Problem Chosen	2023 MCM/ICM Summary Sheet	Team Control Number
C		# 2305804

Wordle killer: The counterattack of information theory

Summary

In the past year, a word game named Wordle has gained popularity among people, leading to a frenzy of discussions. In this paper, we develop and analyze some models to discover the mathematical and statistical regularity behind this attractive game. Through our analyses, we hope to unveil the mechanics of the game.

For problem 1, we propose ARIMA and **FB-Prophet** models, to predict the number of reported results for a given time point, and to explore the factors that influence a player's choice between game modes. We use the indicators to display that FB-Prophet is more reliable than ARIMA. Therefore, we conclude the prediction interval on March 1, 2023, is **[10937.431632, 14937.431632]** by using the prophet. The superior performance of FB-Prophet is attributed to its ability to account for uncertain seasonality in the data. What's more, we think that the reason why the number of reported results varies daily is also because of the seasonality hidden in the data. In the next stage, we extract linguistic and statistical features from the solution words, which are the **number of vowels, number of repeated letters, number of unique letter, Orthographic neighbor and word frequency**. To check the relevance between each feature and the ratio, where the ratio is defined as the number in Hard Mode divide by the number of reported results, we apply correlation analysis. However, the results show that the features we obtain do not have strong relevance to the ratio. It is suggested that **players are not influenced by the solution word when choosing a game mode.**

In problem 2, we simplify the process of Wordle, and perform **Monte Carlo Simulation** on 355 words to obtain the score distribution of a word. We predict the distribution of word 'EERIE', which is **[0, 0, 7, 48, 42, 5, 0]**. To check the correctness of the strategy, we apply **Chi-Square Test** and obtain that the confidence of the prediction is **77%**. In our model, uncertainties arise from the difference between our simplified model and the complex reality.

In terms of problem 3, we defined the difficulty coefficient as **how many steps players need to solve the puzzle on average if every step is the optimal choice.** First of all, we acquire the actual average steps based on the distribution we got in question 2. If the actual average steps minus the average optimal steps are larger than 0, then we consider it an easy puzzle, instead, it is a hard one. Before developing our model, we first introduce the information entropy, which determine the information given by a word. Our model is constructed based on Monte Carlo Simulation and measure the **goodness** of word in selection stage. We introduce two indicators, the **information entropy** and the **weight of the words** to define the goodness of a single guess

As for problem 4, using the given distribution, we calculated the average score for each word. Through our observations, we have found that the average score is influenced by both the number of **repeated letter** and the **frequency of letters** in the word.

Last but not least, we summarize our findings and write a letter for the Puzzle Editor of the New York Times.

Keywords: FB-Prophet, Monte Carlo Simulation, Information Entropy, χ^2 Test

Contents

1 Introduction

1.1 Background information

During the pandemic, people have to be isolated at home, boringly and annoyingly. However, there was a couple who are dedicated to creating some interesting things to overcome this special period, they are Josh Wardle and Palak Shah. Since Ms. Shah is a big fan of word games, her boyfriend, a software engineer created a word-guessing game named Wordle.

At the very beginning, the game is designed for just two of them. However, with an aesthetic color scheme and easy-to-understand game rules, Wordle suddenly captured widespread attention. Within a 6×5 grid, players have to input a 5 letters word to guess the final answer with no more than 6 guesses. Each time players submit a word, it will return information related to the final answer, where green represents a correct letter in the correct position, yellow means a correct letter in the wrong position, and grey stands for the wrong letter.

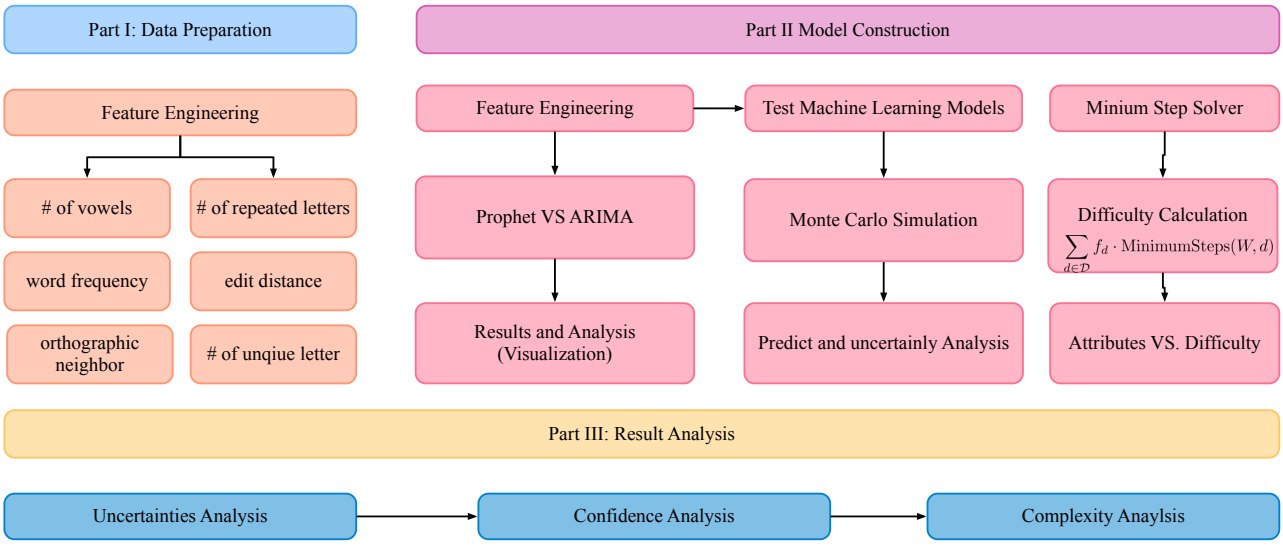
The love story beneath the Wordle game is romantic, yet, for us, we are more fascinated by finding attractive math and statistic regularity behind this welcome game.

1.2 Clarification and Restatements

To excavate the regularities behind Wordle, we need to:

1. Pre-process the data provided by COMAP's official.
2. Establish a model to explain the reason why the number of reported results varies daily and give a prediction interval of the number of results on March 1, 2023.
3. Extract some attributes of the word and check whether there are relations between the attributes and the ratio of the Number in hard mode to the Number of reported results, and give the reasons.
4. Develop a model to predict the distribution of the reported results for a specific solution, and forecast the result distribution of the word "EERIE",
5. Clarify the uncertainties in the model and predictions. Give confidence in our prediction model.
6. Construct a model to classify solution words by their difficulty, and extract attributes from different categories.
7. Use the model above to recognize how difficult the word "EERIE" is, and justify the accuracy of the model.
8. Discover some other characteristics from the given data set.
9. Write a letter for the Puzzle Editor of the New York Times and show our results.

1.3 Our work



2 Assumptions and Justifications

1. Players always give valid guesses in the dictionary which has **finite** words.
   ↪ Justification: The wordle developer stipulate that each guess is in the corpus defined by the developer. We learned from the source code that there are about 13,000 words in the corpus.

2. Assume the player never knows the solution list.
   ↪ Justification: We learn from the source code that wordle will only pick a word in a wordlist of 2,000 words as solution which is no available for players.

3 Abbreviations and Symbols

Before we begin analyzing the problems, it is necessary to clarify the abbreviations and symbols that we will be using in our discussion. These are shown below in Table 1:

Notation	Explanation
$\mathcal{D}$	The dictionary of a wordle game
$\mathbb{G}_i$	The possible guesses word list at the i^{th} guess
W_i	The word input by the user in the i^{th} guess
f_W	the frequency of the word W
$\mathcal{C}_{\mathbb{ANS}}(W)$	The correctness of guess W under the answer $\mathbb{ANS}$

Table 1: The notation we will used in the future discussion

4 Model Preparation

4.1 Data Preparation

1. We obtain all the possible guess list and possible solution list from the source code of wordle.

2. We obtain all the frequency of wordlist from wolfram [1].

4.2 Data Cleaning

4.2.1 Wrong sum of percentages

The instruction has already mentioned that: "the percentages may not always sum to 100% due to rounding.", we did observe some data whose sum of percentages is a little bit larger than 100%, but we believe that these data are still within a reasonable range, so we did not remove them. However, we still found one data with a total percentage exceeding the reasonable range, which is 126%. Therefore, we remove it from the dataset.

4.2.2 Outliers removal: reported results are too few

In the tour of the data, we found that the number of reported results is only 2569 on 2022-11-30. However, the second and third-fewest number of reported results had 15,554 and 20,001 observations, respectively, which is about 7 times larger than the previous one. So we are convinced that the data on 2022-11-30 is an outlier and remove it from the dataset.

4.2.3 Words not in the dictionary

As we assume in assumption 1, every solution word should be in the dictionary and have a fixed length of 5. We found a solution word '**marxh**' is not is the dictionary and two solution words are the length of 4, they are '**clen**' and '**tash**' respectively. We remove these words to make our dataset satisfy our requirements.

5 Model I

5.1 Method Overview

By using the given data, we have adopted the ARIMA and FB-Prophet model to obtain a prediction interval for the number of reported results on March 1, 2023. Compared two models' performances by using MAE, RMSE, and MAPE, we use the better one to summarize the reason that causes the variation of people who play the game.

After extracting features of words according to the linguistic significance and statistical significance, we check the relation between the features and the ratio of Number in Hard Mode to the Number of reported results. However, the outcomes reveal that the relations between them are very weak. Under this observation, we propose reasonable explanations to interpret it, including the definition of the ratio, features, and the players' behavior.

5.2 ARIMA Model

Notation	Explanation
p	order of the AR(Auto Regressive)
d	the number of differences order
q	order of the MA(Moving Average)
ϕ	AR Coefficient
θ	MA Coefficient

Table 2: The variables and parameters we will used in the ARIMA

Autoregressive Integrated Moving Average(ARIMA), a commonly used time series forecasting model, is a combination of the Autoregressive(AR) model and the Moving Average(MA) model. In $\mathrm{ARIMA}(p, d, q)$, d is the order of differencing, p is the order of AR, and q is the order of MA.

The AR(p) model is defined as the following equation.

$$z_t = \alpha + \phi_1 z_{t-1} + \phi_2 z_{t-2} + \cdots + \phi_p z_{t-p} + w_t \tag{1}$$

where $z_{t-1}, z_{t-2}, ..., z_{t-p}$ are the past values, $\phi_1, \phi_2, ..., \phi_p$ are the parameters of the model, and w_t is white noise.

The MA(q) model is defined as the following equation.

$$z_t = \alpha + w_t + \theta_1 w_{t-1} + \theta_2 w_{t-2} + \cdots + \theta_q w_{t-q} \tag{2}$$

where $w_t, w_{t-1}, ..., w_{t-q}$ are the error terms of the model for the respective lags mentioned in the AR model.

Finally, the $\mathrm{ARIMA}(p, d, q)$ is defined as follows.

$$z_t = \alpha + \sum_{i=1}^{p} \phi_i z_{t-i} + w_t + \sum_{j=1}^{q} \theta_j w_{t-j} \tag{3}$$

5.3 FB-Prophet model

Facebook Prophet model, which is an open-source tool from Facebook, is based on a decomposable additive model. It can accommodate multiple period seasonality, floating holidays, and piecewise trends. Unlike ARIMA, the FB-Prophet model ignores the temporal dependence of the data, so there is no need to perform some special operations on data to maintain the isochronism of the data. Moreover, it can automatically handle missing values and outliers. The FB-Prophet model, with three main components, is defined as follows.

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t \tag{4}$$

where $g(t)$ is the trend function, $s(t)$ stands for seasonality, $h(t)$ represents the effect of holidays and ϵ is the error term. What has to be mentioned is, the formula above, named additive model, is suitable when trend and seasonality act independently. In our case, the multiplicative model is much better, since the size of the seasonal effect depends on the size of the trend to some extent.

$$y(t) = g(t) * s(t) * h(t) * \epsilon_t \tag{5}$$

5.3.1 The trend model

The trend model is the core component of FB-Prophet, which analyzes and fits non-periodic changes in time series. To satisfy different conditions, it provides two trend models.

The first one is called the Piecewise Logistic Growth model, which is used if there is saturation when the trend reaches a certain level. The formula of it is:

$$g(t) = \frac{C(t)}{1 + \exp(-(k + \mathbf{a}(t)^T\delta)(t - (m + \mathbf{a}(t)^T\gamma)))} \tag{6}$$

The most important part of this model is $C(t)$, which is the model capacity.

The second trend model is a model based on segmented linear functions. Here is the model:

$$g(t) = (k + \mathbf{a}(t)^T\delta)t + (m + \mathbf{a}(t)^T\gamma) \tag{7}$$

where same as before k is the growth rate, θ is the change in growth rate, and m is the offset parameter.

5.3.2 Seasonality

$S(t)$ represents the periodic variation of the time series. Expressed by the Fourier series, $S(t)$ can simulate weekly, monthly, annual, and other periodical trends. The formula of the model is shown as follows.

$$s(t) = \sum_{n=1}^{N} \left(a_n \cos\left(\frac{2\pi nt}{P}\right) + b_n \sin\left(\frac{2\pi nt}{P}\right) \right) \tag{8}$$

Here, n denotes the number of periods used in the model, t is the length of the period of the desired time series, and $2n$ represents the number of parameters to be estimated to fit the seasonality. The setting of n needs to be considered in conjunction with t. The model will have a better performance in fitting complex seasonality with a bigger n.

People generally say that a time series exhibit seasonality by checking whether the mean value of the series varies regularly and periodically. In this present case, the surge point of the number who play Wordle is considered as a 'season'. However, this season is uncertain. By default, FB-Prophet only returns the trend and the uncertainty of the observation noise. To obtain the seasonal uncertainty, we use a complete Bayesian sampling during the programming procedure.

5.3.3 Holidays and events

Generally speaking, holidays and events will have a great influence on time series prediction. The FB-Prophet model incorporates these influencing factors into the model as a priori knowledge, which has major significance for the improvement of the model accuracy. The principle of the model is:

$$h(t) = Z(t)\kappa \tag{9}$$

where $Z(t)$ is indicator function, and $\kappa \sim \text{Normal}(0, \nu^2)$.

5.3.4 The evaluation of the model

To evaluate the performance of two time-series models, we use the following indicators:
Mean Absolute Error(MAE):

$$\text{MAE} = \frac{1}{N}\sum_{k=1}^{N}|z_k - \hat{z_k}| \tag{10}$$

Root Mean Square Error(RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(z_k - \hat{z_k}^2)} \tag{11}$$

Mean Absolute Percentage Error(MAPE):

$$\text{MAPE} = \frac{100}{N}\sum_{k=1}^{N}\left|\frac{z_k - \hat{z_k}}{z_k}\right| \tag{12}$$

where z_k is the actual value, and $\hat{z_k}$ is the predicted value given by the model for the k_{th} instance, $\hat{z}$ is the average value of z, and N is the total number of samples.

5.3.5 Our results and explanations

Model	MAE	RMSE	MAPE
ARIMA	3757.6584	21603785.0163	0.1324
Prophet	0.5500	0.4125	0.3369

Table 3: ARIMA VS. Prophet

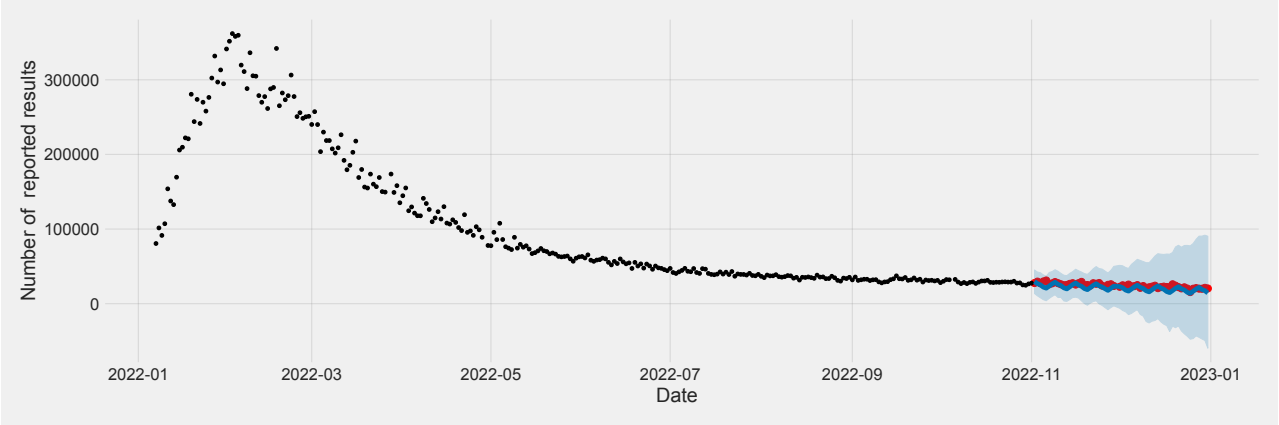


Figure 1: Actual VS. Prophet prediction

Results in the Table 3 show that FB-Prophet has a better performance in predicting the number of reported results. However, the confidence interval given by FB-Prophet is inferred based on historical data and model assumptions. Although it can be used to access the reliability and accuracy of prediction results, it does not provide information on the probability distribution of possible future values. Thus, here, we do not believe the interval given by FB-Prophet.

The single prediction value given by FB-Prophet on March 1, 2023, is 12937.431632. Based on the observed historical values, we find that the number of people who play games on two adjacent days is generally not too different. Therefore, to make the predicted value the midpoint, we construct the interval by floating 2000 up and down from the midpoint. In the end, the prediction interval given by us is [**10937.431632, 14937.431632**].

As we introduced above, FB-Prophet has a good performance when the time series have seasonality. The number of people who play Wordle may increase due to some external factors, which can be generalized as human behavior, and this surge point we can consider as 'season'. For example, when people are sharing the game on Twitter or media outreach this game, other people who do not know the game or do not like it will also want to have a try. Later, it will cause a "Wordle Season" because of the advocacy of the game. Moreover, when there are holidays or weekends, people will have time to remember to play Wordle. All this unpredictable behavior causes a variety of daily reported results.

5.4 Feature Engineering

By reading some linguistic articles and observing given data, we extract some attributes of words. They can be generally divided into two major parts. The first part is the attributes of the letters in the word, which contains the number of vowels, the number of repeated letters, and the number of unique letters. The second part is the attributes of an independent word, which

includes the number of orthographic neighbors, word frequency, and edit distance.

"The percentage of scores reported that were played in Hard Mode" can be represented as a mathematical term:

$$Ratio = \frac{\text{Number in hard mode}}{\text{Number of reported results}} \times 100\% \tag{13}$$

which can also be recognized as the percentage of people who choose to play the Hard Mode. To determine how the observed features affect people whether play the Hard Mode of Wordle or not, we use correlation to test the relevance.

$$Corr(X,Y) = \frac{Cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E\left[(X-\mu_X)(Y-\mu_Y)\right]}{\sigma_X \sigma_Y} \tag{14}$$

5.4.1 Number of vowels

The English alphabet has 26 letters, and it is composed of 5 vowels (a, e, i, o, u), 19 consonants, and 2 letters (y and w) that can function as vowels and consonants. A word in English is a combination of several syllables. In most cases, a syllable contains one vowel sound and one consonant sound in English. Since the 5 letters limit in Wordle, the number of vowels in a solution word is vary from 0 to 4.

5.4.2 Number of repeated letters

Some letters will repeatedly appear in a word. It shows some features to some extent. Therefore, we calculate the number of repeated letters in each word, such as the word "apple" has two repeated letters and the word "letter" has 4.

5.4.3 Number of unique letter

The number of unique letter of a word is defined as the occurences of the each letter is we only count repeated letters once. For example, the number of unique letter of word "apple" is 4 since we count the letter 'p' only once.

5.4.4 Orthographic neighbor

In the realm of linguistics, there is an idea of orthographic neighbor, which stands for a word that differs from another word with the same length by only one letter. We count the number of orthographic neighbors of a word, and later compute the correlation between the Ratio and it to check the degree of relevance.

5.4.5 Word frequency

To describe the commonness of a word, we use the frequency of word in typical published English text [1], denoted as f_w for word w.

5.4.6 Expected weighted edit distance

Players need to have a strong ability to associate one word with another in order to succeed in a wordle game. So we introduce a string metric which is a way of quantifying how dissimilar two words are to one another. Edit distance $d(a, b)$ is the minimum-weight series of edit operations that transforms a into b. If $d(a, b)$ is small, which means player can easily associate a with b. However, it's not enough, for example, we can easily transform $reset$ to $resat$ in a single step, but the frequency of $resat$ is too small that the player may never guess this word, so we define the weighted edit distance:

$$d_{\text{weighted}}(a, b) = \frac{d(a, b)}{f_b} \tag{15}$$

then we define expected weighted edit distance of a word w:

$$\mathbb{E}[w]_{d_{\text{weighted}}} = \frac{\sum_{d \in \mathcal{D}} d_{\text{weighted}}(w, d)}{\text{size}(\mathcal{D})} \tag{16}$$

5.4.7 Our result

Neighbors	Vowels	Frequency	Repeated Letter	Distance	Unique Letters
0.021983	0.079431	-0.114413	0.078283	0.059525	-0.081866

Table 4: Correlation between percentage and word's attribute

After calculating the correlation between each feature and *Ratio*, we can conclude that there is no strong relation between them. That is, these typical linguistic or statistical features of the word do not have a significant relevance with the percentage of people who play the Hard Mode.

The indicators of the correlation analysis are the features and the Ratio. We select features from the solution words, and the Ratio is defined as the ratio of the number of people who play Hard Mode to the total number of people who play Wordle, which represents the percentage of people who play Hard Mode. It makes sense that there is no relation between them because players do not know the solution word while they are guessing the word. To prove our idea, let us first assume there is a relationship between the difficulty of the solution word and the ratio,

then it means that people are more likely to choose to play hard mode if the solution word is easy or hard. However, it is conflict with the definition of Wordle, and so do the other features of solution word. In fact, the ratio is more related to players' behavior and preference but not the features of the solution word.

6 Probability Distribution Forecasting Models Using Monte Carlo

6.1 Performance of machine learning

Model	MSE	RMSE	MAE	MAPE	R^2
BPN	57.748	7.599	6.068	27.665	0.112
Random forest	58.252	7.632	5.951	27.908	0.104
XGboost	15.792	3.974	2.891	57.303	0.024
KNN	15.698	3.962	3.043	53.51	0.03

Table 5: Performance of Machine Learning

To predict the distribution of the percentage of score, the first method come into our mind is machine learning. However, we have tried many models to do the prediction, but none of them return a good performance. The detailed information is shown in Table 5

6.2 The Monte Carlo Simulation

Since all the results we have gotten from machine learning and correlation analysis act poorly, we realize that both of them are not the appropriate method to predict the distribution of the score's percentage. We delved into Wordle and deductively reasoned the whole process of it, and discovered that a player chooses a word to play at a time is a random event. Each player has a different word-choosing preference and this is also a random event. Therefore, the game can be seen as a random experiment. For a specific solution word, the score that a player finally gets while playing the game is a discrete random variable. After multiple repeated games played by different users, it will obtain a probability mass function. To estimate the possible outcomes of these uncertain events, we use the Monte Carlo Simulation.

The basic idea of the Monte Carlo Simulation is to take a large number of samples from the target distribution and use these samples to analyze the properties of the distribution. Based on the Glivenko-Cantelli theorem, we know that when the sample size goes to infinity, the empirical distribution function will converge with the real distribution function. Supported by the Law of

Large Numbers, we know that the sample mean converges to the population expectation as the sample size approaches infinity. It represents that our simulation is more accurate if we have more samples. To confirm our suspicions, we have established a model based on these ideas to simulate the distribution of existing data.

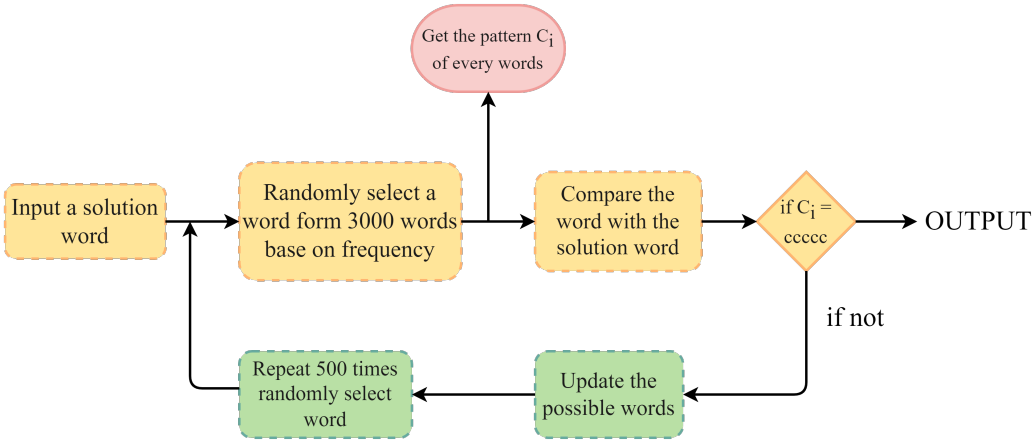


Figure 2: Monte Carlo Simulation Process

6.3 Monte Carlo Simulation results

For this problem, in order to predict the precentage of (1,2,3,4,5,6,X) for the word 'EERIE' on March 1, 2023, we use the Monte Carlo Simulation to predict probability distribution of scores. We can obtain the result, the precentage of (1,2,3,4,5,6,X) = [0,0,7,48,40,5,0].

6.4 Uncertainties and the confidence of the model

6.4.1 The confidence of our model

Since we are sampling from a probability mass function, let $pmf_{\text{pred},W}$ denoted the predicted $pmf_{\text{actual},W}$. A χ^2 test is conducted to check whether there are differences between $pmf_{\text{pred},W}$ and $pmf_{\text{actual},W}$, We are objective to test

$$H_0 : pmf_{\text{pred},W} = pmf_{\text{actual},W}$$
$$V.S.$$
$$H_1 : pmf_{\text{pred},W} \neq pmf_{\text{actual},W}$$

if p value is greater then 0.05, which means we can not rejected null hypothesis, which conclude that $pmf_{\text{pred},W} = pmf_{\text{actual},W}$, which indicate that our result if good.

We apply the test for every W, and we obtain that 23% of the word reject the null and hypothesis, which means we are 77% confident on our results.

The results show that the Monte Carlo Simulation is a practical method and our model is valid. Choose the word 'EERIE' as the solution word, and play a repeated game 50000 times.

6.4.2 Uncertainties of the model

Model uncertainty: Our model used to estimate the distribution is an approximation of the real system, and may not capture all of the complexities of interactions that affect the system.

Sampling uncertainty: In our model, the target distribution is a word list with a size of 2000. These selected words all have a word frequency greater than 20%. However, this is not quite in line with reality. Players are possible to choose a word that is not that commonly used. Another point is every input word is selected randomly from the possible answer word list. In practice, when encountered with the same list of possible answers, people are more likely to use remember and use a word that has a higher frequency.

6.4.3 Improvement

To improve the model, first of all, we can add a weight to each word according to the word frequency, so as to bring the word selection process closer to real player manipulation. Moreover, we can increase the number of simulations or make the target word list more reliable to improve the accuracy of the simulation.

7 Difficulty Classification Model Based on Information Theory

7.1 Difficulty of a wordle game

Players are dedicated to solving a wordle game in the minimum number of steps, that's also how we define the difficulty coefficient:

$$\Gamma(W) = \sum_{d \in \mathcal{D}} f_d \cdot \mathrm{MinimumSteps}(W, d) \tag{17}$$

where W is the answer of the wordle game, d is the initial guess. Since Wordle is NP-hard [2] and the find the winning strategy of solving a wordle game is NP-complete [3], we simply purpose a greedy algorithm to solve it.

7.2 Define the wordle game mathematically

7.2.1 Correctness of the word

We define $\mathcal{C}_{\mathbb{ANS}}(W)$ is the correctness of a word of length n under the solution $\mathbb{ANS}$ (length is also n), then $\mathcal{C}_{\mathbb{ANS}}(W)$ is a vector,

$$[c_1, c_2, c_3, \cdots, c_n] \tag{18}$$

then we can calculate the correctness by calculating every c in $[c_1, c_2, c_3, \cdots, c_n]$, where:

$$c_i = \begin{cases} \texttt{M} & \text{if } \alpha_i \in \{\beta_1, \cdots, \beta_n\} \wedge \alpha_i \neq \beta_i \\ \texttt{C} & \text{if } \alpha_i = \beta_i \\ \texttt{W} & \text{if } \alpha_i \notin \{\beta_1, \cdots, \beta_n\} \end{cases} \tag{19}$$

where α_i is the i^{th} character of W, β_i is the i^{th} character of $\mathbb{ANS}$,

7.2.2 'Information' given by a guess

Let's play a real wordle game, assume the initial guess W_1 is 'slate', the correctness is $[\texttt{M}, \texttt{W}, \texttt{M}, \texttt{W}, \texttt{W}]$ for 'slate', then what should we guess in the next round ? An intuitive idea is to find the possible answers and choose one, for example:

$$\text{abbas}, \text{abris}, \text{abysm}, \text{abyss} \cdots$$

We can obtain the all **possible candidate answers** by:

$$\mathbb{G}_2 = \{\forall g \in \mathbb{G}_1, \mathcal{C}_g(W_1) = \mathcal{C}_{\mathbb{ANS}}(W_1)\} \tag{20}$$

where $\mathbb{G} = \{g_1, g_2, \cdots, g_n\}$ and $\mathbb{G}_1$ is $\mathcal{D}$ in this case since players do no know the solution list(See Assumption 2), that is, we reduce the possible answer list from $\mathbb{G}_1$ to $\mathbb{G}_2$, we can define the information we obtain from this guess:

$$I = \frac{\mathbb{G}_2}{\mathbb{G}_1} \cdot -\log_2 \frac{\mathbb{G}_2}{\mathbb{G}_1} \tag{21}$$

7.3 Greedy Algorithm: Solve Wordle in minimum steps

We purpose the a greedy algorithm to maximum the goodness of each guesses, which is show in algorithm 1

Algorithm 1 Minimum number of guess for a wordle game

Input: dictionary $\mathcal{D}$, initial guess $W_1 \in \mathcal{D}$, solution of the game $\mathbb{ANS}$

Output: the minimum number of guess to reach the correct answer

Initialization $\mathbb{G}_1 \leftarrow \mathcal{D}$, $i \leftarrow 1$
loop
  $\mathcal{C}_i \leftarrow$ correctness of W_i using $\mathcal{C}_{\mathbb{ANS}}(W_i)$
  if $\mathcal{C}_i =$ `CCCCC` **then**
    Return number of guess i
  end if
  $\mathbb{G}_{i+1} \leftarrow \{\forall g \in \mathbb{G}_i, \mathcal{C}_g(W_i) = \mathcal{C}_i\}$
  $w_{i+1} \leftarrow \underset{g_n}{\text{argmax}}\, \text{Goodness}(g_n)$ where $g_n \in \mathbb{G}_{i+1}$
  $i \leftarrow i + 1$
end loop
Return number of guess i

7.4 Measure of Goodness

7.4.1 Expected Information

As we mentioned before, we can quantify the information provided by each user's guess every time it is made. An intuitive idea is to use the word which can provide maximum information in each guess. The more information provided by a word, the better the word is. So we measure the goodness of word by the epected value

$$\text{Goodness}(W) = E[I_W] = -\sum_{\mathcal{C} \in \mathbb{C}} P_W(\mathcal{C}) \cdot \log_2 P_W(\mathcal{C}) \tag{22}$$

where $\mathbb{C}$ is all possible 3^5 correctnesses and $P_W(\mathcal{C})$ is given by:

$$P_W(\mathcal{C}) = \left(\frac{\{\forall g \in \mathbb{G}, \mathcal{C}_W(g) = \mathcal{C}\}}{\mathbb{G}} \right) \tag{23}$$

7.4.2 Weighted Information

If $E[I_{W_1}] = E[I_{W_2}]$, we compare f_{W_1} and f_{W_2} to decide which to choose.

7.5 Classification of a word

We divide word into two classification, easy and hard, a word is recognized as a hard word if $E[score] - E_{\text{user}}[score] > 0$, vice versa.

7.6 Attribute of word for each classification

As we defined before, the difference between $E[score]$ and $E_{\text{user}}[score]$ gives out the classification of a word. However, $E[score]$ normally are not correlated with the attribute of the word, but the each guess. So the attribute of word for each classification is determine by $E_{\text{user}}[score]$. In our observation, we found that

1. if **num of repeat letters** increases, the $E_{\text{user}}[score]$ goes larger.
2. if **num of unique letters** increases, the $E_{\text{user}}[score]$ goes lower.

We can conclude that the hard word usually have more repeated letters, simple word usually have more unique letters.

7.7 Our results

The difficulty of word 'EERIE' can be calculated using the Equation (17)

$$\Gamma(\text{EERIE}) = 4.1343$$

The expected scored to solve this word is 4.43 which is obtain from the distribution predicted in problem 2. Since $4.43 > \Gamma(\text{EERIE})$, we classify the word is **HARD**.

8 Other features

By using the given distribution, we calculate the average score for every given solution word. Shown in Figure 3. Sorting them from small to large, we select the first ten words and the last ten words for extracting features. The word with a bigger average score represents players needing more steps to solve the puzzle on average. Otherwise, players can achieve the answer just in fewer steps.

It is easy to analyze that word with a smaller average score basically has no duplicate letters, and they have some letters that appear very frequently in life, such as 't' or 'a'. On the opposite,

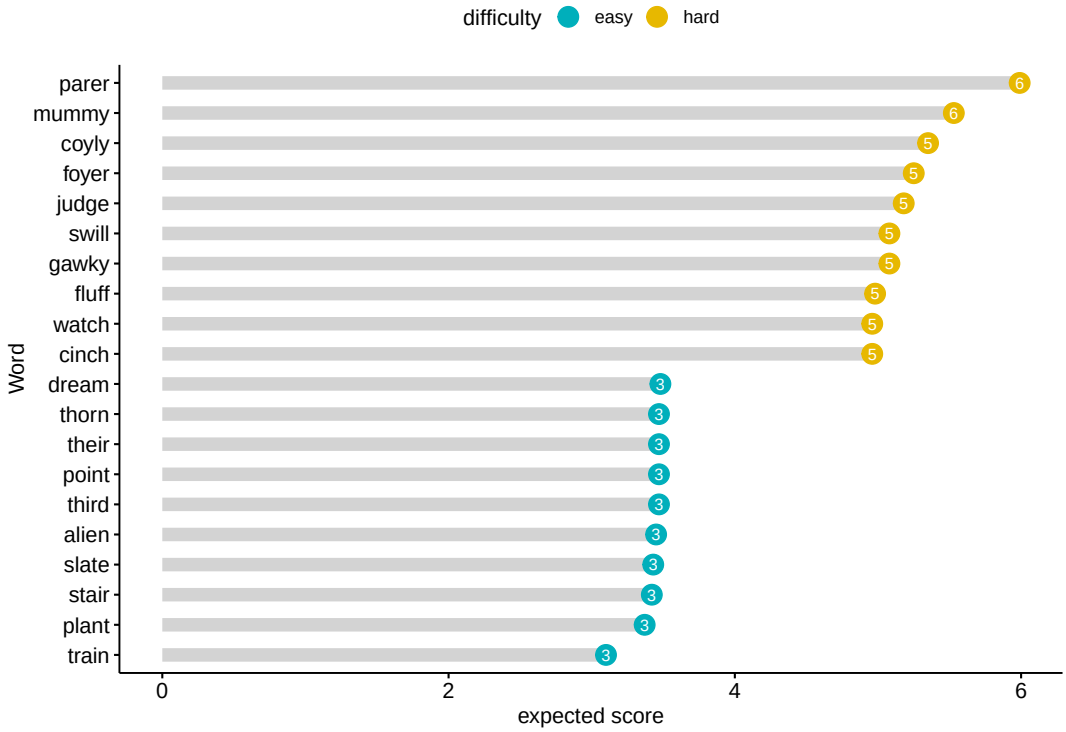


Figure 3: Top 10 hard word and easy word analysis

the word that has a higher average score has more duplicate letters, and some letters that do not appear frequently showed up in the word, such as 'w' or 'y'.

Under the words of the current data set, by counting the number of all letters, we get that e is the most, q, z, j are the least, almost 0.

9 Strengths and Weaknesses

9.1 Strengths

1. High interpretability: In problem 1, we use the FB-Prophet model, which performs well in understanding the underlying pattern of the data. When dealing with a time series, which has uncertain seasonality, such as the one in this problem, FB-Prophet can automatically detect the trend and the changes in the trends. It helps us handle complex reality cases in predicting the number of people who play Wordle.

2. High accuracy: Monte Carlo simulation is guided by probabilistic statistical theory. It can provide accurate estimates of complex systems and processes. So even when dealing with the problem of predicting the distribution of the percentage of scores, which is related to

human behavior, it also has high accuracy.

9.2 Weaknesses

1. Lack of probabilistic forecasting: FB-Prophet only provides point estimates of future values, but not a range of probability distributions. Therefore, we use the single predicted value given by FB-Prophet as the midpoint of the interval to construct a reasonable prediction interval by historical observations, which is not convincing enough.

2. Assumptions and simplifications: Monte Carlo Simulation often needs to raise assumptions and simplify the real problem to develop a model. Although these simplifications can make the simulation more tractable, they can also lead to erroneous or inaccurate results.

3. High time complexity: The model we developed in question three is used to solve an NP-hard question. The algorithm is NP-complete, which will cost a large amount of time.

10 A Letter to the Puzzle Editor of the New York Times

TO: The puzzle editor of the New York Times

FROM: MCM Team # 2305804

DATE: February 21, 2023

SUBJECT: Unveil the mystery of Wordle.

Dear Sir or Madam:

We are glad to have the opportunity to write to you with some of our findings and thoughts on the game Wordle. Our team has developed three models to provide you with information about the game's dynamics. These models allow you to predict the number of people who play Wordle, the distribution of the percentage of the score, and the difficulty of the solution word.

The first model is based on FB-Prophet, which has a very outstanding performance in predicting time series data that has seasonality. Since the data of people who play Wordle is affected by many external factors that cannot be predicted, FB-Prophet is a very suitable model for you. The evaluation given by error indicators shows that our model acts well.

The second model is a procedure of Monte Carlo Simulation. By randomly selecting a word from the answer list, we can simulate the process of players playing the game. Utilizing this method, we can obtain a predicted distribution of the score's percentage after determining a solution word.

We develop the third model based on Information Theory and Mathematics for you to evaluate the difficulty of the solution word. This model is an advanced version of the second model, which just adds two new indicators for choosing the best word to get the solution more quickly. We would like to explain these two indicators to you. The first one is information entropy, which defines how much information can be provided under the solution word and a guessing word. The second one is the weight of the word, it is reasonable for players to choose a more common

word as a guess. Therefore, after adding weight to the word, the simulation of the game is closer to the reality of the situation.

We would like to highlight that the difficulty of the solution word has no relation to how many people choose Hard Mode, but in somehow it does have a relation to the score.

We sincerely hope that our models and findings can help you to improve Wordle and make it even more enjoyable for people who love it. Thanks for taking you time to read our letter.

Best,

MCM Team #2305804

References

[1] W. Research, "Wordfrequencydata." https://reference.wolfram.com/language/ref/WordFrequencyData.html", 2016. [Accessed: 18-February-2023].

[2] D. Lokshtanov and B. Subercaseaux, "Wordle is np-hard," *arXiv preprint arXiv:2203.16713*, 2022.

[3] W. Rosenbaum, "Finding a winning strategy for wordle is np-complete," *arXiv preprint arXiv:2204.04104*, 2022.

[4] L. Ramalho, *Fluent python*. O'Reilly Media, Inc., 2022.

[5] W. McKinney, *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython*. O'Reilly Media, Inc., 2012.

[6] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[7] T. pandas development team, "pandas-dev/pandas: Pandas," Feb. 2020.

[8] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[9] M. L. Waskom, "seaborn: statistical data visualization," *Journal of Open Source Software*, vol. 6, no. 60, p. 3021, 2021.

[10] S. J. Taylor and B. Letham, "Forecasting at scale," *The American Statistician*, vol. 72, no. 1, pp. 37–45, 2018.

[11] I. M. Hunter, "The solving of five-letter anagram problems," *British Journal of Psychology*, vol. 50, no. 3, pp. 193–206, 1959.

[12] S. Andrews, "The effect of orthographic similarity on lexical retrieval: Resolving neighborhood conflicts," *Psychonomic bulletin & review*, vol. 4, no. 4, pp. 439–461, 1997.

[13] 3Blue1Brown, "Solving wordle using information theory." Website, 2022. https://www.youtube.com/watch?v=v68zYyaEmEA.