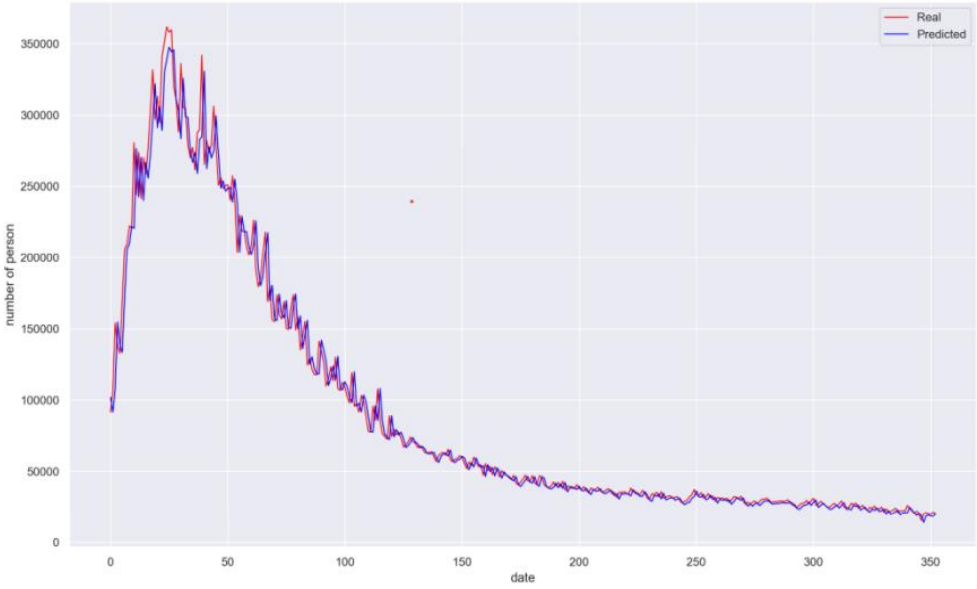

Figure 2: LSTM Forecast Chart

From Figure 2, we can see that the LSTM model can also have a high degree of fit to the nonlinear part of the sequence, which is more accurate than the ARIMA model.

5.1.3 Forecasting Results

In order to make the final forecast value more accurate, this paper use Mean Absolute Percent Error (MAPE) and R2 as the model performance evaluation indexes of the two-time series forecasting methods. The evaluation results are obtained in Table 4.

Table 4: The Evaluation Results

Model	R^2	MAPE
ARIMA	0.983	9.12
LSTM	0.992	6.52

By observing Table 4, it is not difficult to see that the MAPE of the LSTM model is lower and R2 is closer to 1, so the prediction accuracy of the LSTM model is higher than that of the ARIMA model. This is because the LSTM model makes up for the shortcomings of ARIMA for the prediction of the nonlinear part of the data.

Further, this paper predicts that the report result value on March 1, 2023, will be 30539.27, and the value range will be [17586.36, 44379.83].

5.2. Multivariate Time Series Forecasting Model

5.2.1 Word Attempts Prediction Results

After the model is trained on existing data, it predicts the results of EERIE words on March 1, 2023. According to the set label, the input matrix of the word is.

[2, 2.75, 3, 1, 1]

and the percentage of the number of attempts of the word is predicted. After normalization, the prediction results are shown in Table 5.

Table 5: The Prediction Results

	1 try	2 tries	3 tries	4 tries	5 tries	6 tries	X tries
Percentage	0.8791	1.2292	16.4590	36.5376	27.6860	12.4001	4.8091

The prediction result error is shown in Table 6.

Table 6: The Prediction Result Error

	MSE	R2	MAPE
Prediction	1.683	0.991	4.56

From the data in the Table 6, it can be seen that the fitting degree of the model is better and has greater credibility.

5.2.2 Uncertainties in Model and Predictions

The LSTM time series forecasting model optimized by the genetic algorithm this paper adopted has the following uncertainties:

Noise Uncertainty: Noise in the time series is random, the model cannot remove the noise, and there will be noise effects when modeling the percentage of daily attempts. For example, the three-attempt percentage chart is shown in Figure 3.

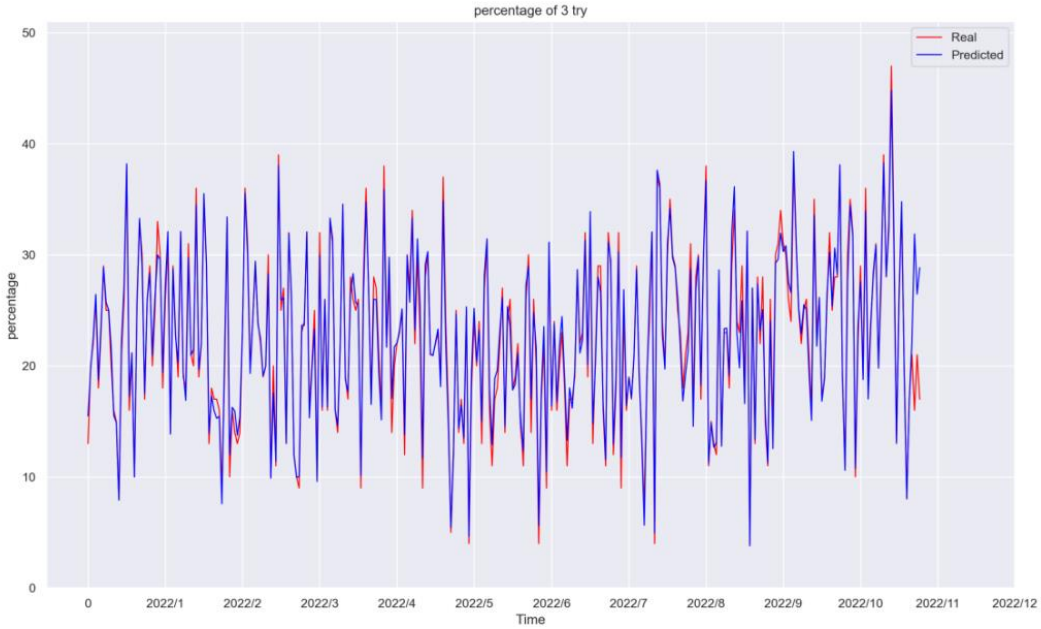

Figure 3: Noise Plot during Data Processing

Parameter uncertainty: In word classification, the selection of label parameters is inferred based on existing data, which is subjective and uncertain. The selection of different parameters will lead to different results, resulting in model uncertainty.

Data uncertainty: There are outliers in historical data, and the standard for eliminating outliers is uncertain. When outliers are deleted, we only consider the data that do not meet the requirements of the topic and the deviation is too large, and keep the data whose sum of the percentage of attempts is not 1, which may lead to different prediction results.

Uncertainty in the future: The time series forecasting model predicts the future based on historical data, and unexpected changes may occur in the future, resulting in differences between the forecast results and the actual values.

5.3. SVM Classification Model

This paper wants to divide the word difficulty into three categories according to the Y in the above formula, and the values 3, 2, and 1 represent difficult, medium, and easy respectively, and the division intervals are [100, 250), [250, 300), [300, 400]. Then this paper uses each label value of the word feature as the input of the mapping function, use the RBF kernel function, use the SVM model, randomly set a part of the existing data as the training set, and set the rest as the test set, and compare the predicted classification with the actual classification. [10]

This paper uses the SVM model to classify the difficulty of the word EERIE, and the results are shown in Table 7:

Table 7: Difficulty Classification Results

Difficulty degree	Difficulty	Accuracy
1	Easy	0.971

That is the difficulty of the word EERIE is easy.

In order to verify the accuracy of the classification, the paper first used the prediction of the word EERIE made by model II and calculated its weighted average according to the calculation method of formula (14). The result was 354.6, which was consistent with the interval of the prediction result. Then, the paper averages the five tag values of each difficulty word as a reference, and draw the radar chart of the EERIE tag value as Figure 4:

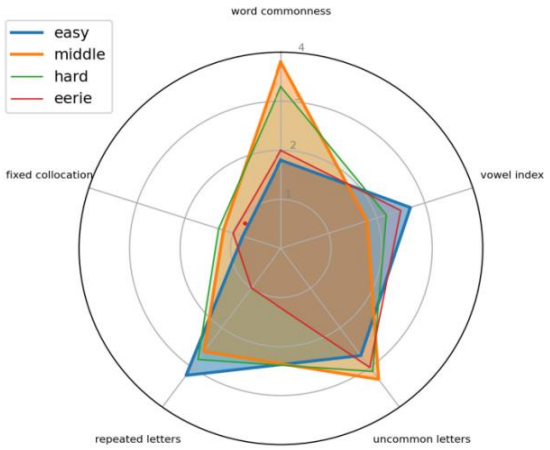

Figure 4: The Radar Chart of the EERIE Tag Value

It can be seen that the eigenvalues of EERIE are closer to simple types of words.

After two aspects of verification, it can be considered with certainty that the model of this paper is accurate for the classification of a given word. At the same time, we can also see the correspondence between each difficulty classification and word features through Figure 8. Among them, the commonness of words and rare letters have the most obvious impact on the difficulty of words, and words with high frequency and few rare letters are more likely to be guessed.

6. Conclusion

This paper analyzed that for the game of Wordle, there is a certain internal relationship and law in the number of people who play it, the number of people who choose the hard mode and the proportion of the number of attempts, and the difficulty of the word has a certain relationship with the commonness of the word, vowel letters, unfamiliar words, repeated words, and fixed collocations. Through the prediction results of the time series and the neural network, we can conclude that the number of players on March 1, 2023, is 30539, and the number of people range is [17586.36, 44379.83]. Further, we used a modified time series model with genetic algorithms to predict that the proportion of attempts for the word EERIR on March 1, 2023, was [0.8791, 1.2292, 16.4590, 36.5376, 27.6860, 12.4001, 4.8091]. Finally, after verification, the classifier we constructed is more accurate, and the difficulty of the word EERIE is determined to be easy. In addition, we also found some interesting relationships between the data, such as a higher percentage of people who chose difficulty when the number of reported results was small; When the proportion of people who chose difficulty was higher, the word showed a lower overall difficulty; There was little correlation between the number of results and the observed word difficulty.

References

[1] Liu Chengliang. Research on Air Quality Index Evolution Prediction Model Combining GCN and LSTM [D]. Nanjing: Nanjing University of Posts and Telecommunications. 2022.

[2] Longfuhai. Study on the feature selection method based on the optimization of genetic algorithms [D]. Guiyang: Guizhou National University, 2022.

[3] Okkalioglu Murat. Imbalance text classification with relative imbalance ratio [J]. Expert Systems with Applications,2023, Volume 217, Issue.

[4] Luo Mao. Research on Support Vector Machine Optimization Algorithm Based on Improved Multiverse Algorithm [D]. Changchun: Jilin University, 2022.

[5] Hans van Halteren. Improving Accuracy in Word Class Tagging through the Combination of Machine Learning Systems [J]. Computational Linguistics ,2001, 27 (2): 199–229.

[6] Nadav Oved. Predicting in-game actions from interviews of NBA players [J]. Computational Linguistics, 2020, 46 (3): 667–712.

[7] Xinru Chen, Ruijie Shen, Shuting Sun. Application of BP Neural Network Based on the Genetic Algorithm in Secondary Modeling of Air Quality Forecast [J]. Academic Journal of Environment & Earth Science, 2022, 4.0(3.0).

[8] Qianlong. Financial timing and short-term prediction model research and application based on ARIMA and cyclic neural networks [D]. Chengdu: Xihua University, 2022.

[9] Magalhães Dimmy. Creating deep neural networks for text classification tasks using grammar genetic programming [J]. Applied Soft Computing Journal, 2023, Volume 135, Issue.

[10] Chen Chen. Based on the sheep of the cryptocurrency market, the multi-variable LSTM price prediction research [D]. Nanjing: Nanjing Information Engineering University, 2022.