
Riddle of Wordle: Mining the Secret of Number Scores & Solution Words

Summary

Wordle is a popular puzzle currently offered daily by the New York Times. The simple rules and clever propagation properties have contributed to its popularity. In this article, we build two prediction models for the prediction of the Twitter report number intervals and result distributions, respectively, and develop a model for classifying the difficulty of solution words.

In TASK 1: After data preprocessing, we build a Wordle report number prediction model based on 3rd-order gaussian regression and a non-homogeneous Poisson process from a statistical perspective. Among them, the **Gaussian regression** is used to predict the trend signs of report numbers, while the **non-homogeneous Poisson process** predicts the stochastic fluctuations of report numbers on this basis. Moreover, we use the popularity relaxation function to correct the stochastic process, which better approximates the popularity change. At a confidence level of 75%, we predict the interval of the number of reports on March 1, 2023 to be [7654, 20154]. In addition, we extract 8 attributes of words in terms of the number of letters, letter location and so on, finding that these attributes **did not have an effect** on the percentage of players' Hard Mode choices. Players' confidence in their performance ability and their play mentality may be the main reasons for whether they choose the Hard Mode or not.

In TASK 2: We first extract the data features that affect the distribution of reported results, including word attributes, and the percentage of difficulty patterns. Then, we build a BP neural network to make preliminary predictions on the distribution of guessing results for a certain solution word in the future. To improve the generalization performance of the prediction results, we build an **integrated BP neural network** based on Bagging. Then, we predict the distribution of the reported results of EERIE on March 1, 2023 as **(0, 1, 6, 25, 31, 25, 13)** (in %). We have more than 80% confidence that the absolute error of the predicted outcome for the percentage of each possible result does not exceed 5%.

In TASK 3: First, we build a word difficulty induction model based on the **K-Means** from the distribution of user's reported data, and divide the difficulty into 4 classes. Then, we explore the association between word attributes and difficulty based on **Pearson's coefficients**, and take the attributes with correlation coefficients greater than 0.6 as difficulty classification attributes to build a word difficulty classification model. Moreover, we find that the frequency of the first and second letters of the solution words, the number of vowels contained in the pronunciation and the number of word properties have a high correlation with the difficulty classification. Finally, the difficulty classification result of EERIR is **the most difficult**.

In TASK 4: While exploring the statistical properties of the number of reports, we find that the distribution of the number of reports showed a similar pattern to its trend over time. In addition, we also notice that the percentage fluctuation of 3 tries to complete the game was the largest in the 359 days of reported outcome distribution data.

Finally, we perform a sensitivity analysis of the model and investigate the effect of changes in the variable parameters of the model on the results.

Keywords: Gaussian regression; Poisson process; BP neural network; K-Means

Contents

1 Introduction	3
1.1 Problem Background	3
1.2 Restatement of the Problem	3
1.3 Literature Review	3
1.4 Our Work	4
2 Assumptions and Justifications	5
3 Notations	5
4 Data pre-processing	6
5 Task 1: Report Number Prediction Model & Game Mode Selection Analysis	6
5.1 Data Exploration	7
5.2 Wordle Report Number Prediction Model	8
5.3 Analysis of Game Mode Selection	11
6 Task 2: A Prediction Model for The Distribution of The Reported Results	14
6.1 Building the BP Neural Network-based Prediction Model for the distribution of word-guessing results	14
6.2 Analysis of Uncertainties Affecting the Model	16
6.3 Analysis of the Results of the Prediction Model	17
7 Task 3: Word Difficulty Classification Model	17
7.1 The Establishment of Word Difficulty Classification	18
7.2 Analysis of Word Difficulty Classification Results	20
8 Task 4: Other Interesting Features	21
9 Sensitivity Analysis	22
10 Model Evaluation and Further Discussion	23
10.1 Strengths	23
10.2 Weaknesses	23
10.3 Further Discussion	23
11 Conclusion	23
References	24
Letter	25


1 Introduction

1.1 Problem Background

Homer is a term used in the sport of baseball and is an informal American English word. Amazingly, Homer (home run) was searched over 79,000 times on the Cambridge Dictionary website and was searched 65,401 times on May 5. With that, Homer became the Cambridge Dictionary's 2022 Word of the Year. You may be wondering why, but it starts with Wordle, a very popular word-guessing game overseas. In 2022, the online puzzle game Wordle was all over social media. And Wordle's answer that day was Homer, which was difficult for non-US users who were not familiar with the word.

Wordle is currently a popular daily puzzle offered by The New York Times and has grown in popularity with more than 60 versions available. Players can choose between "regular mode" or "hard mode. Players attempt to solve the puzzle by guessing a five-letter word in six or fewer attempts, with each guess receiving feedback and a change in the color of the tile (green, yellow, gray). Note: Each guess must be a real word in English. Guesses that are not recognized as words by the contest are not allowed.

: A green tile indicates that the letter in that tile is in the word and in the correct location.

: A yellow tile indicates that the letter in that tile is in the word but in the wrong location.

: A gray tile indicates that the letter in that tile is not included in the word.

1.2 Restatement of the Problem

Considering the background information and the results in this file, we need to solve the following problems:

- Develop a model to account for changes in the number of reported outcomes and create a prediction interval for the number of reported outcomes on March 1, 2023. Analyze the extent to which attributes of words affect players' mode choices.
- Develop a model to predict the distribution of reported outcomes. Analyze the uncertainty factors that exist in the model and predictions.
- Develop a model to classify solution words by difficulty. Identify the attributes of the words associated with each classification.
- Describes other interesting features of the dataset.

1.3 Literature Review

In recent years, with the popularity of the Internet, social networks have gradually become the main medium for discussing what is happening in the real world, and users can generate and disseminate rich data streams on social platforms (e.g. Twitter) to gain insights into hot events that are happening. Popularity modeling and prediction have a wide range of applications in marketing, opinion monitoring, advertising and other scenarios, and time-series-based trend analysis is a research topic that has received much attention in the field of data mining and social network analysis in recent years. The idea of this type of research mainly draws on financial and epidemiological models. Shen et al ^[1] first established a Reinforced Poisson

Processes (RPP) model to predict dynamic prevalence using a heterogeneous Poisson process model, and considered the "rich get richer". Zhao et al [2] developed a SEISMIC model based on the theory of self-excited point processes, assuming that past popularity will affect the future evolution of the process, and used a double stochastic process to portray the contagion of information. Wu et al [3] proposed a Bayesian network-based popularity prediction model (EPAB) based on temporal characteristics, user characteristics and network structure characteristics, and proposed the concept of early patterns to establish the relationship between early feature information and future heat changes.

However, the time series model requires the data set to contain timing information, and the data set that does not meet this condition cannot be modeled. Meanwhile, the sequential model and the deep learning method based on node behavior dynamics are not suitable for the forecast situation of this task based only on the reported data. On the one hand, the existing data set does not contain specific information such as who the reporter is, how many players there are at any given time, etc., so a node model cannot be built based on this data set. On the other hand, techniques such as deep learning are not well interpretable and cannot explain the trend of heat change mathematically, and require more training data.

In this paper, we try our best to extract all the information from the Data File. Aiming at the specific application scenario of Wordle, we not only realize the interval prediction of the number of future reports, but also carry out further analysis on the distribution of report results and the classification of word difficulty.

1.4 Our Work

We put forward three models to mine the information of the reported result data. The structure of our paper is shown in Figure 1.

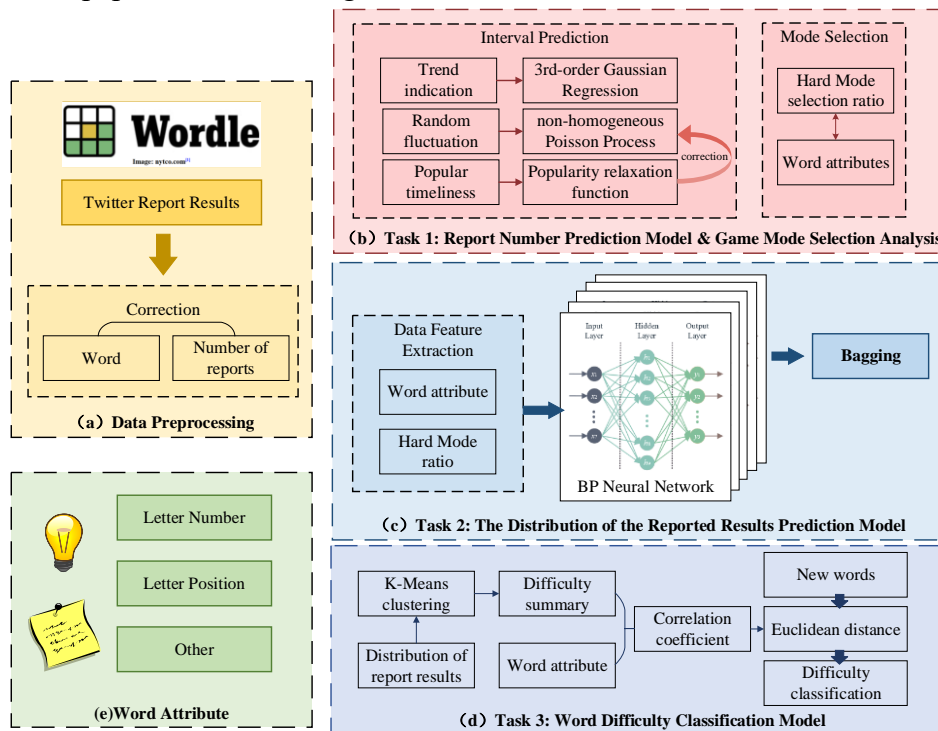


Figure 1: The structure of our paper

The rest of this paper is organized as follows. In Section II, we introduce the premise assumptions and justifications, and common variables in the formulas are mentioned in Section III. In Section IV, the data preprocessing before modeling is carried out. Section V establishes the prediction model of report number interval and explores the relationship between word attribute and pattern selection. In Section VI, we establish the report result distribution prediction model. In Section VII, we propose the word difficulty classification model. Section VIII continues to explore interesting features of the Data File. In Sections IX and X, the sensitivity of the model is analyzed and we further evaluate the advantages and disadvantages of the models. Finally, Section XI gives out the conclusions.

2 Assumptions and Justifications

We make some general assumptions to simplify our model. These assumptions together with corresponding justifications are listed below:

1. It is assumed that the change in the number of users in the report is a true reflection of the change in players in the actual situation.

There may be players who are enthusiastic about the game but do not tweet their results, so the number of reported users is often less than the actual value. However, we assume that players are willing to share their game results.

2. Assume that the game can be played only once per person per day, and that the questions are updated at 0:00 EST every day.

This assumption serves as the established rule of the game. This rule reflects the analyzability of the reported data. Also, it demonstrates the original intention of the game designer, Wardle, who "did not want players to spend more than three minutes per day".

3. It is assumed that in the game's setting, players are seen as people with a certain level of literacy and problem-solving skills.

There is no special connection between the words given in each game, but the player's mastery of the vocabulary directly determines the steps, speed, and correctness of the answers. We assume that the player has the ability to solve the problem and has the option to find the answer online when he cannot guess the answer.

4. It is assumed that the historical data is a good representation of all possible Wordle questions and player answers.

Since we only have 359 days of reported results data for 2022 and as the only reference data set. The data may be unrepresentative, and for the sake of analysis, we assume that it can show the question-and-answer patterns to some extent.

3 Notations

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

Table 1: Notations used in this paper

Symbol	Description
t_i	time, where i represents the number of days from that date to January 7, 2022

$y(t_i)$	number of results reported on the day t_i
$\lambda(t)$	the mean value of the number of reports on the day t
f_α	frequency of a given letter α in 359 words of result data
p_{mn}	the ratio of words with the n th letter m to all words
k	number of clustering algorithm centers of mass

4 Data pre-processing

Before building the model, a preliminary check of the data in the report is needed. According to the rules of Wordle, each word is 5 letters long. But there are unusual statistics of 4 or 6 letters in the data. Errors in words can interfere with the analysis of word attributes later, so the word data were corrected based on past answer data¹. Based on the relationship between the number of reports, we found that there was a large deviation between the number of results of No. 529 and the values on the before and after dates. Therefore, we considered it as abnormal data and corrected it by taking the average value of the data for each of the two days before and after. The overall pre-processing process is shown in Figure 2.

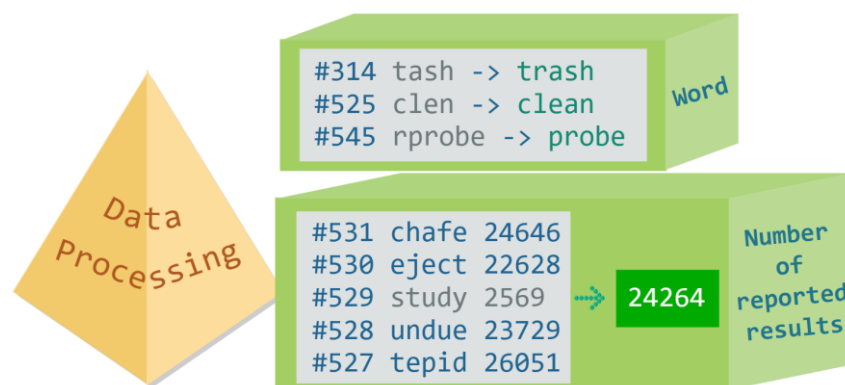


Figure 2: Data pre-processing

5 Task 1: Report Number Prediction Model & Game Mode Selection Analysis

To explore the variation pattern of the number of reported results obtained from Twitter over time, we first develop an interpretable model for describing and predicting the number of reports. From a statistical point of view, we portray the long-term temporal trends and stochastic fluctuations in the number of reports based on 3rd order gaussian regression and non-homogeneous Poisson process, respectively. Moreover, we observe that the size of the stochastic fluctuations in the number of reports is not only time-dependent but also related to the current heat level, so we introduce a popularity relaxation function to modify the stochastic process model. Finally, we enumerate eight attributes related to words and analyze the influence of

¹ Data source: <http://www.stockq.org/life/wordle-history.php#all>

word attributes on the choice of game mode with players through scatter plots.

5.1 Data Exploration

The number of reported results keeps changing over time, and Figure 3 shows the dynamic pattern of game hotness over time from the perspective of the number of people (the date takes January 7 as the starting point). In general, there is a certain pattern of popularity propagation law in the number of reports on the time scale. When Wordle exploded in the early days, there was a significant increase in the number; however, when the popularity period passed, the number showed a downward trend and leveled off, as shown in Figure 3(a). It is worth noting that there is a small random fluctuation between the number of reports per day and the overall trend. In addition, Figure 3(b) depicts the growth statistics of the cumulative number of reports over time (359 days in total). In other words, the process of the number of reports over time can be divided into two parts, which are trend signs and random fluctuation.

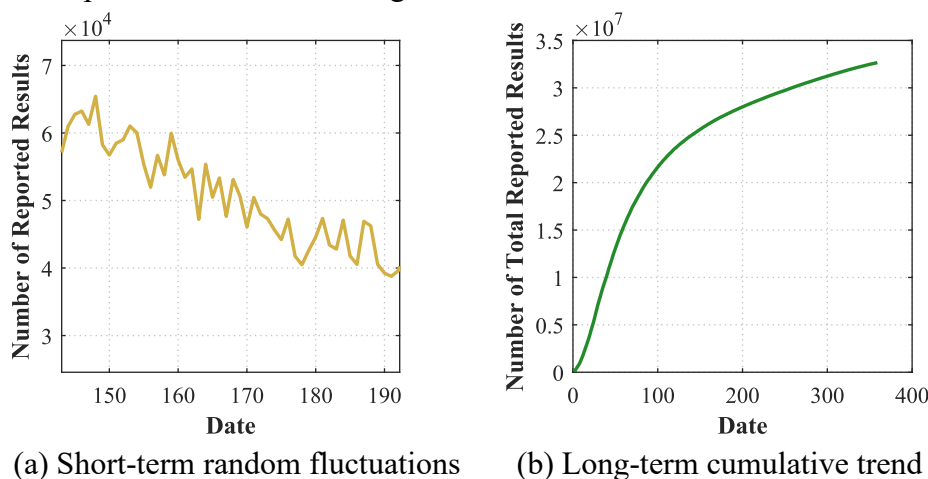


Figure 3: Description of the number of reports

At the same time, we find that there is a correlation between the fluctuation of the number of reports within a short period and the number of reports of the game in that period. Considering the social property of sharing game reports on Twitter, we approximate that the number of reports in a certain period represents the recent hotness of the game.

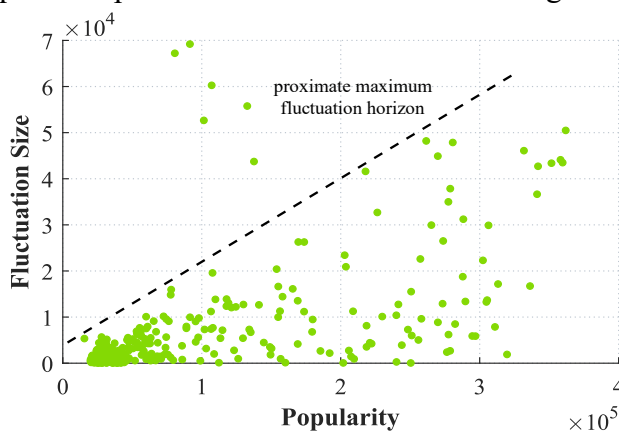


Figure 4: The relationship between the size of fluctuations in the number of reports and the popularity of the game

As shown in Figure 4, its horizontal axis is the two-period moving average of the number

of game reports, and its vertical axis is the fluctuation of the daily number of reports relative to the two-period moving average of the day. It can be seen that as the mean value of the sliding window for the number of reports increases, the magnitude of fluctuations becomes larger, and the boundaries of the magnitude of fluctuations appear roughly linear. It is more difficult to accurately predict the number of game reports in this period when the game is popular.

5.2 Wordle Report Number Prediction Model

5.2.1 The establishment of report number prediction model

We want to build a mathematical model based on existing data to describe the process of changing the number of reported results on Twitter over time and to predict the popularity in a certain period in the future, and the model is explanatory for the change process. The problem is a popularity prediction problem that has often been discussed in recent years.

By reviewing the literature ^[4], we learned that two classes of heat prediction algorithms are commonly used in the industry, including temporal models based on node behavior dynamics and deep learning-based methods. However, they do not apply to the scenario studied in this paper. This is mainly due to the following two reasons:

- 1) The existing dataset does not contain specific information such as who the reporter is and how many people in total. It is not sufficient to build a node model based on this dataset.
- 2) Deep learning does not have good interpretability and requires more training data to achieve better prediction results.

Therefore, we developed a Wordle Report Number Prediction model based on 3rd-order Gaussian regression and a non-homogeneous Poisson process from the statistical perspective.

● A trend prediction model based on Gaussian regression

In the Data File, there is a clear sign of a trend in the time series of the number of reports. We tried several regression algorithms to fit the trend in the number of reports over time, and the best result was a 3rd order Gaussian regression with a regression equation of:

$$G(t; \boldsymbol{\theta}) = A_1 \exp\left[-\left(\frac{t-B_1}{C_1}\right)^2\right] + A_2 \exp\left[-\left(\frac{t-B_2}{C_2}\right)^2\right] + A_3 \exp\left[-\left(\frac{t-B_3}{C_3}\right)^2\right] \quad (1)$$

where $\boldsymbol{\theta} = [A_1, A_2, A_3, B_1 \dots C_1, C_2, C_3]$ is the regression coefficient and t is the time in days.

Then, we use the least squares method to regress it. Let the observation of the number of daily reports be $y(t_i)$, and the regression result be:

$$\hat{G}(t; \hat{\boldsymbol{\theta}}) = \sum_{n=1}^3 \hat{A}_n \exp\left[-\left(\frac{t-\hat{B}_n}{\hat{C}_n}\right)^2\right]$$

Then its loss function is:

$$L(\hat{\boldsymbol{\theta}}) = \sum_{i=1}^{359} [y(t_i) - \hat{G}(t_i; \hat{\boldsymbol{\theta}})]^2 \quad (2)$$

We take $\arg \min_{\hat{\boldsymbol{\theta}}} L(\hat{\boldsymbol{\theta}})$ as the regression result, and the corresponding $\hat{G}(t; \hat{\boldsymbol{\theta}})$ is the predicted trend.

● Report number prediction model based on non-homogeneous Poisson process

The Poisson distribution describes the probability of a certain number of events occurring

over a period of time under the condition that the event occurrence rate is constant, and thus can describe the probability of a certain number of reports uploaded in a day. We assume that the number of reports each day obeys a Poisson distribution, then these Poisson distributions form a non-homogeneous Poisson process in time, i.e., a Poisson process in which the arrival intensity varies with time.

The number of reports on the day t is a random process $X(t)$ that obeys a non-homogeneous Poisson process with arrival intensity $\lambda(t)$. The probability that the number of reports on the day t is:

$$P_k(t) = P\{X(t) = k\} = \frac{\lambda(t)^k}{k!} e^{-\lambda(t)} \quad (3)$$

where the meaning of $\lambda(t)$ is the mean value $m_X(t) = E[X(t)]$ of the number of reports on the day t . However, the mean value function cannot be derived from the available statistics. Therefore, we retreat and use the trend prediction result $\hat{G}(t)$ from the previous Gaussian regression to approximate the mean function of the number of reports $m_X(t)$ instead, such that $\lambda(t) = \hat{G}(t)$.

Thereby, the random fluctuations of the reported number can be well portrayed by introducing a non-homogeneous Poisson process.

● Random process correction based on popularity relaxation function

Since the random process $X(t)$ mentioned above is not a completely independent incremental process in practice, the size of its random fluctuation is affected by its popularity. By analogy with the life cycle of online public opinion^[5], this paper divides the life cycle of Wordle's popularity trend. Considering that the data are counted from January 7, the initial "formation" stage is omitted.

- i. **Explosion Period:** the number of players surges due to the growth of popularity and the sharing of results on social platforms. The index of attention and action of Twitter users on this type of topic soars to its peak and fluctuates with greater uncertainty in its scope.
- ii. **Fading Period:** the popularity is time-sensitive as the novelty of the game has passed for players. And, players' desire to share their achievements decreases, but it does not mean that the number of players decreases at this time. Nevertheless, the overall fluctuation of popularity is lower compared to the burst period.
- iii. **Dormant Period:** the popularity leveled off, there were still many loyal players to the game, and the conversation remained. In general, the ups and downs do not change much.

In this paper, we observe that the random fluctuations in the number of reports do not exactly obey the Poisson distribution when the game is popular, and the fluctuations are significantly larger. As the game's popularity waned, so did the fluctuations. Therefore, we introduce a popularity relaxation function to modify the random process model.

As mentioned in Section 5.1, the boundary of the popularity relaxation phenomenon can be approximately reduced to a linear boundary, so we define the popularity relaxation function as $f(k) = l \cdot k + m$, with k being the number of reports and l, m being constants.

The modified random process arrives at the intensity function as:

$$\overset{\circ}{\lambda}_k(t) = \lambda(t) \cdot f(k) \quad (4)$$

Therefore, the probability that the number of reports is k on the day t after the correction of equation (3) is:

$$\mathring{P}_k(t) = \frac{\mathring{\lambda}_k(t)^k}{k!} e^{-\mathring{\lambda}_k(t)} \tag{5}$$

We can calculate the prediction interval $[\lambda(t) - lb, \lambda(t) + rb]$ for the number of reports on the day t with a certain confidence level β based on $\mathring{P}_k(t)$, as shown in equation (6).

$$\begin{cases} lb = \arg \min_N \left| \frac{\beta}{2} - \sum_{n=1}^N \mathring{P}_{\lambda(t)-n}(t) \right| \\ rb = \arg \min_M \left| \frac{\beta}{2} - \sum_{m=1}^M \mathring{P}_{\lambda(t)+m}(t) \right| \end{cases} \tag{6}$$

5.2.2 Establishing prediction intervals for future reported number results

Based on a trend forecasting model with Gaussian regression, we predict the long-term trend in the number of reports. The regression results in $RMSE = 6034.7$, $R^2 = 0.99553$. This indicates that the regression results are more explanatory of the trend in the number of reports, and the root mean square error is about 2 orders of magnitude smaller than the number of reports. The regression coefficients are shown in Table 2, and we will show the specific trend prediction effects together with the prediction intervals.

Table 2: Regression coefficient of the trend prediction model

$\hat{\theta}$					
A_1	1.57e+05	B_1	33.01	C_1	30.79
A_2	9.69e+04	B_2	48.2	C_2	75.31
A_3	4.846e+04	B_3	5.864	C_3	386.7

Then, by modifying the report number prediction model of the non-homogeneous Poisson process, the report number prediction interval of 75% confidence is obtained. Figure 5 shows the model's present description of the reported number results with future predictions, and the horizontal coordinates are the number of days to January 7, 2022.

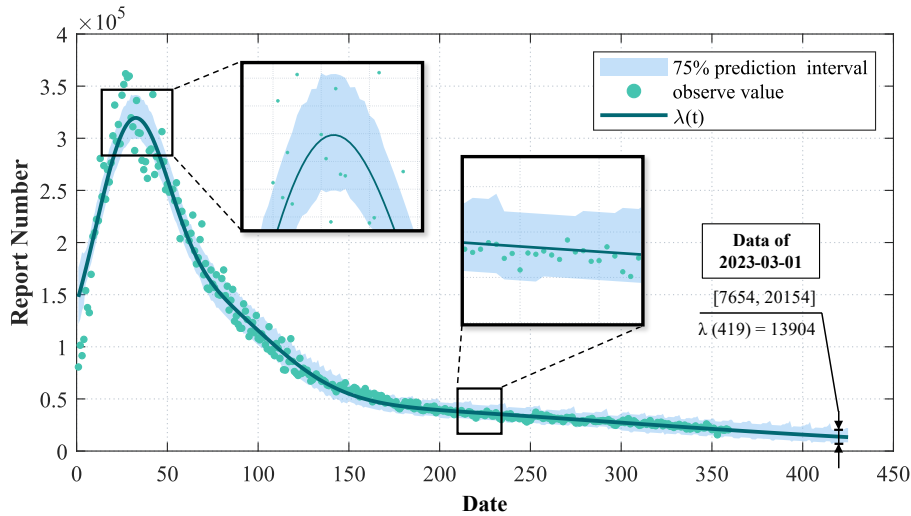


Figure 5: Report number trends and 75% confidence level intervals predicting

From the figure above, we can find that our model can predict the long-term trend of the number of reports more accurately, and can also roughly estimate the random fluctuation interval of the number of reports per day. It is worth noting that the prediction interval is a good reflection of the timeliness of the heat, and the number of users shows a surge and significant fluctuation when the "Date" is about 40, as the number of users is in the "Explosion Period". When the "Date" is about 220, the number of users is in the "Dormant Period", and the number and fluctuation are relatively small and tend to be stable.

We predict that the number of reported results converges to **13,904** on March 1, 2023 (when the value of the horizontal coordinate is 419 in Figure 5). The results of the prediction intervals at different confidence levels are shown in Table 3.

Table 3: Prediction interval for the number of reports (on March 1st)

Confidence level	Left border of the prediction interval	Right border of the prediction interval
75%	7654	20154
85%	5434	23657

In general, the overall change pattern of the number of reports is determined by the social attributes and social laws of the game. And this changing pattern shows a clear trend, so a better prediction effect can be obtained through the regression model.

Based on the overall change trend, the number of reports also has a certain degree of random fluctuation. This stochastic fluctuation has the statistical characteristic of changing with time and heat. Therefore, the stochastic process can be applied to describe it.

5.3 Analysis of Game Mode Selection

5.3.1 Analysis of word attributes

First, we analyze the possible word attributes involved, which can be mined from the existing dataset in 3 main aspects: letter frequency, letter position and common word roots, etc.

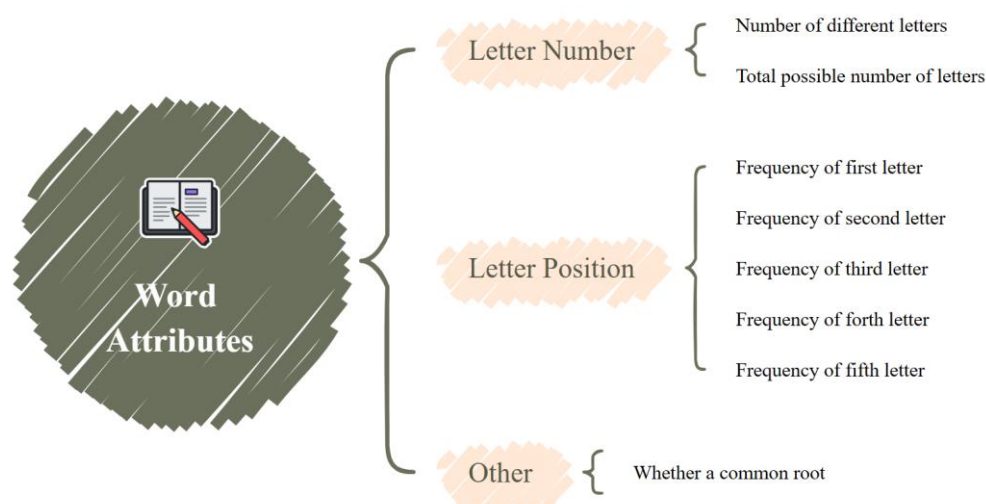


Figure 6: Analysis of word attributes

- **Number of different letters:** It is expressed as the number of different letters in a word. Statistically, the range of values is 3 to 5. For example, the value of this attribute for the word "happy" is 4. This attribute reflects the internal variability of the word.

- **Total possible number of letters:** It represents the sum of the frequencies of all letters in a word. Assuming that the frequency of the letter " α " in the 359-word result data is f_α , the value of this attribute for the word "happy" is $f_h + f_a + 2f_p + f_y$. This attribute is an indication of the overall usage tendency of the word.
- **Frequency of first letter:** It indicates the frequency of the first letter of a word. For example, in 359 data items, the total number of the first letter of each word is 359, and the percentage of words with the first letter "h" is p_{h1} , then the value of this attribute for the word "happy" is $p_{h1}/359$. This attribute reflects the local position tendency of the word.
- **Frequency of second letter:** It indicates the frequency of the second letter of a word. For example, in 359 data items, the total number of the second letter of each word is 359, and the percentage of words with the second letter "a" is p_{a2} . The value of this attribute for the word "happy" is $p_{a2}/359$. This attribute reflects the local position tendency of the word.
- **Frequency of third letter:** It indicates the frequency of the third letter of a word. For example, in 359 data items, the total number of the third letter of each word is 359, and the percentage of words with the third letter "p" is p_{p3} , then the value of this attribute for the word "happy" is $p_{p3}/359$. This attribute reflects the local position tendency of the word.
- **Frequency of fourth letter:** It indicates the frequency of the fourth letter of a word. For example, in 359 data items, the total number of the fourth letter of each word is 359, and the percentage of words with the fourth letter "p" is p_{p4} . The value of this attribute for the word "happy" is $p_{p4}/359$. This attribute reflects the local position tendency of the word.
- **Frequency of fifth letter:** It indicates the frequency of the fifth letter of a word. For example, in 359 data items, the total number of the fifth letter of each word is also 359, and the percentage of words with the fifth letter "y" is p_{y5} , then the value of this attribute for the word "happy" is $p_{y5}/359$. This attribute reflects the local position tendency of the word.
- **Whether a common root:** It indicates whether a word has common roots within it. For example, if the word "manly" contains the root "-ly", then the word has a value of 1; otherwise, it has a value of 0. This property reflects the local regularity of the word.

5.3.2 Analysis of the influence of word attributes on pattern selection

We wanted to find out whether the 8 attributes of the word listed in the previous section affected the user's choice of game mode. So, for each attribute, a scatter plot comparing the relationship between daily word attributes and Hard Mode choices was plotted in Figure 7.

In the figure below, the horizontal coordinates of each scatter plot are the percentage of Hard Mode choices (in %). It can be noticed that the individual attributes do not have a strong correlation with the percentage of Hard Mode. **The presented word attributes do not affect the proportion of reported data for the Hard Mode.** We believe that the reason for this phenomenon is that players are not informed about the solution word in advance. That is, the word attributes are unknown before most players choose the game mode, so the player's choice is not highly correlated with it.

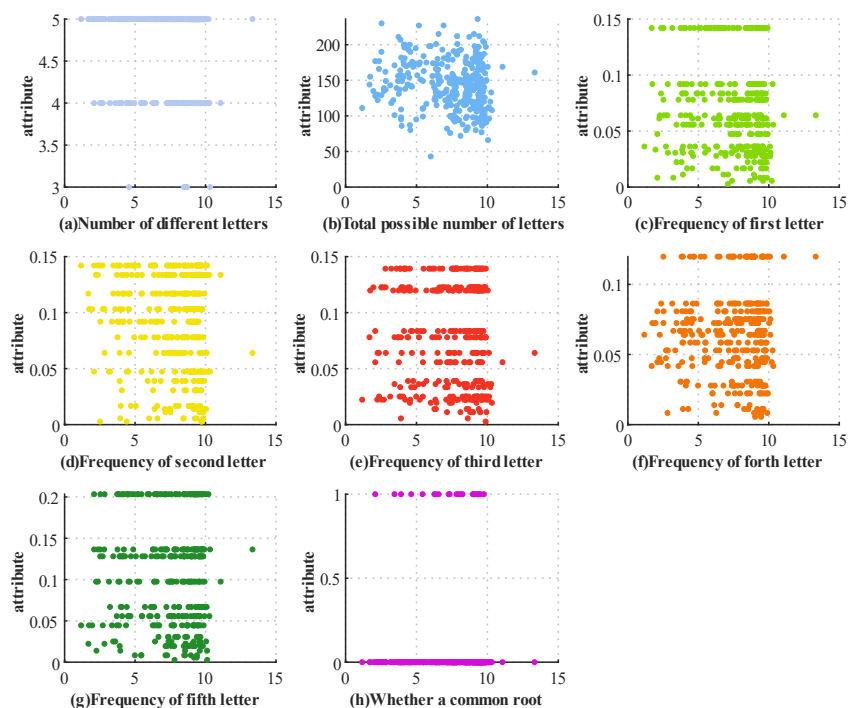


Figure 7: Correlation between the proportion of Hard Mode choices and word attributes

So what are the main factors associated with the Hard Mode selection ratio?

Table 4 provides a further visualization of the word content and date of some of the Hard Mode percentages. We can find that there are no obvious word attribute characteristics in the top ten reports and the bottom ten reports in the proportion ranking. However, the data with a high percentage of Hard Mode choices tend to appear in the last three months of 2022, while the data with a low proportion tend to appear in the first month of 2022 with a high correlation with time.

Table 4: The percentage of scores reported in Hard Mode

Top 10			Bottom 10		
Date	Word	Percentage	Date	Word	Percentage
2022/11/1	piney	13.33%	2022/1/16	solar	2.36%
2022/9/16	parer	11.07%	2022/1/14	tangy	2.35%
2022/11/30	study	10.66%	2022/1/15	panic	2.26%
2022/10/23	mummy	10.32%	2022/1/12	favor	2.23%
2022/10/15	catch	10.27%	2022/1/10	query	2.09%
2022/12/26	judge	10.21%	2022/1/9	gorge	2.09%
2022/10/30	waltz	10.12%	2022/1/11	drink	1.96%
2022/10/12	ionic	10.11%	2022/1/8	crank	1.74%
2022/10/29	libel	10.08%	2022/1/7	slump	1.69%
2022/12/25	extra	10.04%	2022/2/13	robin	1.17%

To further explore this pattern, we visualized the change of the percentage of Hard Mode choice over time, as shown in Figure 8(a), the color of the heat map fades from blue to green as time increases, and there are only few dates with outlier fluctuations, i.e., the percentage of

Hard Mode choice shows an increasing trend, which can also be seen in the scatter plot of Figure 8(b). There is a very clear trend that the percentage of Hard Mode selection increases with time, not only incrementally, but also the growth rate is gradually slowing down.

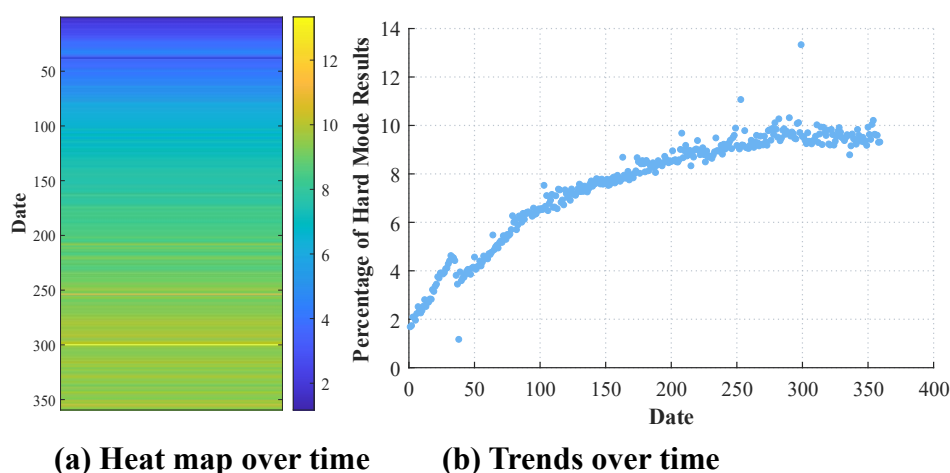


Figure 8: Description of the Hard Mode selection percentage

We believe that, on the one hand, this is because as the time spent playing the game increases, the user's ability to perform in the game is also sufficiently improved, so some users who love challenges will gradually tend to choose a higher difficulty mode. On the other hand, many users may play with a more casual mindset and do not consider increasing the difficulty of the game even though their ability has increased. In other words, players' confidence level in their performance ability and their playing mentality may be the main reason for whether they choose the Hard Mode, rather than the attributes of the words.

6 Task 2: A Prediction Model for The Distribution of The Reported Results

To predict the distribution of future report results, we first extracted and constructed the data features. Then, we build a BP neural network model to take 7 data features as input and output the distribution of 7 guessed word results. Finally, the Bagging algorithm was adopted to integrate multiple BP neural networks to derive the final prediction results through a hard voting mechanism to reduce the generalization error of the prediction results.

6.1 Building the BP Neural Network-based Prediction Model for the distribution of word-guessing results

Considering that the result distribution of the number of word guesses shared on Twitter is likely to be different from that of the entire player population, and is related to many factors that are difficult to quantify and count, including the players' mindset and the familiarity of most players with the word of the day. Therefore, mechanistic modeling of the word-guessing outcome distribution may be difficult and unrealistic, and we decided to model the data for this problem. Of course, the approach implicitly assumes the condition that the historical data is a good representation of all possible questions and player answers for wordle, which is **the biggest uncertainty of our model**.

Given the solution word and the corresponding date, we have access to the properties of the words and all the information that can be predicted based on the time course. The total number of historical data is 359, which is likely to cause underfitting problems for deep learning algorithms, while the size of the data is acceptable for BP neural networks. Therefore, we decided to build a prediction model for the distribution of reports based on BP neural networks.

6.1.1 Extraction and construction of data features

Before building the neural network specifically, we first extract and construct the data features. As mentioned earlier, we want to predict the distribution of word-guessing results based on the solution words and the corresponding date. The sources of data features at this point include the words themselves and the predictable time course.

The information that can be obtained from the words themselves has been explored in Section 5.3.1, i.e., word attributes. We classified word attributes into 3 categories, including letter frequency, letter position, and common root words. Among them, our statistics for the attribute of **whether it contains common word roots** resulted in unbalanced data. As a Boolean value, only about 20 out of 359 data for this attribute had a true value. Therefore, this attribute is of no value for the training of the neural network, and we do not keep this feature. In addition, the feature **Total possible number of letters**, which has some repetitiveness, is also removed to compress the data dimension.

Finally, we also selected **the percentage of people who chose the Hard Mode** as a feature because the difficult choice of the game also affected the distribution of the number of guesses. As analyzed in Section 5.3.2, this feature has a clear trend and small fluctuations over time, so the feature is predictable for a determined future date. The value of this characteristic in the second half of 2022 mostly fluctuates around 9.5% and the variation is not significant anymore. So we make some simplifications and assume that it will remain this way for some time to come, without making more precise predictions. We take the average value of the difficult mode selection ratio for the latter 59 days as the forecast value.

6.1.2 Construction of BP neural network

For the report result distribution prediction problem, 7 data features such as **Number of different letters** constitute the input space, while 7 reported result distributions constitute the output space. The mathematical essence of the problem is to fit a mapping from a 7-dimensional input space to a 7-dimensional output space. A review of the literature ^[6] shows that for fitting such finite-dimensional space mappings, building neural networks usually requires only one hidden layer. Therefore, the BP neural network in this model has a total of three layers, which are the input layer, the hidden layer, and the output layer.

Since the input space and output space are both 7-dimensional, the number of neurons in the input layer N_i and the number of neurons in the output layer N_o are both 7. Referring to the empirical formula proposed in the literature ^[7], we set the number of hidden layer neurons N_h that satisfy equation (7).

$$N_h = \frac{N_s}{\alpha(N_i + N_o)}, \alpha = [2, 10] \quad (7)$$

where N_s is the size of the training set and α is a constant.

At this point, we have completed the construction of the BP neural network, the structure

of which is shown in Figure 9.

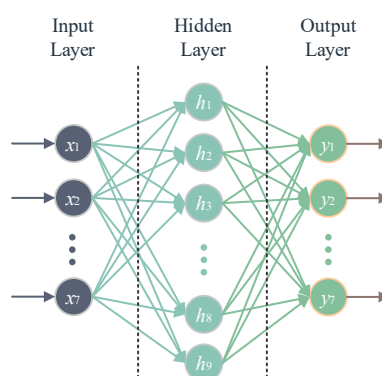


Figure 9: Structure of BP neural network

6.1.3 Bagging-based prediction model with integrated BP neural network

In the training of the BP neural network, we randomly selected the training set according to 85%. After several tests, we found that the neural network obtained from each training always has unacceptable errors at very few samples in the test set. Moreover, the repetition rate of samples with large prediction errors was not high for different neural networks. Therefore, we decided to adopt the Bagging algorithm to integrate multiple BP neural networks. Then, the final results are derived through a hard voting mechanism, which has the effect of reducing the generalization error of the prediction results.

In the integration algorithm, we obtain m different sub-training sets by randomly sampling 85% of the overall data m times, and each time the percentage of sampling is also 85%. Then, m neural networks are obtained based on these sub-training sets trained with the same parameters. Next, these neural networks are used to predict the distribution of word guesses. All neural networks output their respective predictions directly and the final prediction is obtained by a minority-majority voting mechanism [8]. The flow of this integrated algorithm is shown in Figure 10.

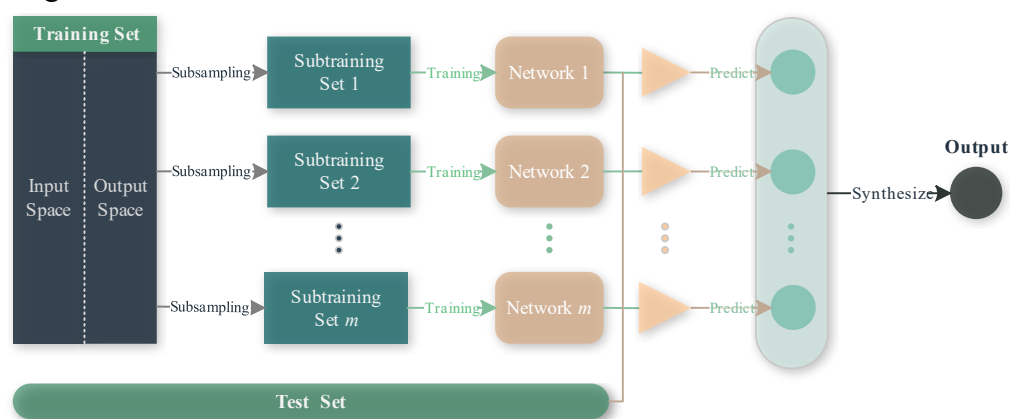


Figure 10: Schematic diagram of integrated BP neural network

To avoid the influence of accidental errors on voting results, $m > 50$ is used for training.

6.2 Analysis of Uncertainties Affecting the Model

The dataset we have and the selected data features are perhaps not well representative of Wordle's questioning and players' answering, which would lead to poor generalization of our model.

In addition, the size of the dataset used for training, although acceptable, is still relatively limited. In the process of randomly selecting sub-training sets by the integration algorithm, the proportion of identical samples among the sub-training sets may be quite high. This can make the homogeneity between individual neural networks too high to function as an integration and voting mechanism.

6.3 Analysis of the Results of the Prediction Model

First, we arrange the data sets in descending order by time and select the top 85% of the data as the total training set. The remaining 15% of the data will be used as the test set to verify the model effect. Then, 85% of the total training set is randomly selected as the sub-training set each time, and m neural networks are trained. Finally, these m neural networks are integrated and the prediction effect is tested with the test set.

After several attempts, we found that we were able to achieve better prediction results when $N_h = 9$, $m = 100$. In the test set, the model's $RMSE = 3.6975$ and the correlation coefficient is $R = 0.8869$. As shown in Figure 11, the model has different prediction effects on the distribution of the seven guess word results. Among them, the prediction result of 3 tries has the largest error, but the mean square error still does not exceed 5%.

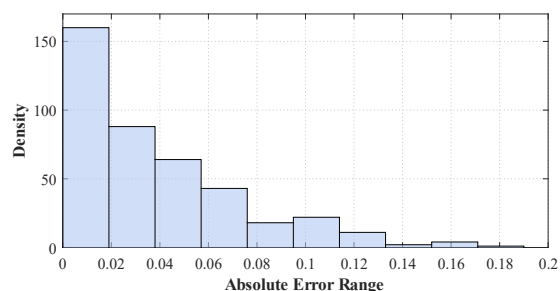
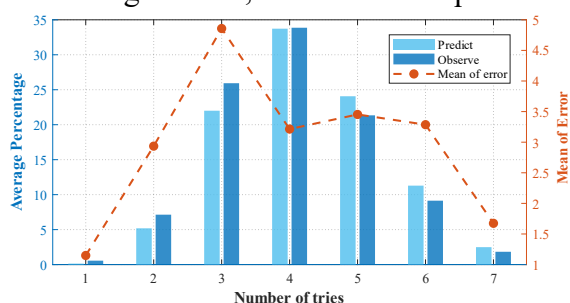


Figure 11: Mean Comparison and Mean Squared Distribution

Figure 12: Absolute error distribution

The distribution of all errors was also counted. As shown in Figure 12. Although errors are inevitable, most of them are maintained within 6%. Among them, the absolute error within 2% accounts for about 38%, and the absolute error within 5% reaches more than 80%. Therefore, we have more than 80% confidence that the absolute error of the prediction result does not exceed 5%.

Finally, we predict the distribution of the number of guesses for the word **ERRIE on March 1, 2023**, as shown in Table 5.

Table 5: Prediction of the distribution of results for the word **ERRIE**

1 try	2 tries	3 tries	4 tries	5 tries	6 tries	(X)
0%	1%	6%	25%	31%	25%	13%

7 Task 3: Word Difficulty Classification Model

To classify the solution words reasonably, we first classified the difficulty based on the K-Means clustering algorithm. Then, we explored the association between word attributes and difficulty classification based on Pearson correlation coefficients, and constructed a word difficulty classification model. Finally, the new words can be classified according to this correlation.

7.1 The Establishment of Word Difficulty Classification

7.1.1 Word difficulty induction model based on K-Means clustering

Before categorizing solution words by difficulty, we need to define the difficulty first. To make the model result more similar to the user's game experience, we decided to define a difficulty division based on the user's guess count distribution. It should be noted that this is only an indication of difficulty, not the reason for determining the difficulty of solution words.

According to our definition of difficulty, the distribution of guesses reflects the difficulty of the word. Therefore, the first step to classifying vocabulary difficulty is to summarize the distribution of guessing times into certain categories. The essence of this problem is to explore the homogeneity and difference between the frequency distribution of historical guessing words. From a mathematical point of view, the homogeneity and difference of data can be described by distance, and the data can be grouped by distance. Therefore, we decided to use the K-Means algorithm to summarize the difficulty of the word.

First, a higher percentage of completed guesses indicates a higher difficulty of the puzzle. Therefore, the difference in difficulty in the distribution of the number of guesses is an absolute difference that can be described by the Euclidean distance. Then the difference between the guessing result \mathbf{A} and the guessing result \mathbf{B} is reflected in the Euclidean distance of the distribution vector of the two, as shown in equation (8).

$$D_E(\mathbf{A}, \mathbf{B}) = \sqrt{\sum_{i=1}^7 (A_i - B_i)^2} \quad (8)$$

Then, we need to determine how many levels of difficulty to classify, i.e., determine the number k of centers of mass for the K-Means clustering algorithm. We will find the best result k by multiple attempts. The goal is to make the difficulty classification without duplication or omission.

Next, k samples are randomly selected as initial clustering centers in the whole set of samples $x = \{x_1, x_2 \dots x_{359}\}$, respectively, and noted as:

$$\mu_1^{(0)}, \mu_2^{(0)}, \dots, \mu_k^{(0)}$$

Since we have decided to use the Euclidean distance to measure the differences between samples, we take equation (9) as the optimization objective of the algorithm, i.e., to minimize the Euclidean distance of all samples to their cluster centers.

$$J(c, \mu) = \min \sum_{i=1}^{359} \|x_i - \mu_{c_i}\|^2 = \min \sum_{i=1}^{359} D_E(x_i, \mu_{c_i}) \quad (9)$$

Where c_i is the cluster to which sample x_i belongs.

The distance between each sample point and each centroid is calculated iteratively. At the same time, the samples are allocated to the cluster with the smallest distance from the corresponding centroid:

$$c_i^{(t)} = \arg \min_m \|x_i - \mu_m^{(t)}\|^2$$

At the end of each iteration, the average distance of sample points in each cluster is calculated as the centroid of the next iteration:

$$\mu_n^{(t+1)} = \frac{1}{b} \sum_{i:c_i^{(t)}=n}^b x_i$$

At the same time, the objective function value $J^{(t)}$ of this iteration is compared with that of $J^{(t-1)}$ of the previous iteration. The objective function has converged, and then the clustering ends. The final c is the difficulty level of each sample, and μ corresponds to the typical value of each difficulty level. In addition, the distance matrix $dist$ between each sample and its clustering center is also obtained.

7.1.2 Correlation analysis of word attributes and difficulty ratings based on Pearson coefficients

After obtaining the difficulty grading results, we used Pearson correlation coefficients to analyze the association between each attribute of the word and the difficulty grading. This is because it is more difficult to confirm the causal relationship between attributes and difficulty, but correlation can be used instead when classifying.

First, we selected the frequency of occurrence of letters in five positions, such as Frequency of first letter (F1), Frequency of second letter (F2) mentioned in Section 5.3.1, as the word attributes to be analyzed. Based on this, we further counted the number of word classes (WCN) and the number of vowels (VN) contained in each solution word in the dataset. So far, we have obtained 7 word attributes $a_{i1}, a_{i2}, \dots, a_{i7}$ to be analyzed.

Then, we select the samples closest to the respective clustering centers as typical samples from the k difficulty levels. The k sets of representative typical attribute vectors $S_j = [a_{j1} \ a_{j2} \ \dots \ a_{j7}]$ are obtained, where $i = 1, 2, \dots, 359$, denotes the i th sample, and $j = 1, 2, \dots, k$, denotes the typical sample serial number of the j th difficulty grading.

For the t th attribute of the i th sample, we calculate the Euclidean distance between that attribute of the sample and the corresponding attribute of the k typical samples, respectively.

$$\widehat{dist}_{ij}^{(t)} = D_E(a_{it}, a_{jt})$$

We let $W_i^{(t)} = [dist_{i1} \ dist_{i2} \ \dots \ dist_{ik}]$, denoted as the attribute distance vector. And let $\widehat{W}_i^{(t)} = [\widehat{dist}_{i1}^{(t)} \ \widehat{dist}_{i2}^{(t)} \ \dots \ \widehat{dist}_{ik}^{(t)}]$, denote the clustering distance vector.

Next, we calculated the Pearson correlation coefficients for the attribute distance vector $\widehat{W}_i^{(t)}$ and the cluster distance vector $W_i^{(t)}$, and obtained the correlation coefficients:

$$\rho_i^{(t)} = \frac{\text{cov}(\widehat{W}_i^{(t)}, W_i^{(t)})}{\sigma_{\widehat{W}_i^{(t)}} \sigma_{W_i^{(t)}}$$

Where $\sigma_{\widehat{W}_i^{(t)}}$, $\sigma_{W_i^{(t)}}$ denote the variances of $\widehat{W}_i^{(t)}$ and $W_i^{(t)}$, respectively.

Finally, we counted the mean value of the correlation coefficient $m_\rho^{(t)}$ for each attribute for all samples. We also set the boundary $M = 0.6$ when $m_\rho^{(t)} > M$ considering that the t th word attribute and the difficulty are correlated. Finally, we filtered the attributes to obtain a vector of attributes $attributes_i = [a_{i1} \ a_{i2} \ \dots \ a_{i7}]$.

7.1.3 Word difficulty discrimination based on Euclidean distance

For the future solution word, we can judge its difficulty by calculating its similarity to

each typical sample. Since we built a prediction model for the distribution of the number of guesses for a given solution word at a future date in Section 6.2, we have two bases for judging the difficulty of words. One is based on the predicted distribution of the number of guess words, and the other is based on the attribute vector of the solution word *attributes*.

1. Difficulty determination method based on the predicted distribution of reports:

The prediction model in Section 6.2 is used to predict the distribution of the number of guesses of words to obtain the prediction \hat{x} . Then the difficulty \hat{c} is judged based on the Euclidean distance and the assignment principle of the K-Means algorithm.

$$\hat{c} = \arg \min_i \|\hat{x} - \mu_i\|^2$$

where μ_i is the center of mass of the i -th difficulty, $i = 1, 2, \dots, k$.

2. Difficulty determination method based on attribute vectors:

The attribute vector $attributes_x$ of the solution word is counted, and then its difficulty \hat{c} is judged based on the Euclidean distance and the assignment principle of the K-Means algorithm.

$$\hat{c} = \arg \min_j \|attributes_x - attributes_j\|^2$$

where $attributes_j$ is the typical sample attribute vector for the j th difficulty, $j = 1, 2, \dots, k$.

7.2 Analysis of Word Difficulty Classification Results

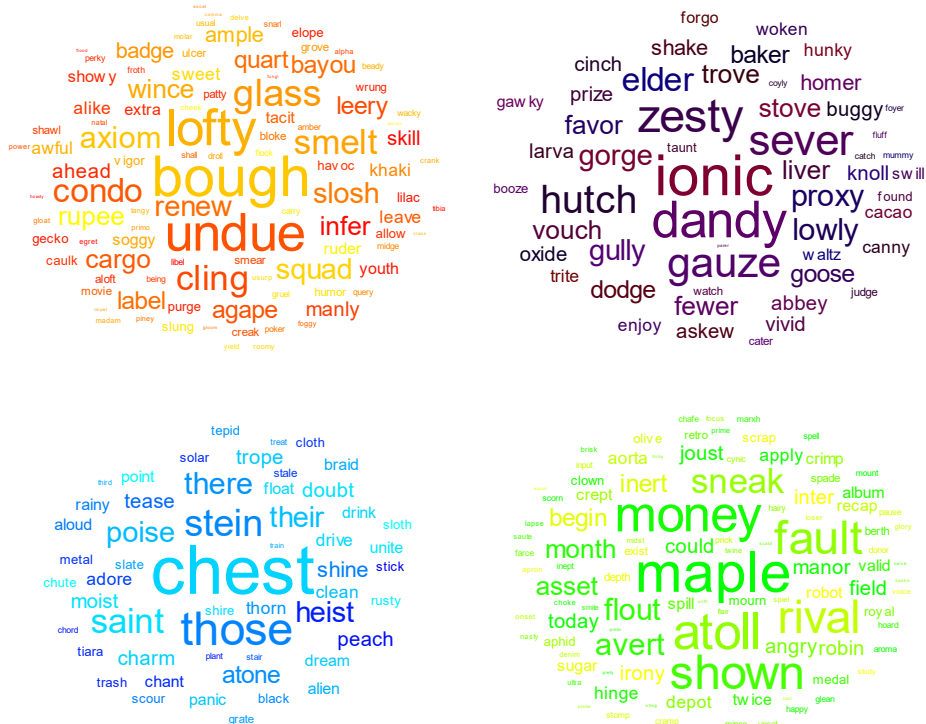


Figure 13: Difficulty classification result word cloud

The results were obtained by K-Means clustering in four categories of difficulty, which were classified as Easy, Normal, Hard, and Master in order of difficulty from small to large, and a typical word was selected for visual analysis, as shown in Figure 14.

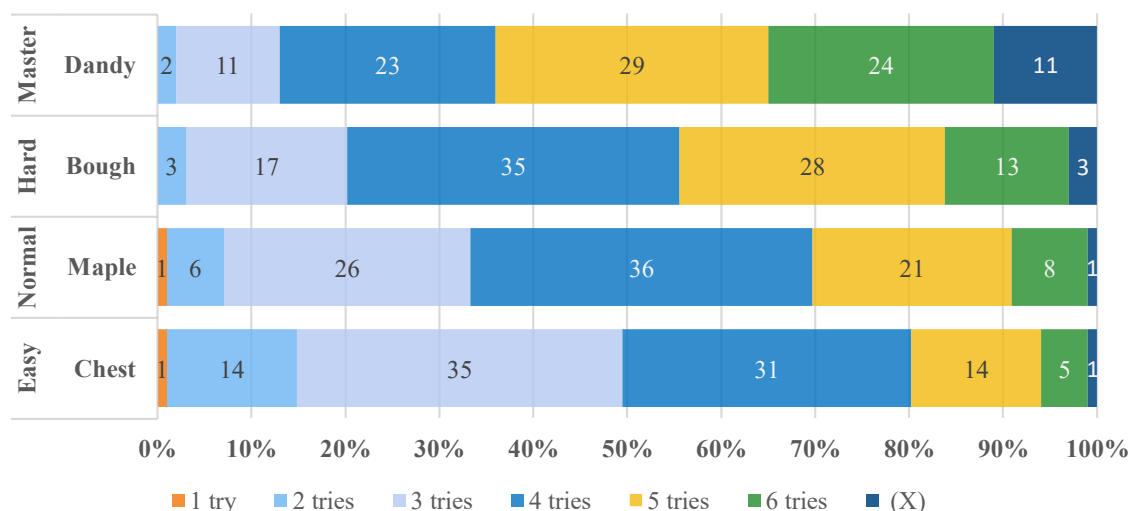


Figure 14: Word difficulty classification results bar chart

As we can see from the above graph, when the puzzle is easier, the distribution of reports tends to guess the solved word less often, and vice versa. For example, the distribution of reports for the word "Chest" is mainly in 2tries, 3tries, and 4tries, and the distribution is smaller in 6tries and 7tries.

Table 6: Correlation coefficient of word attributes and difficulty

m_ρ						
F1	F2	F3	F4	F5	WCN	VN
0.637634	0.813854	0.336972	0.417339	0.311806	0.81528	0.8989

After calculating the Pearson correlation coefficients between word attributes and difficulty ratings, we obtain the results in Table 6. The above table shows that some attributes have a strong correlation with difficulty. We take the word attributes with m_ρ greater than 0.6 as the difficulty classification attributes and form the attribute vector $attributes_i = [a_{i1} a_{i2} a_{i6} a_{i7}]$.

Finally, we calculate the distance from the attribute vector of the word EERIR to the attribute vector of the j th difficulty typical sample $attributes_j$, which are R1=4.2385, R2=3.0021, R3=2.1823, R4=0.6250, and the closest distance to "Dandy". Thus, the difficulty of the word EERIR was obtained as "Master".

8 Task 4: Other Interesting Features

In exploring the statistical properties of the number of reports, we found that the density of the distribution of the number of reports over 359 days showed a similar pattern to its trend over time. Then, we fitted the distribution of the number of reports. As shown in Figure 15, a good result can be achieved by fitting the number of reports using a log-normal distribution. Referring to Figure 16, it can be found that the distribution obtained from the fit shows a fairly high similarity to the change in the number of reports over time.

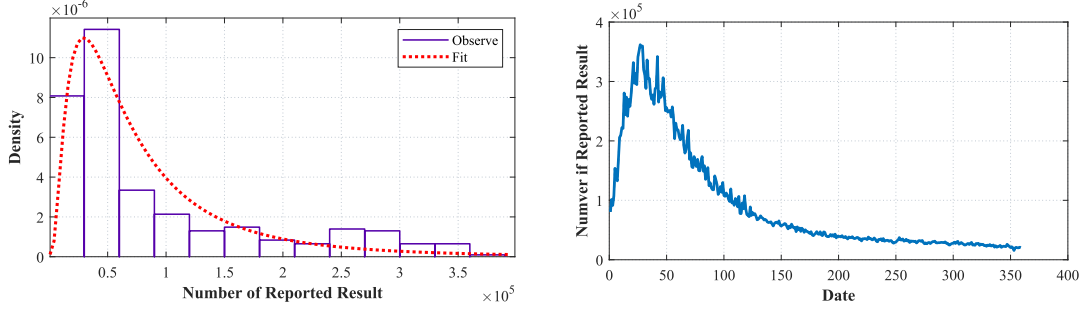


Figure 15: Fitting to the distribution of report number **Figure 16: Change in report number over time**

However, we note that the distribution of the number of reports is not uniform in time, and this distribution does not take into account the time factor. So this fit does not provide much information for predicting the number of reports. This led us not to investigate the phenomenon in more depth. However, we speculate that there may be some interesting statistical properties under this phenomenon.

In addition, when we explored the distribution of users' reported results, we found that the percentage fluctuation of 3 attempts to complete the game was the largest. As shown in Figure 17, we believe that this phenomenon is strongly related to the low prediction accuracy of our report distribution prediction model for 3-attempt game completion.

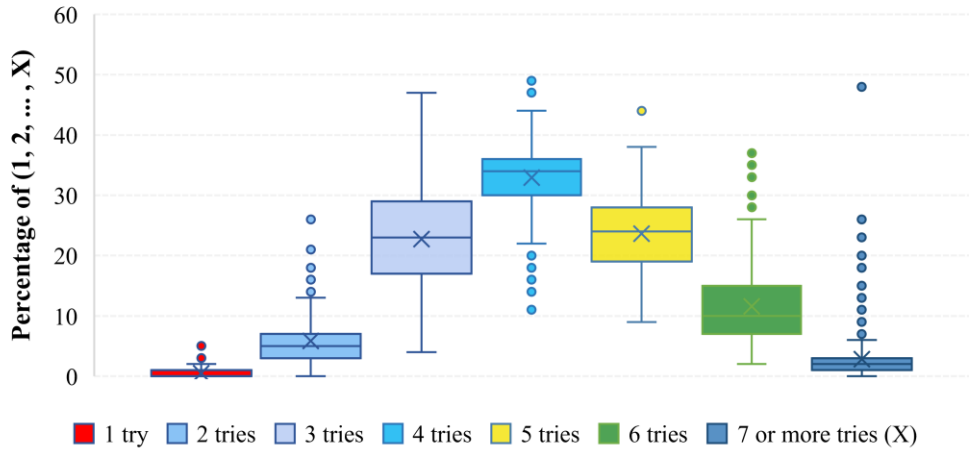


Figure 17: Report distribution box chart

9 Sensitivity Analysis

For the prediction model of the number of reports in Task 1, we fine-tune the intercept k of the popularity relaxation function by multiplying it by a certain scale factor δ to observe the sensitivity of the model. The average change in the corresponding prediction interval when the confidence level is 75% is then calculated as:

$$m_\delta = \frac{\sum_{n=1}^N (lb_n + rb_n - \widehat{lb}_n - \widehat{rb}_n)}{2N}$$

where N is the length of the sequence. Then, we take $\delta = 1.1, 1.5, 1.8$ separately for

observation. When $\delta = 1.1$, the average variation of the prediction interval reached about 8.3%. When $\delta = 1.5$, the average variation of the prediction interval reaches about 32.71%. When $\delta = 1.8$, the average variation of the prediction interval reaches about 78.2%, and the left boundary $\delta = 1.8$ of the prediction interval is close to 0 at this time.

This indicates that the model we have developed is somewhat robust, but is not very applicable to situations where there are large stochastic fluctuations even in steady periods.

10 Model Evaluation and Further Discussion

10.1 Strengths

- We decompose the time series into trend signs and stochastic fluctuations and then make separate predictions. The model built according to this has a good interpretation of the time series patterns.
- We do not define the difficulty of words by ourselves, but generalize the difficulty based on the results of user play. Therefore, the difficulty rating we obtained is more relevant to the user's play experience.
- We applied the integration algorithm to weaken the generalization error caused by chance due to neural network training. Eventually, better prediction accuracy than individual neural networks is achieved.

10.2 Weaknesses

- For the popular phases of the game, the prediction interval given by our reported volume prediction model is much wider compared to other phases. This leads to the possibility that our model's prediction results in such phases may not be very informative.
- Our model for predicting the distribution of reported outcomes relies heavily on the representativeness of the dataset. However, for the size of the provided dataset, it is more difficult to ensure the representativeness of the overall sample space.

10.3 Further Discussion

The model we build is based only on the existing data given in the report, but other available information can be further integrated. In the prediction model of the number of players, in addition to using time series information, we can also analyze the attractiveness of the game based on the text information posted by users and obtain players' emotional experience of the game; based on the personal information of users who share the results, we can simulate users as nodes and build an information dissemination model based on social network analysis, which in turn can portray the user behavior that affects the heat of game dissemination and make a more comprehensive popularity prediction.

11 Conclusion

To mine the Wordle report information, we propose a series of novel models to solve the problem of predicting the number of reports and the distribution of results. At the same time, we fully analyze the attributes of solution words and can classify the difficulty of a given word.

The proposed model only depends on the Data File and has good interpretation and rationality.

1. The report number prediction model can combine the long-term trend of quantity change over time with random fluctuations. The results of Gaussian regression on the number of reports describe the trend of the number of reports over time, while the non-homogeneous Poisson process describes the random fluctuation of the number of reports based on the trend. According to the life cycle of heat, we also introduce the popularity relaxation function to modify the random process model.
2. The proportion of players choosing a game mode is less affected by the attributes of words, but is highly correlated with time. The level of confidence players have in their ability to perform and how they play is probably the main reason why they choose Hard Mode.
3. The distribution of the reported results prediction model can combine the word attribute characteristics and the time characteristics that affect the proportion of Hard Mode. We predict the distribution of guessing results based on the BP neural network, and integrate the predicted results of neural networks through the Bagging algorithm to improve the generalization performance of the model.
4. The word difficulty classification model can combine the distribution of report results with the attributes of words. Among them, the distribution of reported results only reflects the difficulty, while the word attributes are the root cause of the different difficulties. We divided the difficulty based on the K-Means algorithm, and explored the correlation between word attributes and difficulty classification based on the Pearson correlation coefficient. Finally, new words can be classified by difficulty based on this correlation.

References

- [1] Q. Zhao, M. A. Erdogdu, H. Y. He, A. Rajaraman, and J. Leskovec, "SEISMIC: A Self-Exciting Point Process Model for Predicting Tweet Popularity," *ACM*, 2015.
- [2] H. W. Shen, D. Wang, C. Song, and A. Barabási, "Modeling and Predicting Popularity Dynamics via Reinforced Poisson Processes," *AAAI Press*, 2014.
- [3] Q. Wu, C. Yang, X. Gao, H. Peng, and G. Chen, "EPAB: Early Pattern Aware Bayesian Model for Social Content Popularity Prediction," in *2018 IEEE International Conference on Data Mining (ICDM)*, 2018.
- [4] Z. Zhen, S. Shao, X. Gao, G. Chen, "Social Circle and Attention Based Information Popularity Prediction," *Journal of Computer Science*, 2021.
- [5] J. Feng, "Research on social network event heat prediction based on text analysis," *Harbin Institute of Technology*, 2018.
- [6] G. Ian, B. Yoshua, C. Aaron, "Deep Learning: Adaptive Computation and Machine Learning series," *The MIT Press*, 2016.
- [7] C. D. Ari, "How to choose the number of hidden layers and nodes in a feedforward neural network? " *StackExchange*, <https://stats.stackexchange.com/questions/181/how-to-choose-the-number-of-hidden-layers-and-nodes-in-a-feedforward-neural-netw>.
- [8] L. Kai, L. J. Cui, "Diversity and Performance Comparison for Ensemble Learning Algorithms," *Computer Engineering*, 2008.

Letter

Dear Puzzle Editor of the New York Times,

We were happy to dig into the mystery behind Wordle's data through this modeling. Either for the joy of uncovering a puzzle or to spit out a question that is too difficult, the share button that appears immediately after completing Wordle allows players to bring their emotions of the moment to more people. Wordle has become a social currency as people continue to share it.

Based on the reported results data collected on Twitter over the past year, we built three models: a report number prediction model for interval prediction of future report numbers; a report distribution prediction model for predicting the distribution of the number of guesses on future report results; and a word difficulty classification model for classifying the difficulty of solving a given vocabulary. We have obtained some valuable findings that come from comprehensively digging into the deeper information of the reports, helping you to discover the secrets in last year's game data and to develop new puzzles.




First, we built a report number prediction model to describe past or predict future report number intervals to help you more easily track Wordle's popularity trends. We found that the overall pattern of change in the number of reports is determined by the social properties and social patterns of the game, while the random fluctuations in the number of reports have statistical properties that vary over time and in popularity.

We analyzed the properties of the daily solution words, which included features such as the number of letters, their locations and special roots. We preliminarily analyzed whether words influenced players' game mode choices. We concluded that the proportion of Hard Mode choices increased over time and then leveled off, while word attributes did not affect the proportion for Hard Mode. Players' level of confidence in their performance ability and their play mentality may be the main reasons for whether they choose the Hard Mode or not.

Next, we built a prediction model for the distribution of reported results based on Bagging's integrated BP neural network. We can predict the distribution of reported outcomes based on the future date and the solution word on that date. Simply put, this model takes into account both time series and word attributes, and helps you predict the distribution of players' guesses for a word on a given day after you have decided on the puzzle.

Then, based on the distribution of reported results, we classify the word difficulty based on the K-Means algorithm and explore the association between word attributes and difficulty by the Pearson correlation coefficient. Thus, we can calculate the difficulty of the solution word from the attributes of a newly given word based on the difficulty classification model.

Finally, we list the following results, which may provide some reference for you:

-  The interval for the number of reported results on March 1, 2023 is [7654, 20154].
-  The distribution of the results of EERIR on March 1, 2023 is (0, 1, 6, 25, 31, 25, 13).
-  The difficulty of EERIE words is "Master" (Hardest).

Thanks for taking the time out of your busy schedule to read my letter. Hope our advice can help.

Yours Sincerely,
Team # 2311717