A research on the Prediction of Wordle Based on Machine Learning

Yitian Yin [1, *], Junzhe Jin [2]

[1] School of Mechanics, Civil Engineering and Architecture, Northwestern Polytechnical University, Xi'an, China, 710129

[2] School of Software, Northwestern Polytechnical University, Xi'an, China, 710129

* Corresponding author: ytyin@mail.nwpu.edu

Abstract. This paper investigated the popularity and difficulty level of Wordle, an online daily puzzle game. The study examined the number of players reporting scores, the number of players on hard mode, and the percentage of players who guessed the word. After removing outliers and misspelling words, the study used time series analysis to predict future numbers of reported results. We found that Wordle had entered the decline period and recommended the last 150 days' smooth data for more accurate prediction interval results. Furthermore, the study developed the Wordle Word n-tries Percentage Prediction Model, which accurately predicts the associated percentages of tries required to solve a given word. The model uses the Regressor Chain algorithm to correlate independent variables such as word frequency, lexical properties, number of common letter combinations, and date with dependent variables. Based on the Decision Tree, the model predicts the associated percentages of tries required to solve a given word.

Keywords: Machine Learning, ARIMA, Prediction Model, Wordle.

1. Introduction

In recent months, an online daily puzzle game called Wordle has taken the Internet. As players around the world guess the same five-letter word, the choice of guessing skill is as important as the difficulty of the question [1-2]. This paper uses the ARIMA time [3] series algorithm to predict the number of players at a future date for Wordle. Likewise, this study will analyze different attributes of a word to assess the difficulty of the word being guessed. In Ivan Li 's research Analyzing difficulty of Wordle using linguistic characteristics to determine average success of Twitter players [4]. It was proved that the linguistic properties of words such as commonality have impacts on player success. However, the models created in the paper also struggle to hold as a result of the dataset and the lack of predictor variables.

This paper takes 6 dependent variables into consideration (frequency of each letter, word frequency, lexical properties, number of common letter combinations, number of repeated letters, date). To make the prediction model more accurate, we use Regressor Chain to process datath-at makes processed data more suitable for predicting by Decision tree, which make more accurate prediction of the difficulty of words.

2. Wordle Future players Prediction Model

2.1. Data Preparation

2.1.1. Data Pre-processing

The given dataset covers 359 days from January 7 to December 31, 2022 and includes Wordle answers. After basic processing in Excel, errors were discovered in some of the given words. Corrections were made to misspelled words, such as "tash" on April 29 and "clen" on November 26, which had only 4 letters and violated the rules. The resulting dataset was complete and ready for analysis.

In data covering 2022, only the player count on November 30 is in the thousands, while others are in the tens of thousands or more. We suspected an error and replaced the value of 2569 with 23184, the average player count from November 29 and December 1. This adjustment eliminated the outlier and made the data more accurate.

2.1.2. Data Collection

We counted 359 corrected words and summarized the frequency of occurrence of letters in each position of five-letter words. Such as the highest frequency of occurrence of s in the first letter and e in the last letter.

Get word frequencies from the COCA corpus. We chose COCA (Corpus of Contemporary American English) as our word frequency statistics database.

2.1.3. Data Visualization

It is easy to obtain a decreasing trend after February 2, 2022, which is consistent with the game life cycle rule. Thus, the next step is to fit the regression model to predict the number of game players from January 1, 2023 to March 1, 2023.

The product life cycle is defined as "the cycle through which every product goes through from introduction to withdrawal or eventual demise". Products sequentially go through the same stages: introduction, growth, maturity and decline. We keep the horizontal coordinate as time and define the vertical coordinate as game impact, and we draw a game life cycle image.

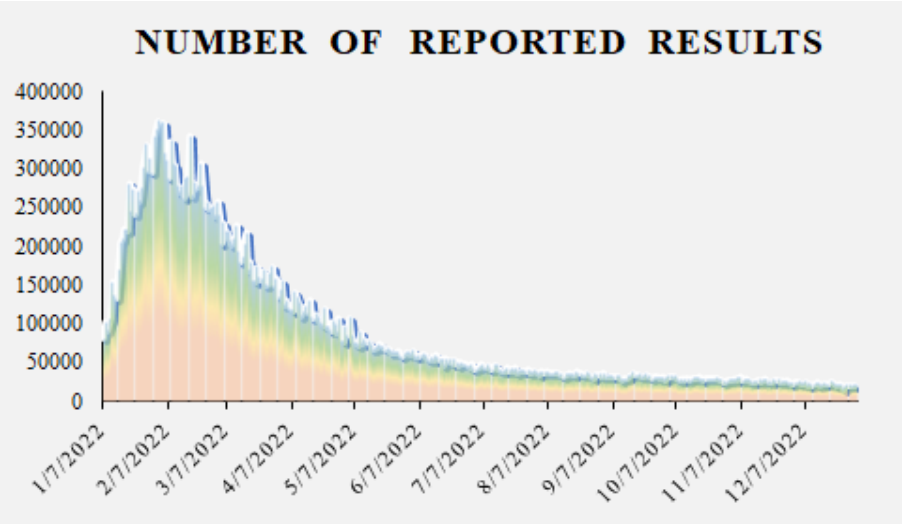

Figure 1. Comparison of Number of Reported Results and Game Lifecycle

Figure 1 shows that the wordle 2022 daily player count is similar with game lifecycle image. It is concluded that the game has entered a period of decline and identified a declining trend.

2.2. Model Preparation

The number of wordle daily players to be predicted is arranged in chronological order to form a time series. It is easy to obtain that there is no sudden change in the number of players per day in wordle and the variance of random change is small within the forecast time range until March 2023. It is reasonable to assume that the past and present evolutionary trends will continue to evolve into the future when the time series ARIMA algorithm is used for forecasting [5-7].

The ARIMA model combines three basic methods:

(1) Autoregressive (AR).

Describe the relationship between the current Wordle player count and the historical values and use the historical time data of the variable to predict itself. The order of the AR model is denoted as $^{\rho}$, and let $\{X_t, t = 0, \pm1, \pm2, \cdots\}$ be a zero-mean smooth series satisfying the following model:

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \cdots + \phi_p X_{t-p} + \varepsilon_t \quad (1)$$

Where: ε_t is the zero mean, variance is the smooth white noise of σ_ε^2; X_t is the autoregressive series of order p=1, abbreviated as AR(p) series; ϕ is the regression parameter vector.

(2) Integrated (I).

When the time series tends to be smooth, it needs to be differenced. The order of differencing is noted as the value. In this problem, the initial time series d does not tend to be smooth, and after the first-order difference, a smoother time series *Xt* is obtained.

Let $\{X_t, t = 0, \pm 1, \pm 2, \cdots\}$ be a non-stationary series and there exists a positive integer d:

$$\nabla d\, Xt = Wt \quad (2)$$

Where the operation ∇^d is a d order backward difference operation.

(3) Moving average (MA).

The moving average model is concerned with the accumulation of error terms in the autoregressive model. the order of the MA model is denoted as q. Let $\{X_t, t = 0, \pm 1, \pm 2, \cdots\}$ be a zero-mean smooth series satisfying the following model.

$$X_t = \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \cdots - \theta_q \varepsilon_{t-q} \quad (3)$$

Where: ε_t is a smooth white noise with zero mean and variance σ^2 is? X_t is a moving average series of order q, abbreviated as MA(q) series; θ is the moving average parameter vector.

The model is called ARIMA (p, d, q), where p is the autoregressive term, q is the moving average term, and d is the number of differences made by smoothing the time series. Combined with the game life cycle function, we programmed our model in R language to estimate the number of players using the 359-day wordle history time series and the 150-day player time series after the game life cycle enters the "decline" period, respectively.

2.3. Model Cacluation

(1): Stabilization

The time series must be stationary. We use the augmented Dickey-Fuller unit root test to test for smoothness. Typically, if the ADF test yields a p-value less than 0.05, the time series is stable. If it is unstable, we us the difference method to transform the non-stationary process into a stationary process. As can be seen from Table 1, the 359-day time series tends to be stable after the first-order difference, while the 150-day time series itself is more stable.

Table 1. *P* value of 359 days' and 150 days' time series before and after difference

	P value before difference	P value after difference
359 days	0.6301	0.01
150 days	0.01	0.01

(2): Selection of p, q and d.

The autocorrelation function (ACF) and the partial autocorrelation function (PACF) were used to determine the values of p and q. The optimal model output was obtained as ARIMA (1,1,0) when the 359-day historical time series was forecasted. The ARIMA (2,1,2) model was obtained when forecasting with the last 150 days of historical data. The confirmation method is shown in the table 2.

Table 2. Confirmation method of p value and q value

Model	AR(p)	MA(q)	ARMA(p,q)
ACF	attenuation tends to 0	truncation after q-order	attenuation tends to 0 after q-order
PACF	truncation after p-order	attenuation tends to 0	attenuation tends to 0 after p-order

(3): White noise test

The residuals were judged to be normally distributed by drawing Q-Q plots with their additive line fit, and then the residuals were further judged to be correlated with each other by white noise test of the fitted model. From the Figure 2, we can see that the residuals follow normal distribution and the p-value is greater than 0.05, which proves that the residuals are not correlated with each other. We proceed to the next step of model prediction. After the game life cycle image estimation, the white noise of the latter 150 days' time series is significantly smaller than the white noise of the original complete time series.

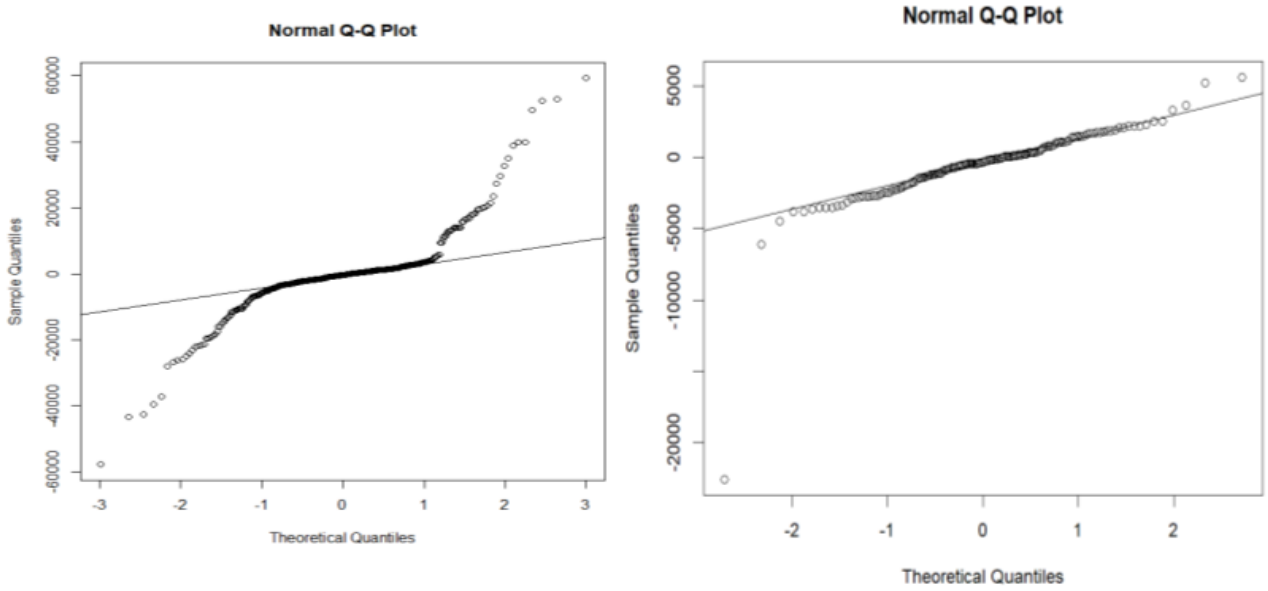

Figure 2. ARIMA (1,1,0) and ARIMA (2,1,2) Normal Q-Q Plot

The p-values of the white noise test outputs of both models are very close to 0, which indicates it obeys a normal distribution with mean 0, i.e., white noise.

(4) Prediction

We plotted the ARIMA (2,1,2) and ARIMA (1,1,0) prediction images for comparison (Fig 3 shows the prediction), where the purple and gray shaded parts are the result fluctuation intervals. Since the game life cycle images were combined, it can be concluded that the 150-days estimated results have a smaller fluctuation interval. The number of reported results on March 1, 2023 is more consistent with the real value.

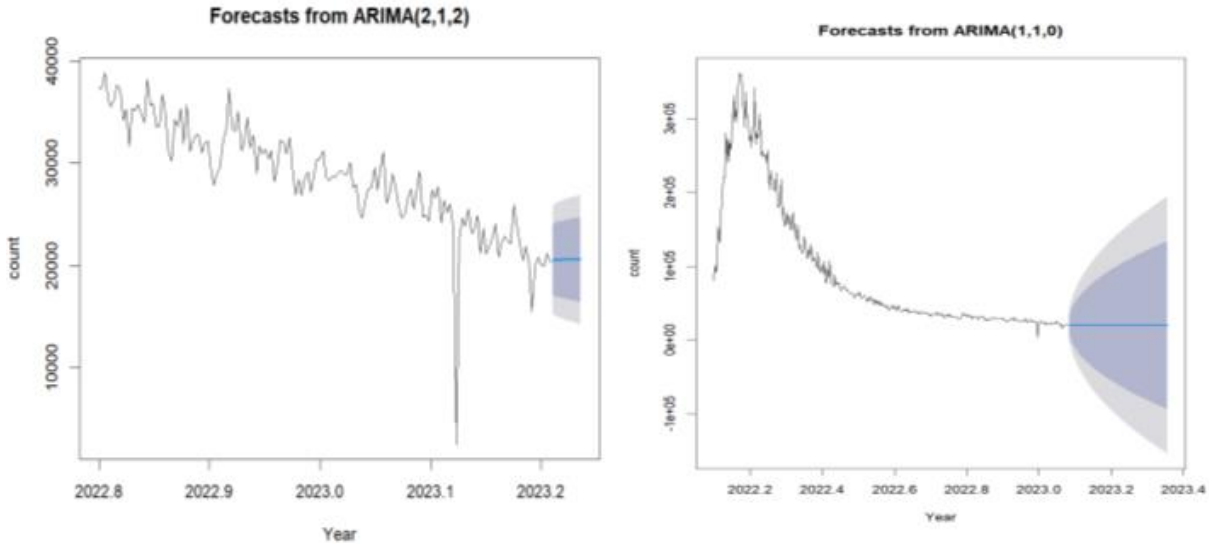

Figure 3. Forecasts results from ARIMA (2,1,2) and ARIMA (1,1,0)

3. Word n-tries percentage Prediction Model

3.1. Preparation

3.1.1. Model

(1) Regressor Chain

Regressor chain is a multi-output model that arranges regressors into a chain. This will create one model per output. The prediction of the first output will be used as a feature in the second output. The prediction for the second output will be used as a feature for the third, etc [8]. This "chain model" is therefore capable of capturing dependencies between outputs.

The large data prediction model for the user's electricity consumption is implemented in the Clementine software.

(2) Decision Tree

Decision tree is a hierarchical model used in decision support that depicts decisions and their potential outcomes, incorporating chance events, resource expenses, and utility. This algorithmic model utilizes conditional control statements and is non-parametric, supervised learning, useful for both classification and regression tasks [9]. The tree structure is comprised of a root node, branches, internal nodes, and leaf nodes, forming a hierarchical, tree-like structure.

It is a tool that has applications spanning several different areas. Decision trees can be used for classification as well as regression problems. The name itself suggests that it uses a flowchart like a tree structure to show the predictions that result from a series of feature-based splits. It starts with a root node and ends with a decision made by leaves. Figure 4 shows the structure of a decision tree.

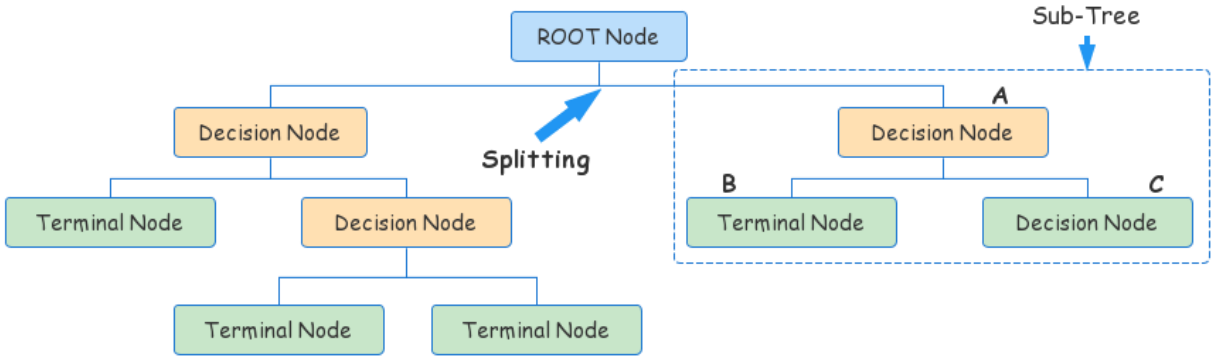

Figure 4. Decision Tree

3.1.2. Accuracy Analysis Method

We will evaluate the predicted value $\hat{Y}$ and the true value Y. We use the following metrics to show the difference between the predicted and true values, and the smaller the error, the better the performance of the model evaluation.

$$\hat{Y} = \{\hat{y}_1, \hat{y}_2, \cdots, \hat{y}_n\}$$
$$Y = \{y_1, y_2, \cdots, y_n\} \quad (4)$$

(1) Mean Square Error (MSE)

Mean Square Error (MSE), a measure that reflects the degree of difference between the estimated quantity and the estimated quantity. [10-11] Let t be an estimate of the overall parameter θ determined from the subsample, the mathematical expectation of $(\theta - t)^2$ is called the mean square error of the estimate t. It is equal to $\sigma^2 + b^2$, where σ^2 and b are the variance and bias of t, respectively.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 \quad (5)$$

Interpretation: the smaller the value, the more accurate the machine learning network model is, and the opposite, the worse.

(2) Root Mean Square Error (RMSE)

Root Mean Square Error (RMSE), from the name, we can all guess what it means. With an extra root, the "root" means just add a root sign. RMSE is the square root of the average squared difference between predicted and true values [12]. The number of observations *n* is always limited, and the true value can only be replaced by the most trustworthy (best) value.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{6}$$

Explanation: when the difference between our predicted value and the true value is smaller, the higher the accuracy of the model.

(3) Coefficient of determination R^2

The extent to which the model can be interpreted as a change in the dependent variable due to the independent variable.

$$R^2 = \frac{SSR}{SST} = \frac{\sum(\hat{y}_i - \bar{y})^2}{\sum(y_i - \bar{y})^2} \tag{7}$$

Explanation: The coefficient of determination is an indicator to assess how good the regression model is [13]. R^2 also takes a range of 0 to 1 and is usually expressed as a percentage. For example, if the R^2 of the regression model is equal to 0.7, then it means that this regression model can explain 70% of the prediction results.

3.2. Solution Prdcedure

Our aim is to build a decision tree model based on the 359 sets of historical data available. The corresponding percentages to predict the percentage of new words afterwards.

The question involves independent variables and one or more dependent variables. However, Direct machine learning cannot predict all dependent variable values simultaneously. Correlation between dependent variables requires consideration of predicted values in series. we need to consider the predicted values of other dependent variables in the series to predict the next target value. We need to consider the predicted values of other dependent variables in the series to predict the next target value.

Regressor Chain is applied to correlate a set of independent variables (percentage of different tries) and dependent variables (frequency of each letter, word frequency, lexical properties, number of common letter combinations, number of repeated letters, date) in a certain order. [14-16]

This study investigated factors affecting difficulty levels in Wordle, including text reading level, English word formation, and time impact on player success rates. We use three key elements to determine feature values: letter probabilities, word frequency, and the date the word was given. These features provided insights into underlying patterns and informed predictive model development to enhance gameplay.

After determining feature values, we preprocess known data for the decision tree. The "word" data is divided into letters that can't be recognized in Python modeling. To numericalize this column, we count the frequency of each letter in the position.

The frequency of the 5-letters is used as the five features of the word. The frequency and time of the word are used as the feature variables. The distribution ratio of the number of successes (1, 2, 3, 4, 5, 6, X) in the table is used as the target variable to build the learning prediction model.

We Divide the training set and test set. The existing data is divided into a training set and a test set in the ratio of 7:3.

Decision tree learning involves selecting the best feature and partitioning the data recursively. The feature is used to divide the data into subsets, which are classified by unique characteristics. Recursion continues until optimal classification is achieved through partitioning the feature space and constructing the decision tree.

To build the decision tree, we place all training data at the root node, select an optimal feature, and partition the data into subsets based on that feature. Each subset has the best classification under current conditions. If these subsets are already classified correctly, construct leaf nodes for them and divide the subsets accordingly. If not, select new optimal features and continue segmenting until all subsets of the training data are classified or there are no suitable features. Partitioning each subset into leaf nodes generates a well-defined decision tree.

As shown in Figure 5, the specific process is as follows:

(1) Partitioning process. Iterate through all feature columns and all sample points of each feature column in turn. According to each sample point, try to divide the data set into two parts and calculate the sum of the mean squared errors of the two subtrees.

(2) Find the minimum mean square error, its corresponding feature columns and sample points. This determines the partitioning attributes and partitioning values of the current layer.

(3) After several rounds of operations, an exact and uniform partition is formed for the data set and the algorithm stops. All attributes and partition points that have a large impact on the data collection can be determined. (The graphs of the first, fouth, seventh times are shown in Fig 5)

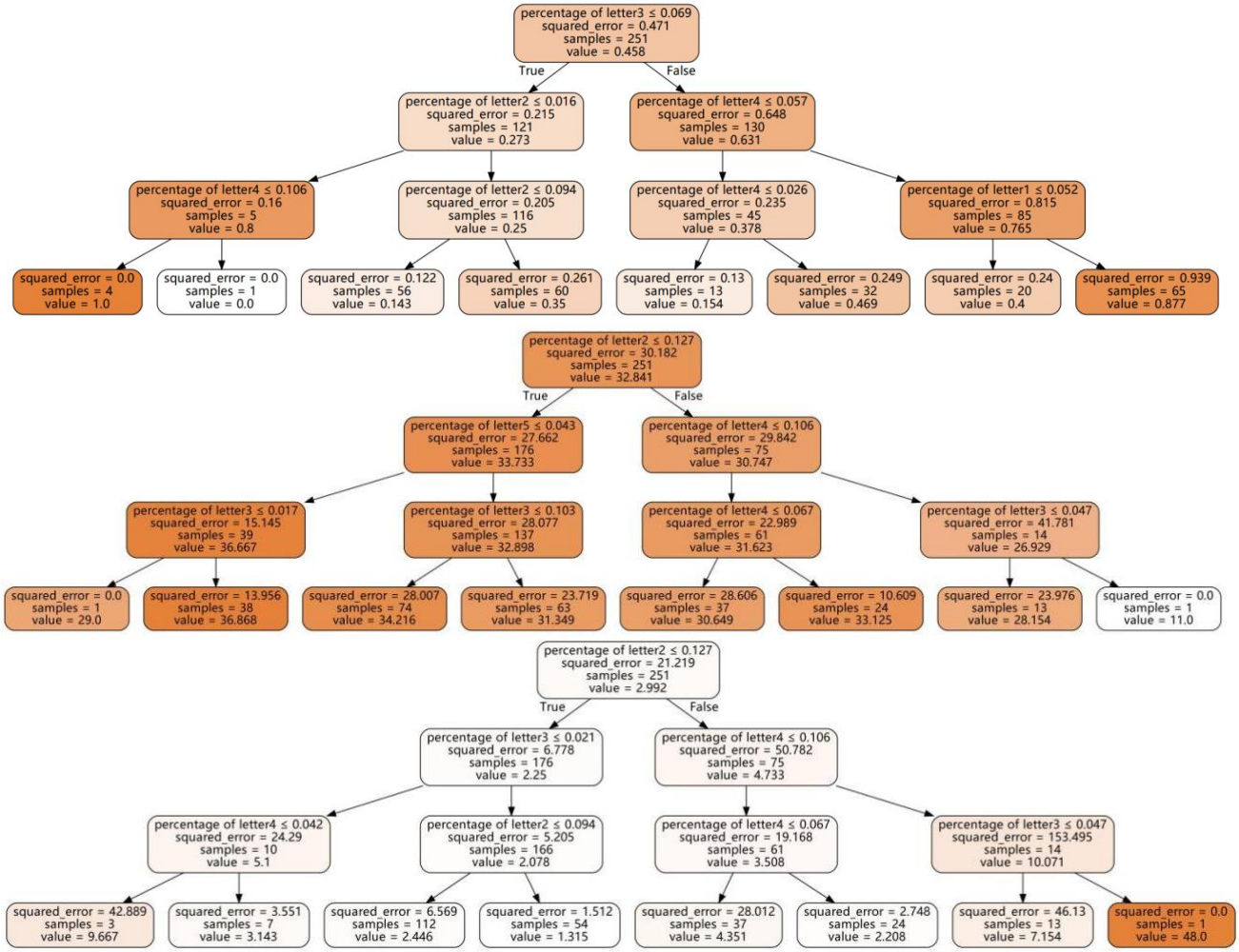

Figure 5. Decision Tree of the first, fouth and seventh times

4. Results

4.1. The establishment of simulation model

The large data prediction model for the user's electricity consumption is implemented in the Clementine software.

4.2. Analysis of experimental results

We can see that the actual values of the lines basically match the predicted values. By selecting the load prediction results of 403 and 411 lines. There are also some errors, especially in the peak period of electricity consumption, as shown in Table 3.

Table 3. Comparison of power load forecasting of 403 line

Comparison	Power	Forecasting
A	12937	92387
B	928735	29837
C	894523	23894

The BP neural network has better prediction performance and relatively small error, which can meet the demand completely, and has fast prediction speed and convenient operation.

4.3. Enhancing Wordle Gameplay with Predictive Models and Decision Trees

The study created a decision tree to help Wordle players guess new words with new dates. Players can enter a word's five letters, predicted date, and corresponding values into a table. We enhance the model with the decision tree which outputs the percentage of successful guesses predicted by the study. This approach lets players predict their chances of correctly guessing a given word.

Table 4. Letter Frequency Table

Letter	1st letter	2nd letter	3rd letter	4th letter	5th letter
a	0.078	0.138	0.123	0.089	0.026
b	0.056	0.003	0.023	0.006	0.000
c	0.092	0.014	0.026	0.054	0.014
d	0.034	0.006	0.034	0.023	0.058
e	0.028	0.081	0.083	0.011	0.208
f	0.062	0.000	0.011	0.043	0.009
g	0.048	0.006	0.026	0.011	0.020
h	0.031	0.095	0.006	0.066	0.058
i	0.028	0.066	0.123	0.000	0.006
j	0.006	0.003	0.003	0.032	0.000
k	0.008	0.003	0.011	0.069	0.040
l	0.036	0.118	0.023	0.040	0.069
m	0.056	0.017	0.020	0.066	0.020
n	0.017	0.049	0.060	0.043	0.046
o	0.011	0.135	0.140	0.023	0.032
p	0.062	0.032	0.040	0.000	0.020
q	0.008	0.006	.0.000	0.000	0.00
r	0.036	0.092	0.054	0.077	0.095
s	0.140	0.012	0.023	0.060	0.009
t	0.084	0.040	0.037	0.077	0.127
u	0.022	0.043	0.077	0.029	0.003
v	0.017	0.003	0.020	0.029	0.000
w	0.031	0.017	0.014	0.014	0.009
x	0.000	0.014	0.000	0.006	0.000
y	0.006	0.006	0.023	0.000	0.127
z	0.003	0.000	0.000	0.009	0.003

Overall, the decision tree, in combination with the Wordle Word n-tries Percentage Prediction Model. In Table 4, We provide a reliable strategy for players who seek to improve their performance and success rate in the game. By leveraging these tools, players can make more informed decisions about their game play and enhance their overall experience.

As shown in Figure 6, we take the prediction of the word "EERIE" on March 1, 2023 as an example. The percentage of guesses is [0.01, 0.05, 0.17, 0.32, 0.27, 0.12, 0.03]. Players only need to query the table, enter the eigenvalues and date.

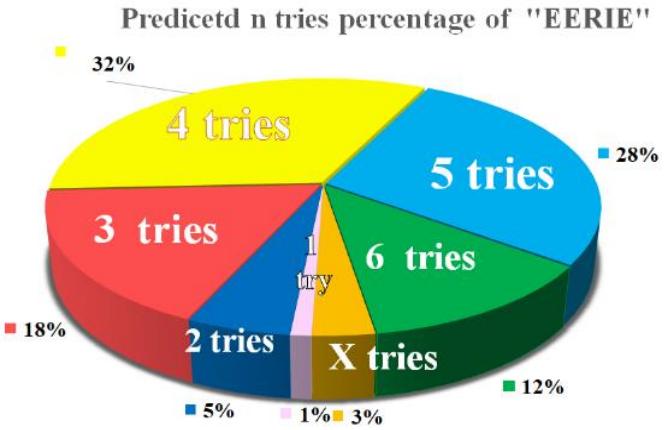

Figure 6. Percentage of n tries for the word EERIE

The respective errors are as follows

Table 5. The Analysis of Forest Tree Respective Errors

	MSE	RMSE	R^2
Train Set	3.314	24.22	0.934

As shown in Table 5, MSE proves the degree of difference between the estimated and estimated quantities is small. RMSE shows the deviation between the observed value and the true value is small, which demonstrates that the model has excellent accuracy.

The R^2 is 0.9343, which indicates how much the independent variable causes the change of the dependent variable, i.e., this model this regression model explains 93% of the prediction results.

In summary, the accuracy of the model we built is good and we have 93% confidence in the prediction of the model.

5. Conclusion

This paper offers solutions for predicting the number of wordle players the next day. In addition, Wordle Word Difficulty Assessment Model is built to assess the difficulty of a five-letter word. It aims to achieve inputting five-letter English words, outputting the predicted the percentage of different tries times on the specified date. This provides reference value for wordle game designers. A practical application of the ARIMA time series algorithm is implemented. An efficient and far-reaching combination of the Regressor Chain and Decision Tree, a deep learning algorithm. The next step will be to combine more data to classify the difficulty of words and to further improve the accuracy of the model.

References

[1] Bonthron M. Rank one approximation as a strategy for Wordle [J]. arXiv preprint arXiv: 2204.06324, 2022.

[2] de Silva N. Selecting seed words for wordle using character statistics [J]. arXiv preprint arXiv: 2202.03457, 2022.

[3] Kalpakis K, Gada D, Puttagunta V. Distance measures for effective clustering of ARIMA time-series [C]//Proceedings 2001 IEEE international conference on data mining. IEEE, 2001: 273 - 280.

[4] Li I. Analyzing difficulty of Wordle using linguistic characteristics to determine average success of Twitter players [J]. 2022

[5] Melki G, Cano A, Kecman V, et al. multi-target support vector regression via correlation regressor chains [J]. Information Sciences, 2017, 415: 53 - 69.

[6] Read J, Martino L. Probabilistic regressor chains with Monte Carlo methods [J]. Neurocomputing, 2020, 413: 471 - 486.

[7] Spyromitros-Xioufis E, Tsoumakas G, Groves W, et al. multi-target regression via input space expansion: treating targets as inputs [J]. Machine Learning, 2016, 104: 55 - 98.

[8] Jijo B T, Abdulazeez A M. Classification based on decision tree algorithm for machine learning[J]. evaluation, 2021, 6 (7).

[9] Box G E P, Jenkins G M, Reinsel G C, et al. Time series analysis: forecasting and control [M]. John Wiley & Sons, 2015.

[10] Myles A J, Feudale R N, Liu Y, et al. An introduction to decision tree modeling [J]. Journal of Chemometrics: A Journal of the Chemometrics Society, 2004, 18 (6): 275 - 285.

[11] Siyu Wei. (2022) The forecast of China's GDP baased on time series model [D], Jinan: Shandong University, 57 - 67.

[12] Lei Xiao. (2013) Anaiysis of China Online Game Vendors' Marketing Strategy [D]. Qingdao: Ocean University of China, 45 - 53.

[13] Wei chao Xu (2012) A Review on Correlation Coefficients [J] Journal of Guangdong University of Technology29 (3), 13 - 16.

[14] Gardner. D, Davies. M (2014) A New Academic Vocabulary List[J] APPLIED LINGUISTICS 35 (3), 306 - 325.

[15] Read J, Martino L,... (2015). Scalable multi-output label prediction: From classifier chains to classifier trellises. Pattern Recognition 48 (6), 27 – 65.

[16] de Silva N. Selecting seed words for wordle using character statistics [J]. arXiv preprint arXiv:2202.03457, 2022.