| Problem Chosen | 2023 MCM Summary Sheet | Team Control Number |
|---|---|---|
| C | | 2321439 |

**Predicting Wordle Results**

Wordle guessing game is an educational and fascinating logic game. In this paper, a headcount-time series prediction model based on Auto-TS was developed to construct a prediction interval for the total number of future reports. The difficulty generalization score of any attribute of the word is based on the rank and ratio evaluation method of entropy weighting. A random forest regression model was built to predict the percentage of possible rounds related to new words at future dates. Combining the predicted values, a difficulty classification model centered on the K-means algorithm was then developed to categorize the new words. Finally, the distribution of difficulty percentage is analyzed

Regarding the first question, we proposed a time series prediction model based on Auto-TS to find the best prediction value through genetic algorithm iteratively for the prediction interval of 10612 to 15130 on March 1, 2023, and tested the model with the prediction value backtracking. A comprehensive evaluation model based on the entropy weight method of rank and ratio is established to evaluate the generalized score and difficulty classification of any attribute of the existing words and observe the effect on the number of reports submitted in difficult mode.

For the second question, this team proposes to use random forest regression model for predicting the distribution of the percentage of attempts for each round of the word, extracting multiple attributes of the word, such as vowel frequency, daily use frequency, and so on, screening the feature factors to prevent uncertain attributes from increasing the difficulty on the prediction, and obtaining the optimal combination of feature factors, verifying the model by K-fold cross-validation, and the MAE, and finally the predictions of the percentage of attempts for each type of round for EERIE were 0%, 5%, 19%, 30%, 27%, 16%, and 3%.

For the third question, our team established a difficulty classification model based on the K-means algorithm, based on the optimal features in the second question combined with the distribution features of the percentage of attempts for each round, and considering that it is a difficulty classification, the distribution features of the percentage of attempts are weighted to obtain the difficulty classification model. For EERIE, the predicted distribution of the number of attempts in each round by the second question is first combined with its own word properties for difficulty classification as medium difficulty.

For the fourth question, the research team used the difficulty classification model established in the third question to classify the difficulty of the words, and explored the ranking and changes in the passing rate of different rounds through time cycle analysis, finally drawing conclusions about the characteristics of the dataset and providing constructive proposals for the interaction and optimization of the game.

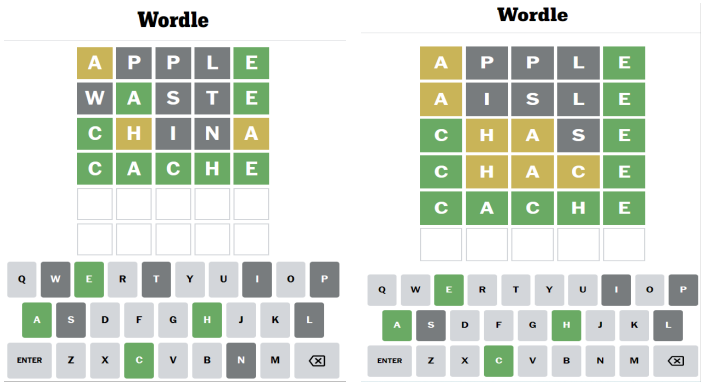
**Keywords:**Auto-ts, entropy weight method, rank sum ratio integrated evaluation, random forest regression, k-means

# Contents

# 1 Introduction

## 1.1 background information

*Wordle* is a popular daily puzzle offered by the New York Times. It is played by players guessing a five-letter word within six attempts, and each guess must be an actual word in English. Fig 1 is an example solution that found the right result in four attempts [1].




(a) Easy (b) Hard

**Fig** 1: Word puzzle solution for February 17, 2023

The game is divided into easy mode and hard mode. In easy mode, players can pick any word to guess. And in hard mode, when the player finds the correct letter in a word, he needs to use it in the subsequent guesses.

## 1.2 Restatement of the problem

This question requires an analysis of the results of the documents given by the organizers and is used to answer the following questions.

- Question one was to develop a model to explain why reported outcomes vary from day to day and to use the model to create a prediction interval for the number of people reporting outcomes on March 1, 2023, and to explain whether the attributes of the words would affect the number of difficult mode people as a percentage of the number of reported outcomes.
- Problem 2 requires the development of a model that predicts the percentage of attempts at a future date and analyzes which uncertainties the model is associated with. For example, predict the percentage of relevance of the word "$EERIE$" on March 1, 2023 and discuss the results.

- Question 3 should develop and summarize a model that classifies solution words by difficulty, identifies the attributes of a given word associated with each classification, and uses the model to account for the difficulty of the word *"EERIE"*.
- Question 4 lists and describes some other interesting features of this dataset.

## 1.3 Problem thinking

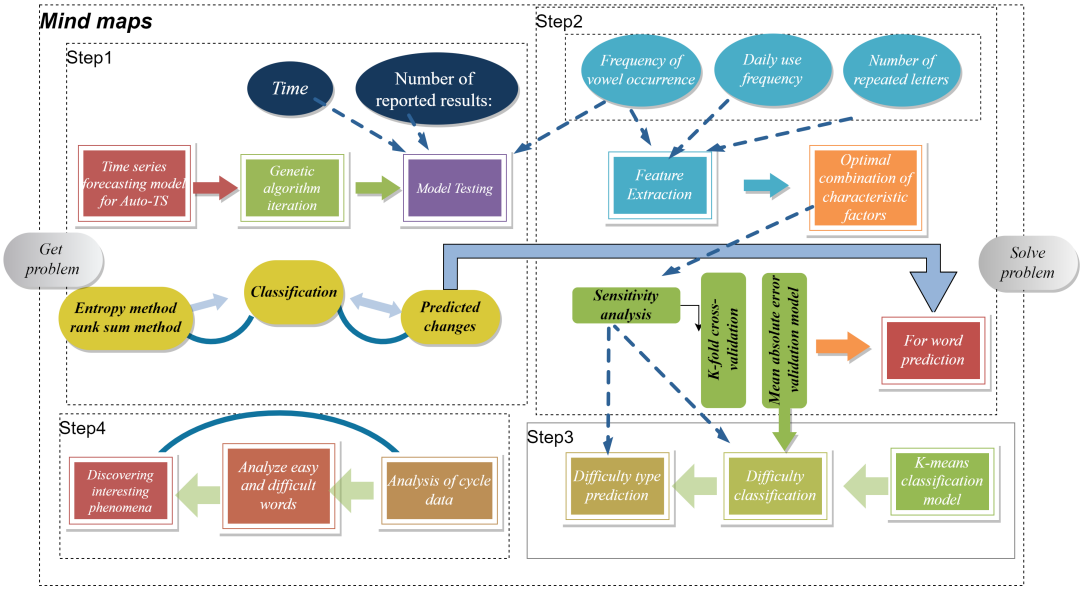



**Fig** 2: Mind maps

Ror the first question, an $Auto-TS$ based time series prediction model is proposed to find the best prediction value by iterative genetic algorithm and back-fit historical data for testing the model. Any attributes of the existing words are generalized by the rank sum ratio evaluation method to observe the effect on the number of submitted headcount reports under difficult patterns.

For the second question, a random forest regression model is used to make predictions, multiple uncertain attributes of words are screened and extracted, the reliability of the model is verified by K-fold cross-validation, and mean absolute error, and finally the percentage of attempts of each type of round is predicted for "EERIE"

For the third question, our team established a difficulty classification model based on the K-means algorithm, and considering that it is a difficulty classification, the distribution features of the percentage of attempts for each round are weighted, and the difficulty classification model is obtained on the basis of the features of the second question. For EERIE, the difficulty classification is first performed by

predicting the distribution of the number of attempts in each round and combining it with its own word feature nature.

For the fourth question, the research team used the difficulty classification model established in the third question to classify the difficulty of words, and explored the ranking and changes of the passing rate of different rounds through time cycle analysis, finally drawing conclusions on the characteristics of the dataset and providing constructive proposals for the interaction and optimization of the game.

# 2 Model assumptions and symbol instructions

## 2.1 Model Assumptions

In the real game environment, there are always many complex factors included. In this paper, we make several assumptions in our model, after which we can modify some of them to optimize our model and make it more applicable to complex realistic environments.

1. It is assumed that between February 17, 2023 and March 1, 2023, the number of games reported will not fluctuate for contest reasons.
2. It is assumed that the game will not have a word length not equal to 5 due to system problems.
3. It is assumed that small changes in the total percentage of players solving puzzles in the given data have no effect on the model.

## 2.2 symbol description

| Symbol | Meaning |
|---|---|
| $R$ | tIt refers to the ordinal number of the sample data sorted by size and is used to compare the size relationship between different samples. |
| $e_j$ | A metric for measuring uncertainty |
| $d_j$ | Information redundancy, which refers to the repetitive and unnecessary parts of the information |
| $w_j$ | The weight of the j-th indicator |
| $RSR$ | A non-parametric statistical method that converts the rank of sample data into a numerical calculation for comparing the size relationship between different samples |
| $Probit$ | A standard unit used to measure probability, typically used in probability calculations in binary response models |
| $MAE$ | A metric used to measure the error of a prediction model, calculated as the average of the absolute values of the differences between the predicted and true values |
| $C$ | The cluster center in cluster analysis refers to the average of all data within the same category and is used to represent the characteristics of that category |

# 3 Model building

*Wordle* game was loved by users as soon as it was launched, and the number of people submitting reports every day is changing over time, and the development of the game's life cycle is predicted by observing historical data, i.e. time series prediction [2].

In this section, to get better prediction results, time series forecasting is performed using $Python's\ Auto-TS$ library, which uses a genetic algorithm to iterate over the more than 100 time series forecasting models it contains to find the optimal solution, i.e., the best model, and to verify the reliability of the model by modeling back to past data.

## 3.1 The idea of question one

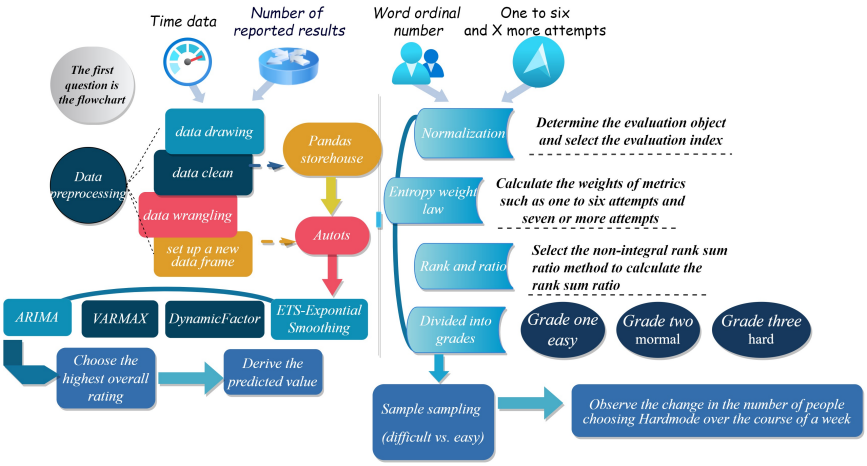



**Fig** 3: The modeling idea of question 1

After data preprocessing the pandas library was imported and the Excel file containing the data was read into a pandas DataFrame and a new DataFrame was created.

The *AutoTS* library was imported and an instance of the AutoTS model was created using some default parameters, iterating the model with high scores for priority output, and finally predicting the data with a step size of 61. The entropy weighting method was used to weight the percentage distribution of attempts and the rank sum ratio was used to obtain a composite rating of any attribute of the word, observing the percentage of difficult versus easy words reported as a percentage of total reports during the week.

## 3.2 Auto-TS based time series forecasting model

### 3.2.1 data preprocessing

First of all, we checked the anomalous data in three dimensions: "Total percentage of puzzles solved", "Word length per day" and "Percentage of people choosing difficult mode". The sum of the percentage of solved puzzles should be around 100% in each round of attempts.

The words *"tash"*, *"clen"*, *"study"*, *"robin"* and *"rprobe"* were found to be anomalies through screening.

### 3.2.2 Model overview

The open source library $Auto-TS$ in *Python* can find the best time series forecasting model using genetic programming optimization. The open source library finds the best model based on genetic programming and optimizes the hyperparameters of the time series forecasting model using genetic algorithms through $Auto-TS$, whose search process consists of the following steps, as shown in Fig 4:

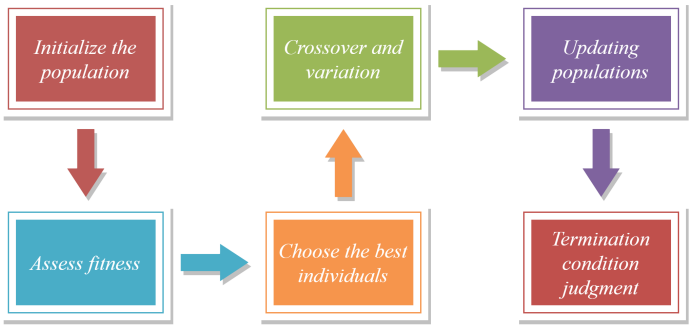



**Fig** 4: Flow chart of the genetic algorithm

First, a set of individuals is randomly generated, and each individual represents a set of hyperparameters taken. These hyperparameters include the type of forecasting model, lag order, seasonal decomposition, regularization, etc. Then each individual is applied to the time series prediction task, and its prediction accuracy index is calculated. At the beginning, the outstanding individuals are selected to perform crossover and variation operations on the parents, iterate with the new population, and judge whether the termination conditions are satisfied, such as reaching the maximum number of iterations, reaching the specified fitness threshold, etc. If they are satisfied, the search process is stopped and the optimal solution is output.

In this paper, the time series $T = \{t_1, t_2, \cdots, t_n\}$ as independent variables and the number of reporters $N = \{n_1, n_2, \cdots, n_n\}$ as the dependent variable inputs, and iterative model training by $Auto - TS$ to obtain Fig 5.

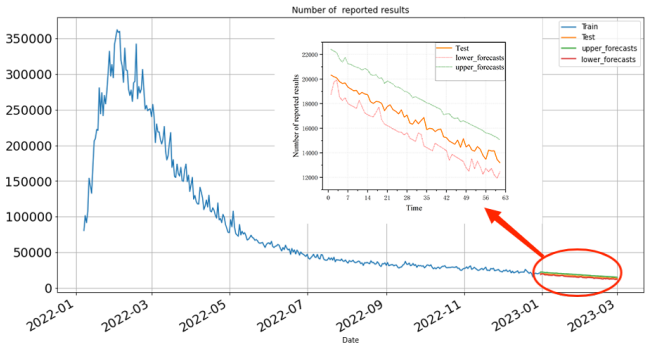



**Fig** 5: Projected headcount curve from January 1 to March 1, 2023

The $x$-line in the graph is the predicted value, which shows a general tendency towards a flat decline.The amount of change tends to level off at about 7,000 fewer reports of overall 60-day change, based on a 95%.In this paper, the predicted number of people for March 1, 2023 is in the range of 10613 to 15131, and the number of people online on that day is 13570 based on accurate software prediction.

### 3.2.3 Reliability Analysis

Reliability can be tested by backtracking the predicted values.

In this paper, the predicted values are back fitted to the historical data, and the reliability of the model is tested by whether the fitted values are within the confidence intervals in the actual historical values. The desired predicted values were obtained through the Auto-TS library. The results of the test are shown in the following Fig6.

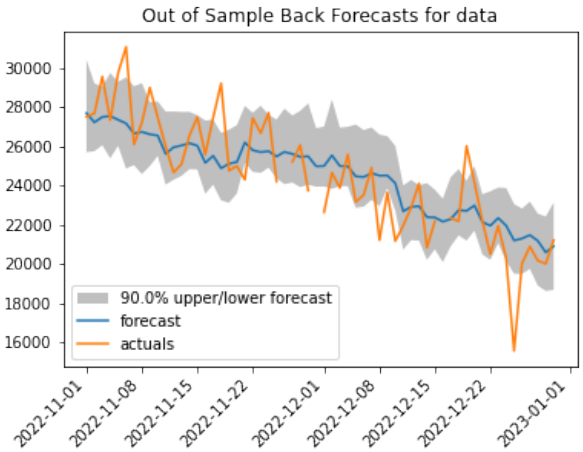



**Fig** 6: Backtrack test result Fig

As shown in Fig6, using the predicted 60-day values obtained by $Auto-TS$ to backtrack, the fitted historical data $foracst$ obtained by backtracking is compared with the actual historical data actuals from November 1st to December 31st, and it can be found that the true value is within the 90% confidence interval of the predicted value, and the reliability of the model has been tested to perform excellent

## 3.3 Entropy weight method and rank sum ratio method

This section describes a method for assessing the difficulty level of words. The method takes into account the percentage of completers with different number of attempts as an indicator. Due to the high degree of variability between rounds of attempts, it was chosen to divide them into positive and negative indicators, and to analyze the available word data based on the entropy weighting method and the rank sum ratio integrated evaluation method using $SPSSPRO$ software to derive their evaluation scores, and to classify the words into three groups: easy, moderate, and difficult.

### 3.3.1 Data normalization

First, this section classifies the percentage of completers as indicators into positive and negative indicators and normalizes them.

The positive index is

$$x_{ij}^{'} = \frac{x_{ij} - min\{x_{ij},\ldots,x_{nj}\}}{max\{x_{1j},\ldots,x_{nj}\} - min\{x_{1j},\ldots,x_{nj}\}} \tag{1}$$

The negative index is

$$x_{ij}^{'} = \frac{max\{x_{1j},\ldots,x_{nj} - x_{ij}}{max\{x_{1j},\ldots,x_{nj}\} - min\{x_{1j},\ldots,x_{nj}\}} \tag{2}$$

where $X_{ij}$ is the $j-th$ indicator of the $i-th$ word

### 3.3.2 Entropy weighting method for weighting

Entropy is a measure of the degree of chaos. The greater the degree of chaos, the greater the entropy and the greater the amount of information contained therein. The lower the degree of chaos, the lower the entropy and the less information it

contains [3].Calculate the entropy value of indicator $j$.

$$e_j = -k\sum_{i=1}^{n} p_{ij}\ln(p_{ij}), j = 1, \cdots, m \tag{3}$$

Where $p$ is the ratio of the sample value in the $i-th$ word $j$ indicator to the total sample value, $n$ represents the number of words, and $m$ represents the number of indicators. After that, by calculating the information entropy redundancy $d_j$.

$$d_j = 1 - e_j, j = 1, \cdots, m \tag{4}$$

The formula is used to calculate the weights of indicators such as one to six attempts and seven and more attempts $w_j$

$$w_j = \frac{d_j}{\sum_{j=1}^{m} d_j}, j = 1, \cdots, m \tag{5}$$

The output results are shown in Tab 1:

**Tab** 1: Entropy weight law

| item | Information entropy value $e$ | Information utility value $d$ | Weight(%) |
|---|---|---|---|
| 2 tries | 0.963 | 0.037 | 50.091 |
| 3 tries | 0.984 | 0.016 | 22.239 |
| 4 tries | 0.994 | 0.006 | 7.789 |
| 5 tries | 0.992 | 0.008 | 10.496 |
| 6 tries | 0.994 | 0.006 | 8.038 |
| 7 or more tries (X) | 0.999 | 0.001 | 1.347 |

where $e$ is the entropy value and $d$ is the information entropy redundancy. The weighting results are shown in Fig7.

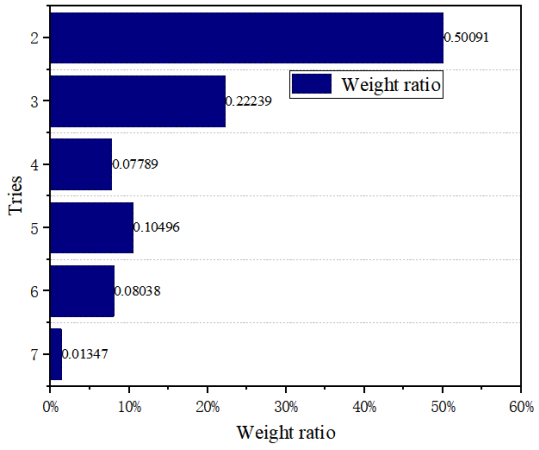



**Fig** 7: Indicator Importance Histogram

### 3.3.3 Rank and Ratio Comprehensive Evaluation Method

In this section, we choose to use the non-integer rank sum ratio method to rank the index values in a way similar to linear interpolation [4], in order to improve the shortcomings of the $RSR$ method of ranking, and there is a quantitative linear correspondence between the compiled rank and the original index values, thus overcoming the shortcomings of the $RSR$ method of ranking, which tends to lose the quantitative information of the original index values.

For the forward indicator, its rank $R\ i_j$ is:

$$R_{ij} = 1 = (n-1) \times X_{ij} \tag{6}$$

For a negative indicator, its rank R ij is:

$$R_{ij} = 1 = (n-1) \times X_{ij}^{'} \tag{7}$$

The rank-sum ratio was calculated

$$RSR_i = \frac{1}{mn} \sum_{j=1}^{m} R_{ij}, i = 1,2,\cdots,n \tag{8}$$

When the weights of each evaluation index were varied, the weighted rank-sum ratio was calculated.

$$WRSR_i = \frac{1}{n} \sum_{j=1}^{m} w_j R_{ij}, i = 1,2,\cdots,n \tag{9}$$

where $w_j$ is the weight of the $j-th$ evaluation index $\sum_{j=1}^{m} w_j\ RSR_i$ is the rank sum ratio of the $i-th$ word.

After data processing and calculation, the table of rank value calculation about *word* and the related $RSR$ ranking were obtained as shown in the Tab2

**Tab** 2: Table of word rank values calculated and associated RSR ranks

| index | $X_1$ :2 tries | $R_1$ :2 tries | $X_2$ :3 tries | $R_2$ :3 tries | $\cdots$ | RSR | RSR Rank |
|---|---|---|---|---|---|---|---|
| slump | 0.115 | 42.309 | 0.442 | 159.186 | $\cdots$ | 0.353 | 202 |
| crank | 0.192 | 69.847 | 0.442 | 159.186 | $\cdots$ | 0.363 | 187 |
| gorge | 0.115 | 42.309 | 0.209 | 75.931 | $\cdots$ | 0.228 | 307 |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| drink | 0.346 | 124.924 | 0.721 | 259.093 | $\cdots$ | 0.553 | 44 |

### 3.3.4 Rank sum ratio comprehensive evaluation model test

After obtaining the $SRS$ ranking, the words in the existing database can be sorted into 3 classes Difficult, average, easy.To verify the evaluation reliability, we performed a linear regression using the $Probit$ derived from the distribution of the $SRS$ and the existing $SRS$ and tested the evaluation model. the distribution of the $SRS$ is the value-specific cumulative frequency expressed in terms of the probability unit $Probit$ .As shown in Tab3

**Tab** 3: Distribution map of SRS

| RSR | frequency | Cumulative frequency$\sum_f$ | Evaluation rank number | Evaluation rank number $/n \times 100\%$ | Probit |
|---|---|---|---|---|---|
| 0.07578280080433214 | 1 | 1 | 1 | 0.2785515320334262 | 2.22798404811873 |
| 0.07671579063497624 | 1 | 2 | 2 | 0.5571030640668524 | 2.4617884586058754 |
| 0.08736605166795601 | 1 | 3 | 3 | 0.8356545961002786 | 2.6070406339477854 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 0.14879318470509365 | 1 | 10 | 10 | 2.785515320334262 | 3.0867050698168907 |

Using the $RSR$ distribution values in the table as the independent variable and the $Probit$ values as the dependent variable, a linear regression was performed and the results are shown in the Tab 4.

**Tab** 4: Distribution of SRS

| | t | P-value | VIF | $R^2$ | F |
|---|---|---|---|---|---|
| constant | -74.24 | *** | — | 0.988 | F=29002.197, P=*** |
| Probit | 170.3 | *** | 1 | | |

where $P-value$ of $*$indicates $p < 0.05$, P-value of $**$ indicates $p < 0.01$, and $P-value$ of $***$ indicates $p < 0.001$. $R^2$ measures the degree of fit of the curve regression, and $VIF$ is used to test whether the model has a multicollinearity problem.

The $t-test$ is a method of comparison of means, while the $F-test$ is used to compare group variance values. In the model developed in this paper, the $P-values$ of both types of tests are less than 0.001, indicating that the results are significant and the original hypothesis that the regression coefficient is 0 is rejected. The $R^2$ value is 0.988 and the $VIF$ values are all less than 10, which proves that the model model fits well and has no multicollinearity problem. According to the formula $y = -0.31 + 0.139 \times Probit$ , the $RSR$ critical value corresponding to the $Probit$ critical value is derived. As shown in Fig 8.

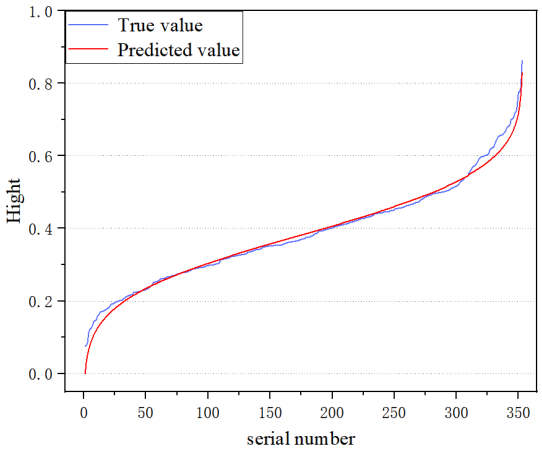



**Fig** 8: Data status classification

The raw data plot of this model shows that the model fitted values are highly similar to the predicted values, showing the same curve, so the model can be considered to work well.

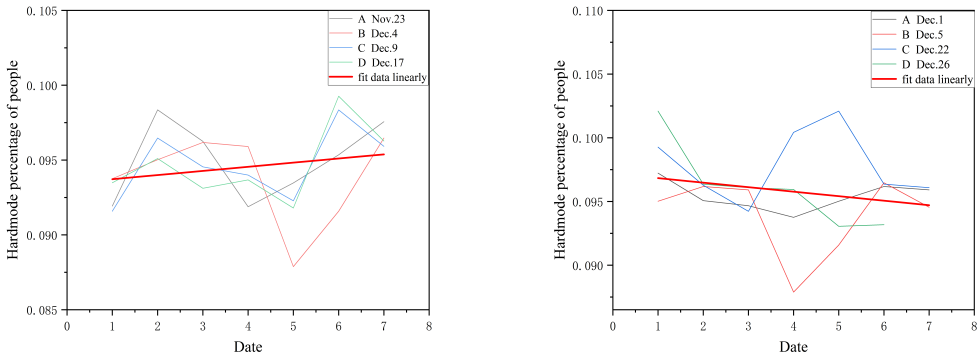
### 3.3.5 Analysis of model results

Finally, the *RSR* thresholds corresponding to the *Probit* thresholds are sorted into three grades based on the difficulty, in general, of simply dividing them into three grades, and the results of the grading are shown in the following Tab5:

**Tab** 5: Entropy weight law

| Grade | Percentile critical value | Probit | RSR Critical value |
|---|---|---|---|
| Gear 1 | [0,15.866] | [0,4] | [0,0.2477] |
| Gear 2 | [15.866 ,84.134] | [4,6] | [0.2477,0.5266] |
| Gear 3 | [84.134 ,+∞] | [6, +∞] | [0.5266 +∞] |

In response to the question of whether any of the attributes of the words affects the percentage of the number of difficulties in the report, the words in the difficulty and easy categories in the 1st and 3rd grades were selected for analysis in this paper, as shown in the following Fig9.




(a) easy (b) hard

**Fig** 9: Mid-week trend in difficulty mode as a percentage of outcome population

The Fig9 visualizes the data one week back, and the thick red line is a linear fit to this data using the built-in *Origin* program, indicating the change in the percentage of people choosing the difficult mode over the week.

The Fig 9 shows that difficult words decrease the percentage of people choosing difficult mode, while easy words increase the percentage of people choosing difficult mode in the short term, and then stabilize. Word attributes may affect the percentage of people in the difficult mode, such as multiple occurrences of the same letter or letters that appear less frequently in the English dictionary.

# 4 Random Forest Regression Model

The goal of this chapter is to predict the percentage of attempts at a future date from the uncertainty properties of words. For this purpose, we extracted six features from the existing word corpus, including vowel letter occurrence frequency, consonant letter occurrence frequency, word nature, initial letter, repetition letter occurrence frequency, and word common frequency, as sample outputs [6].

Among them, the word common frequency feature uses the brown corpus from the NLTK library in Python. For the percentage prediction problem of the number of attempts, we propose to use a random forest regression model for the predicted value output.

## 4.1 Model Overview

The random forest regression model is a supervised learning method. The randomness of the model is manifested in two aspects: samples and features. In

terms of samples, a certain number of samples are randomly selected from the training set each time as the root node samples of each regression tree; in terms of features, each decision tree is built by randomly selecting a portion of features from all the features and then selecting the optimal cut points from them, and random forest regression also has the ability of overfitting resistance and noise resistance. Its prediction process consists of three steps.

In sample points are randomly selected from the total sample $S$ to obtain a training subset of $D = \{S_1, S_2, S_3, ...S_n\}$ of the training subset

Use the sub-training set to build a decision tree, the training process requires a cut for each node, branching superiority quasi as a random selection of m features from all features, and then select the optimal cut point from the m features for the next decision tree.

The prediction results of all decision trees are averaged as the prediction results of the random forest.

## 4.2 Application of the model

### 4.2.1 Data pre-processing

Firstly, the outliers in the total sample were eliminated, and the sample matrix data were obtained by extracting the number of vowel and consonant letters from the words, labeling the lexical properties of the words and recording the first letter of the words. The following Tab 6 shows the first 10 columns of data in the database: frequency of vowel letters ($x_1$), frequency of consonant letters ($x_2$), word properties ($x_3$), initial letters ($x_4$), frequency of repeated letters ($x_5$), and frequency of words commonly used in dictionaries ($x_6$)

**Tab** 6: Characteristic factor matrix

| Index | $X_1$ | $X_1$ | $X_1$ | $X_1$ | $X_1$ | $X_1$ |
|---|---|---|---|---|---|---|
| slump | 1 | 4 | NN | s | 1 | 8 |
| crank | 1 | 4 | NN | c | 1 | 1 |
| gorge | 2 | 3 | NN | g | 2 | 1 |
| query | 2 | 3 | NN | q | 1 | 1 |
| drink | 1 | 4 | NN | d | 1 | 79 |
| abbey | 2 | 3 | NN | a | 2 | 1 |

The frequency of vowel letters, consonant letters, repetition letters, and common frequency of words are numerical features, while the rest of the features are non-numerical features that need to be quantified, e.g., $NN$, $DT$, etc. can be set to 1,2, etc., and the rest of the non-numerical features are transformed according to

the above method.

## 4.3 Model optimization and prediction

### 4.3.1 Tuning parameters

The model parameters are important indicators of the random forest regression model, and there are three influential indicators: the number of decision trees, the maximum number of features, and the maximum depth of the tree. The number of decision trees is theoretically larger, but the computation time also increases accordingly. Therefore, it is not better to obtain a larger number, but to choose a suitable number for high efficiency and accuracy.

In the selection of the number of features, the relationship between the number of features is not determined, so all features are entered by default, i.e., 6 feature variables. The maximum depth of the tree affects the degree of fit, and we set it to 5 by traversal training.

The decision tree and accuracy are analyzed and the relationship is obtained as shown in the following figure 10 FigureDecision tree vs. accuracy graph

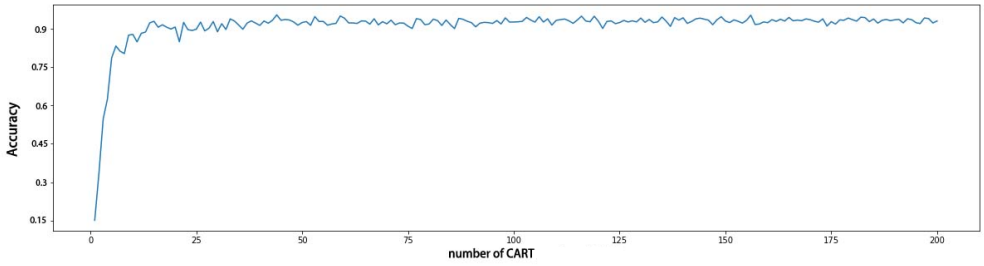



**Fig** 10: Data status classification

The horizontal coordinate is the number of decision trees, and the vertical coordinate is the accuracy rate of the model. According to the figure 10, it can be found that the more decision trees the higher the accuracy rate, and when the number of trees reaches a degree, the gain brought by the trees becomes smaller, when the number of trees is about 20, the accuracy rate is located near 0.9, so we chose the number of trees to 20, for the accuracy rate and efficiency of the subsequent model to be further improved.

After the data preprocessing and the end of parameter selection, the number of successful samples on the third attempt was predicted using a random forest regression model and the results of the model, $h(x)$, were solved as shown in

Equation 10.

$$\bar{h}(x) = \frac{1}{T}\sum_{i=1}^{T}|h(x,\theta_t)| \tag{10}$$

Where: $\bar{h}(x)$ is the model prediction; $h(x_t,\theta_t)$ is the output based on x and $\theta_t$, x is the independent variable.$\theta_t$ is the independent identically distributed random vector; T is the number of regression decision trees.

### 4.3.2 Feature Factor Screening

In order to predict the percentage distribution of the number of attempts in each round more reasonably and to prevent the occurrence of excessive deviation of a certain feature on the percentage distribution, the feature factors were screened by the control variable method to analyze the influence of different features on the prediction of the percentage of the same word in the database, such as eliminating a feature factor, leaving the rest of the variables untouched, and observing the difference between the predicted and actual values. Different combinations of features were used to find out the factors with the smallest error between prediction and actual value. The following Tab 7 shows the prediction with different features for alike. TableInfluence of different combinations of characteristic factors

**Tab** 7: Eigenfrequency processing

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| Predicted values with consonants removed | 0 | 0 | 6 | 24 | 37 | 25 | 7 |
| Predicted values with vowels removed | 1 | 8 | 23 | 31 | 23 | 11 | 2 |
| Actual value | 0 | 6 | 20 | 33 | 27 | 12 | 2 |

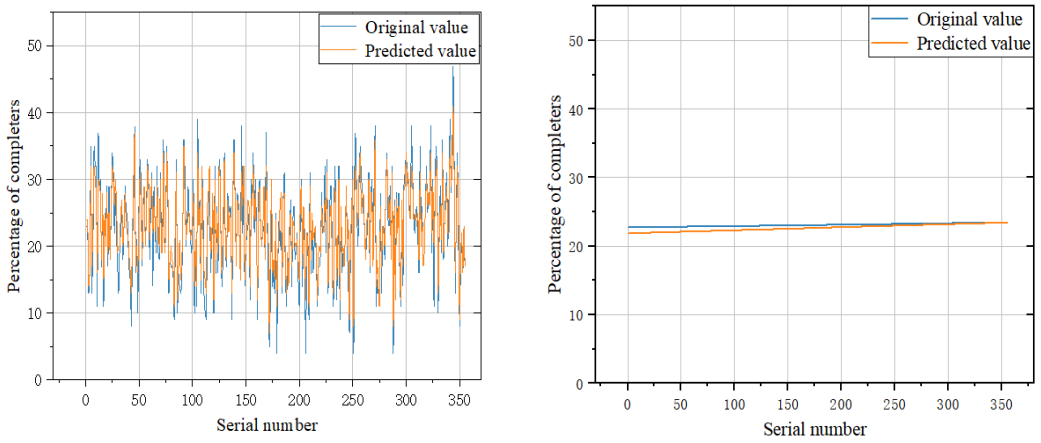
The above is the effect of two of the feature combinations on the prediction value, through multiple permutations and combinations, it was found that only the frequency of common use of words and the frequency of occurrence of repeated letters were used as the feature analysis for the optimal solution of the prediction value, and finally the word samples were compared for the prediction by the number of third attempts.

The data of the percentage of the number of passers for the third attempt were selected for demonstration, and the specific prediction results are shown in the following table 8.

**Tab** 8: Plot of the original and predicted values of the 3 tries data

| | 0 | 1 | 2 | 3 | 4 | 5 | ··· |
|---|---|---|---|---|---|---|---|
| Original value | 23 | 23 | 13 | 16 | 35 | 13 | ··· |
| Predicted value | 24 | 24 | 14 | 16 | 32 | 17 | ··· |

Afterwards, this data is visualized to obtain Fig 11 (a), and the two lines are each fitted to obtain Fig11(b)




(a) Model Fitting (b) Fitting the curve

**Fig** 11: Track Information Map

From the Fig11(ab), it can be seen that the model has excellent fitting effect with small deviation, and the goodness-of-fit of the two curves is calculated, and the value of its goodness-of-fit is obtained as 0.98183899525, and the closer the value of the goodness-of-fit is to 1, the higher the degree of fitting of the regression curve to the predicted value, i.e., the accuracy of the model is as high as 98.183%, and the prediction effect is good.

## 4.4 Testing of the model

### 4.4.1 K-fold cross-validation

K-fold cross-validation (K-fold cross-validation) is a commonly used method to evaluate the performance of machine learning models. In K-fold cross-validation, the original data set is divided into K non-overlapping subsets, and then each subset is used as the validation set in turn, and the remaining K-1 subsets are used as the training set, and the model is trained and tested for$K$ times. Finally, the average of the$K$ validation results is used as the performance index of the model. The specific validation process is shown in the following Fig10.

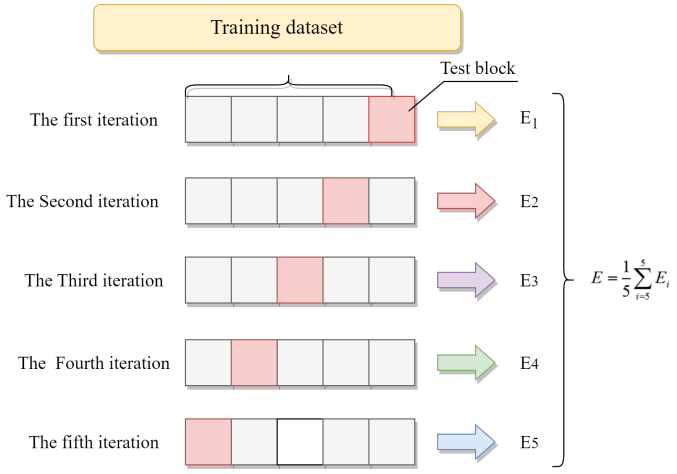



**Fig** 12: 5-fold cross-check process

The main purpose of K-fold cross-validation is to evaluate the performance and generalization ability of machine learning models and avoid overfitting the models on the training data. Since the original dataset is divided into $K$ subsets and each subset is used as a training set and a test set, the generalization performance of the model can be better evaluated. In addition, K-fold cross-validation can avoid the problem of inaccurate model performance evaluation due to poor data partitioning. A 5-fold cross-validation was performed on the model, and its goodness-of-fit R2 is shown in the following Tab 9: TableGoodness of fit $R^2$

**Tab** 9: Plot of the original and predicted values of the 3 tries data

| | $K_1$ | $K_2K_3$ | $K_4$ | $K_5$ | |
|---|---|---|---|---|---|
| Accuracy | 0.87689729 | 0.82064151 | 0.86771054 | 0.86625804 | 0.89563738 |

From the above table, it can be seen that its fit is good and the average goodness of fit is 0.865428953194731, and the model performs well in terms of performance and generalization ability.

### 4.4.2 MAE Mean Squared Error Test

Mean Absolute Error $(MAE)$ is a commonly used model performance metric to measure the average absolute difference between the predicted and true values of a model. In K-fold cross-validation, the mean absolute error of the model on different data sets can be evaluated more accurately because the data set is divided into $K$ non-overlapping subsets, each of which is used as a training set and a test set.

To further validate the model reliability, the $MAE$ mean absolute error test can be performed on it.

$$MAE = \frac{\sum_{i=1}^{n} | \text{ predicted }_i - \text{ actual }_i |}{n} \tag{11}$$

The mean absolute error is the average of the absolute values of the deviations of all individual observations from the arithmetic mean. The mean absolute error can avoid the problem of errors canceling each other, and thus can accurately reflect the magnitude of the actual prediction error [5]. The mean absolute error is denoted as MAE. MAE can evaluate the degree of variability of the data, and the smaller the value of MAE, the better accuracy of the prediction model in describing the experimental data. The number of attempts for each round of the model was calculated as the following Eab10.

**Tab** 10: Mean absolute error of the number of attempts in each round

| Number of attempts | $MAE$ |
|---|---|
| Round 1 | 0.02 |
| Round 2 | 0.06 |
| Round 3 | 0.16 |
| Round 4 | 0.12 |
| Round 5 | 0.07 |
| Round 6 | 0.08 |
| Round 7 | 0.02 |

The mean value of MAE is 0.075714286, which is highly accurate. Because the value of MAE matches well with 0 and the model is judged to be a good prediction model, this team is confident in the predictive ability of the model.

Finally, the results of the random forest prediction algorithm were compared with other algorithms, as shown in Tab 11.

**Tab** 11: Comparison of prediction results of different algorithms

| Models | Words | 1 try | 2 tries | 3 tries | 4 tries | 5 tries | 6 tries | 7 or more tries (X) |
|---|---|---|---|---|---|---|---|---|
| True Value | Slump | 1 | 3 | 23 | 39 | 24 | 9 | 1 |
| | Crank | 1 | 5 | 23 | 31 | 24 | 14 | 2 |
| Random Forest | Slump | 1 | 4 | 24 | 38 | 23 | 9 | 1 |
| Prediction Model | Crank | 1 | 5 | 24 | 32 | 23 | 13 | 2 |
| Neural Network Model | Slump | 0 | 2 | 15 | 34 | 28 | 16 | 5 |
| | Crank | 0 | 2 | 15 | 32 | 27 | 16 | 5 |

From the table, two words ”Slump” and ”Crank” were randomly selected and compared with the true values by the random forest prediction model and the neural network model. The random forest prediction model has similar results to the true value for each attempt, with the error within 1 for the first to seven attempts, while the neural network model differs from the true value by 10+ for the

third attempt. This is because the random forest prediction model has good noise immunity and can handle very high-dimensional data, and can get good results even for the missing value problem, which is one of the reasons why our team is confident in the model.

### 4.5 Analysis of the results of the model

It is assumed that the distribution of the success percentage of players' multiple attempts does not vary with the number of players over time. In this paper, a random forest prediction model is used to predict the word "EERIE".

From the above model, we can predict the expected percentages of each attempt of the word, the pass rate of the first attempt is 0%, the pass rate of the second attempt is 5%, the pass rate of the third attempt is 19%, the pass rate of the fourth attempt is 30%, the pass rate of the fifth attempt is 27%, the pass rate of the sixth attempt is 16%, and the pass rate of seven or more attempts is 3%.

## 5 K-means algorithm difficulty classification model

In this chapter, a model is developed to classify the difficulty of solution words based on the influence of their attributes. Based on the idea of the second question, firstly, "frequency of frequent use of words" and "frequency of repeated letters" were selected as feature values in the third question, and based on the prediction model in the second question, the pass rate of each round of attempts was also considered as a feature for the difficulty classification. The clustering analysis is based on the prediction model in the second question. Based on the similarity principle, data objects with high similarity are classified into clusters of the same class, and data objects with high dissimilarity are classified into clusters of different classes, so that data points in the same group are more similar to each other than to data points in other groups, and thus the difficulty is classified.

### 5.1 Feature Selection and Extraction (Algorithm Process)

First, after selecting the three features of K-means algorithm, we need to extract them separately. Using the features extracted in the second question, i.e., "word frequency features" and "repetitive letter frequency features", combined with the "percentage distribution of multiple attempts", we obtain a n*m matrix, as shown

in Equation 12.

$$X_{ij} = \begin{Bmatrix} X_{11} & X_{12} & \cdots & X_{1m} \\ X_{21} & X_{22} & \cdots & X_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{nm} \end{Bmatrix} \tag{12}$$

The data of n rows of words about m features are obtained, where $i$ is the number of rows of selected words. After finishing sorting the data, we randomly select $k$ initial clustering centers $C_i$ $(i1k)$ from the data set, and calculate the Euclidean distance between the remaining data objects and the clustering center $C_i$ to find the clustering center $C_i$ closest to the target data, the formula is shown below.

$$d(X, C_i) = \sqrt{\sum_{j=1}^{m} (X_j - C_{ij})^2} \tag{13}$$

where $X$ is the data object; m is the dimension of the data object;; $C_i$ is the ith cluster center $X_j$, and $C_{ij}$ is the $j-th$ attribute value of $X$ and $C_i$. We assign the data objects to the clusters corresponding to the clustering center $C_i$, calculate the average value of the data objects in each cluster, use it as the new clustering center for the next iteration, and wait until the clustering center is unchanged in the whole data objects is, the number of iterations stops.

## 5.2 Determination of k-values for clustering

The selection of k-value is generally determined by two methods, one is the elbow rule and the other is the contour coefficient. In this paper, the elbow rule is chosen for the determination of k-value, as shown in Figure 13.

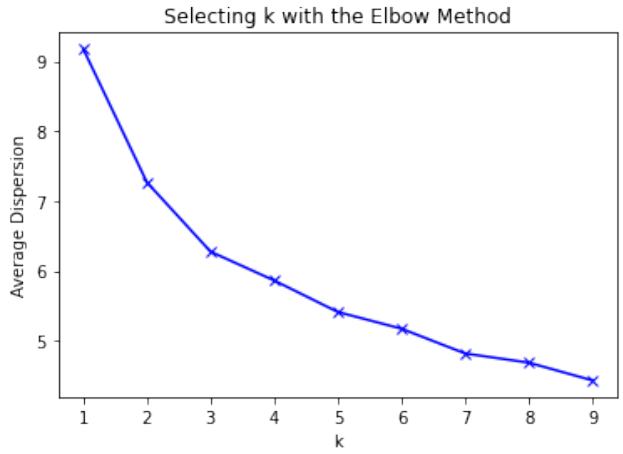


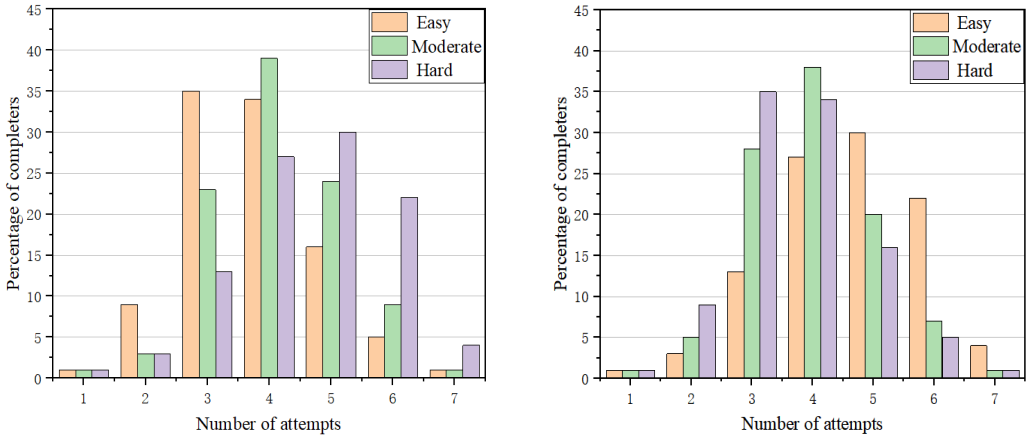

**Fig** 13: Clustering effect of elbow rule method

As shown in Fig13, the horizontal coordinate is the number of clusters and the

vertical coordinate is the average distance of class clusters. From the figure, we can see that when the number of class clusters is 1 or 2, the average distance of samples from the class clusters they belong to decreases very fast, which indicates that changing the number of clusters K will have a great change on the overall clustering structure, that is, such a K value does not reflect the real number of class clusters and cannot effectively cluster the data. And when K=3, the decline of the average distance starts to slow down significantly, indicating that the K value at this time is the relatively optimal number of class clusters. Therefore, data clustering analysis can be performed based on K=3 to obtain better clustering effect and based on the rules of the game, with the data of the last year, we choose to determine the number of clusters as three classes, i.e., easy, medium, and difficult.

## 5.3 Model prediction validation

Based on the K-means algorithm difficulty classification model, our team classifies the database, and in order to verify that the feature factors are picked reasonably, this paper also adds the number of vowels and consonants into the model to retrain the classification, and obtains the difficulty classification data table based on two different feature categories, and picks a word randomly from the prediction results in the three categories of simple, medium, and difficult words The pass rate of each attempt is visualized as shown in Figure 14.




(a) Number of vowels and consonants excluded (b) Number of vowels and consonants added

**Fig** 14: Track Information Map

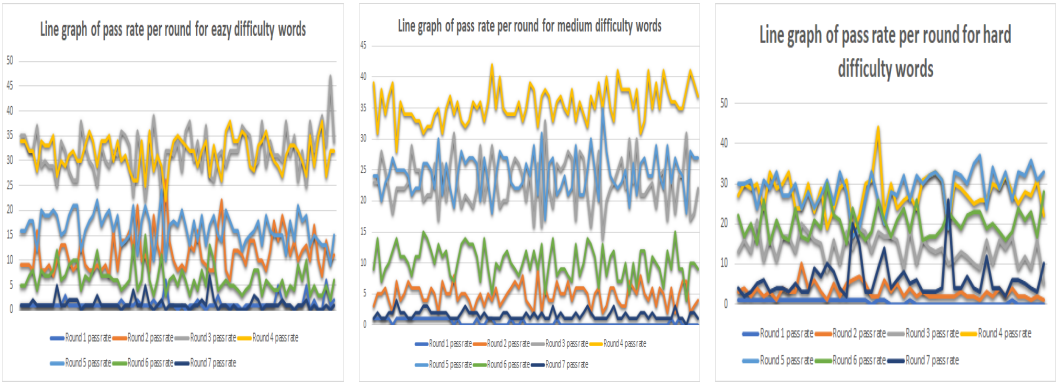
As shown in Fig.14, the difficulty classification model based on ”word frequency”, ”repetition frequency” and ”multiple attempts percentage distribution” is superior to the difficulty classification model with the addition of vowels and consonants. The difficulty classification model based on the ”frequency of frequent use of words”, ”frequency of repeated letters” and ”percentage

distribution of multiple attempts" is better than the difficulty classification model with the addition of vowels and consonants. The correctness of excluding the number of vowels and consonants in the second question was verified.

For the problem of predicting and classifying words by difficulty given in the future. First of all, we need the distribution of the percentage of attempts for each round of the given word. Combining with the model in the second question, we can predict the distribution of the percentage of attempts for the given word, get the expected percentage distribution and then substitute it into the cluster analysis model, which can classify the difficulty of the word by combining the frequency of common use of the word and the frequency of occurrence of repeated letters, etc. Finally, the word "EERIE" was categorized as moderately difficult.

# 6 Reasons for percentage distribution of attempts by round

In the third question, we performed a clustering categorization analysis of existing words and obtained three indicators of different difficulties . The difficulty was classified mainly based on the passing rate of the different rounds, and its characteristics were more consistent with the fact that their rounds were passed. Based on this basis, the three labeled words can be reextracted and classified to obtain a graph of the pass rate over time.It is showed in Fig15 .




(a) Easy (b) Medium (c) Hard

**Fig** 15: Track Information Map

From the analysis of the words of easy difficulty, it can be seen that the third and fourth rounds have the highest clearance rate among all the clearance rates, and the curve fit is very obvious. However, why is the clearance rate not the highest in the first and second rounds in the easy mode? This is because the entropy of the information provided by the correct word hints in the first and second attempts

is not ideal, which leads to the first pass rate being in last place in all difficulties (just like buying a lottery ticket). In the medium difficulty, the fourth round pass rate was the highest value, and the third and fifth rounds oscillated in the second and third rankings, which shows that the fourth round is an important watershed in the difficulty classification. In the difficult difficulty, the fourth and fifth round pass rate is highly concentrated, and the third and fourth round pass rate is also highly close to the fourth and fifth round pass rate. It is worth noting that the seventh round pass rate fluctuates in the process at a higher rate compared to the other two difficulty fluctuations, which also proves that the division of its difficulty is scientific and reliable.

Over time, although the number of people submitting reports fluctuates with the game's life cycle, there is no obvious trend change in the difficulty division of words and the change in pass rate, which indicates that the game has a more stable and benign difficulty and pass rate setting, which will be a good interactive experience for players.

# 7 Strengths and Weakness

## 7.1 Strengths

- The cluster center based on k-means difficulty classification can be used to represent the characteristics of the category, and the word itself can be added to better classify the difficulty of the location sample
- The random forest model used in this paper has results closer to the actual values than other algorithms, and its training speed is fast.
- Auto-TS has automation, high precision, strong ease of use, etc., which can allow users to understand the prediction results and model performance more intuitively.

## 7.2 Weaknesses

- The k-value based on k-means difficulty classification needs to be set manually, and it may be difficult to determine the number of clusters. It is difficult to specifically classify words with insignificant word characteristics, which may produce unreasonable difficulty classification.
- There may be many similar decision trees in the random forest model, masking the real results, and may not produce good classification for small or low-dimensional data.

# References

[1] https://www.nytimes.com/games/wordle/index.html

[2] Kummer L , Nievola J C , EC Parałso. A Key Risk Indicator for the Game Usage Lifecycle[C]

[3] WANG Yangchen,LIN Jianeng,SU Zhiyong. Quality evaluation of big data based on gray entropy weight method[J].Microcomputer Applications,2022,38(01):110-113.)

[4] TIAN Fengjiao. Rank sum ratio method and its application[M]. Beijing: China Statistics Press, 1993.

[5] Jia Junping, He Xiaoqun, Jin Yongjin. Statistics (2nd ed.) [M]. People's University of China Press, 2004. http://news.sohu.com/20140308/n396260442.shtml , 2014-3-8/2016-1-24.

[6] Hutter F., Kotthoff L., Vanschoren J.: Automated Machine Learning, p. 9 C 13. The Springer Series on Challenges in Machine Learning

Shahriari, B., Swersky, K., Wang, Z., Adams, R., de Freitas, N.: Taking the human out of the loop: A review of Bayesian optimization. Proceedings of the IEEE 104(1), 148C175 (2016)

# Wordle puzzle game

Dear New York Times Editor:

We are honored that such a wonderful and fun game was developed during the epidemic, and over time it became a game that many people played at home as well as a way to have fun with our daily word study. We were able to expand our vocabulary, reinforce and learn new words by playing crossword puzzles. At its peak, with millions of users online every day, it was a great game that took the world by storm. Nowadays, the number of people online is decreasing, and the number of people online every day is gradually decreasing, some people may persist for a month or three months, but eventually they can't continue.

Through dynamic modeling, data analysis of 358 days of data in 2022, based on fitting and predicting the curve of the number of people online with existing data, we applied the time series model to predict the total number of people reported on March 1, 2023, roughly between 10613 and 15131, and by making a difficulty level evaluation of any attribute of the word based on the entropy weight method rank and ratio comprehensive evaluation method. It was observed that if a difficulty level word appeared on that day, the percentage of difficulty patterns in subsequent days would first decline, and after decreasing to a certain degree, a slow rebound would be carried out, and overall in a week, the number of users showed a decreasing trend. And when there is an easy word, the percentage of difficult mode will appear to rise, fall and rise again in a period of time, and the percentage of difficult mode remains stable or even slightly elevated. It is possible that this is because the appearance of difficult or easy words reduces or enhances the self-confidence of the number of people who choose the difficult mode, so there will be an increase or decrease in the short term, and then the influence decreases in the following days and tends to level off overall.

By building a random forest regression model we predicted for EERIE the percentage of rounds one to six and higher respectively, 0%, 5%, 19%, 30%, 27%, 16%, 3%, by which the distribution of the percentage of attempts for each round for a given word can be predicted, for the uncertain attributes, we also found that the most influential features among them are " Word frequency" and "number of repeated letter occurrences". We guess that everyday words are more likely to be thought of and are often the first choice for players, while the frequency of repeated letters is due to the fact that when a letter is correctly guessed, one does not think of subsequent occurrences of that letter, but makes attempts to obtain more information about other letters.

For a given word EERIE predicted the percentage distribution by regression model, and finally the word difficulty was classified using the k-means algorithm, which was found to be of medium difficulty, and the classification was believable based on the frequency of daily use of EERIE and the number of letter repetitions.

In addition to this, we analyzed the data for the reasons behind the percentage distribution of attempts in each round. For the easy difficulty, the pass rate was mostly distributed in 3,4 attempts, and the number of successful attempts in the first and second rounds did not stand out, probably due to the small amount of information carried. The medium difficulty was highlighted in the 4th round, while in the difficult difficulty, it was concentrated in the 5th round and beyond.

To this end, we would like to make the following recommendations:

1. at this point the game enters a plateau and the number of game users changes little, players can be attracted by adding gameplay, such as hiding a 5-word riddle on the current newspaper page to increase interest.

2. The word of difficulty or easiness will affect the number of players in subsequent days, which can be combined with our model to adopt some strategies to extend the game life cycle, such as growing the number of customers when the prediction is easy difficulty, and increasing the challenge of the game with the degree of difficulty when it grows to a certain degree to increase the stickiness of the game.

3. An initial difficulty screening can be performed by investigating the relevant attributes of the words and classifying the difficulty level with the model we provide, making it possible to adopt the strategy more efficiently.