Problem Chosen	2023 MCM/ICM Summary Sheet	Team Control Number
C		2311153

WPGM: Reveal Secrets Hidden Behind Wordle and Predict the Game Future

Summary

As the web-based word game called **Wordle** gained its popularity at the beginning of 2022 and has witnessed the journey of a great number of players, the large scale of game data is a tempting resources for studying. In this paper, we propose the **Word-attributes and Players Game Model (WPGM)** to reveal the current relations between word attributes, players numbers and game results distribution, and predict future game information based on **Machine Learning, Clustering Algorithm and Time Series Forecast**.

First, the variation of daily reported results number is explained by the **fitting curve** of two exponential functions superimposed based on **Differential Equations** and **Newton's Law of Cooling**. Then we apply **LSTM** to take the fluctuations into consideration and forecast the time series. **Hypothesis Testing** is later used to calculate the prediction interval.

Second, we select typical word attributes and describe the game results distribution as **Gaussian Distribution** characterized by **Trial Times Mean** and **Trial Times Variance**. Then we conduct **Correlation Analysis** to quantify the relation between word attributes and results distribution as well as the Hard-Mode Player Ratio.

Third, we apply **Principle Component Analysis (PCA)** to extract three relatively independent variables from the word attributes and further study their impacts on the results distribution. We use the **Back-Propagation Neural Network (BPNN)** to train a **multiple non-linear regression model** that characterize the relationship between the three principle variables and the game distribution. We split the sample data into training set and testing set to validate the robustness of the model.

Fourth, we assign **Entropy Weight (EW)** to each Trial Times so that the information of word difficulty implied in distribution data can be fully utilized, and the **Weighted Distribution Scores** are calculated. Later, the words are embedded into $\mathbb{R}^2$ word vectors with both **Trial Times Mean** and **Weighted Distribution Scores** considered. To classify the words into 4 different difficulty categories, we resort to **Fuzzy C-Means Algorithm (FCM)** to cluster the embedded word vectors. The accuracy of classification is calculated by the **Average Membership Degree**. We then apply the **Word Difficulty Classification Model** for the word "Errie", and label it with the highest level of difficulty.

In addition, we conduct further Data Mining, and discover the increasing trend of Hard-Mode Player Ratio, as well as the common properties that difficult and easy solution words commonly share.

Our model passes the sensitivity analysis for slight variation of **Fuzzy Exponent** and **Machine Learning Inputs** and shows its robustness.

In conclusion, via our WPGM system, we are able to simulate the relations between words, players and game results, as well as give a rather accurate prediction in the short future.

Keywords: Wordle; Time Series Forecasting; Correlation Analysis; Principle Component Analysis; Back-Propagation Neural Network; Entropy Weight Method; Fuzzy C-Means Algorithm;

Contents

1 Introduction

1.1 Background

A web-based word game called **Wordle** rapidly got popularity at the beginning of 2022, attracting the players by its neatness, high playability. Although the history of word-guessing game can be traced back to 1955, the age of the Internet gives new birth to it, making Wordle play an important role in both personal recreation and social networks.[1] With one puzzle a day, players only need to take three minutes every day to enjoy the game and have fun competing with their friends for fewer turns to work it out. Hard Mode is also provided for more professional players who wants to challenge themselves.[2]

Due to various attributes of words, data has shown that people tend to have a different distribution for tries with regard to the solution word. Since the popularity of Wordle also changes with time, the total number scores and scores on Hard Mode vary accordingly.

Figure 1: The Popular Game Wordle[3]

1.2 Problem Restatement

1. Design a mathematical model to explain the variation of the number of reported results, and make a prediction on a certain day in the future by applying the model.

2. Find relationship between certain attributes of solution words and the percentage of scores reported that are played in Hard Mode.

3. Develop a model to predict the distribution of the reported results for a future date. Make uncertainty analysis of the model and the predictions, and apply it to a real case (for the word EERIE on March 1, 2023). Then, evaluate the degree of confidence of the model we develop.

4. Develop a model to tell the solution words apart by difficulty. Then, based on the model, identify the attributes of a given word (e.g. EERIE) and quantify its difficulty of guessing it to match it with the classification we make. Finally, make accuracy analysis of the model.

5. Discuss about other interesting features of the data provided.

1.3 Overview of Our Model

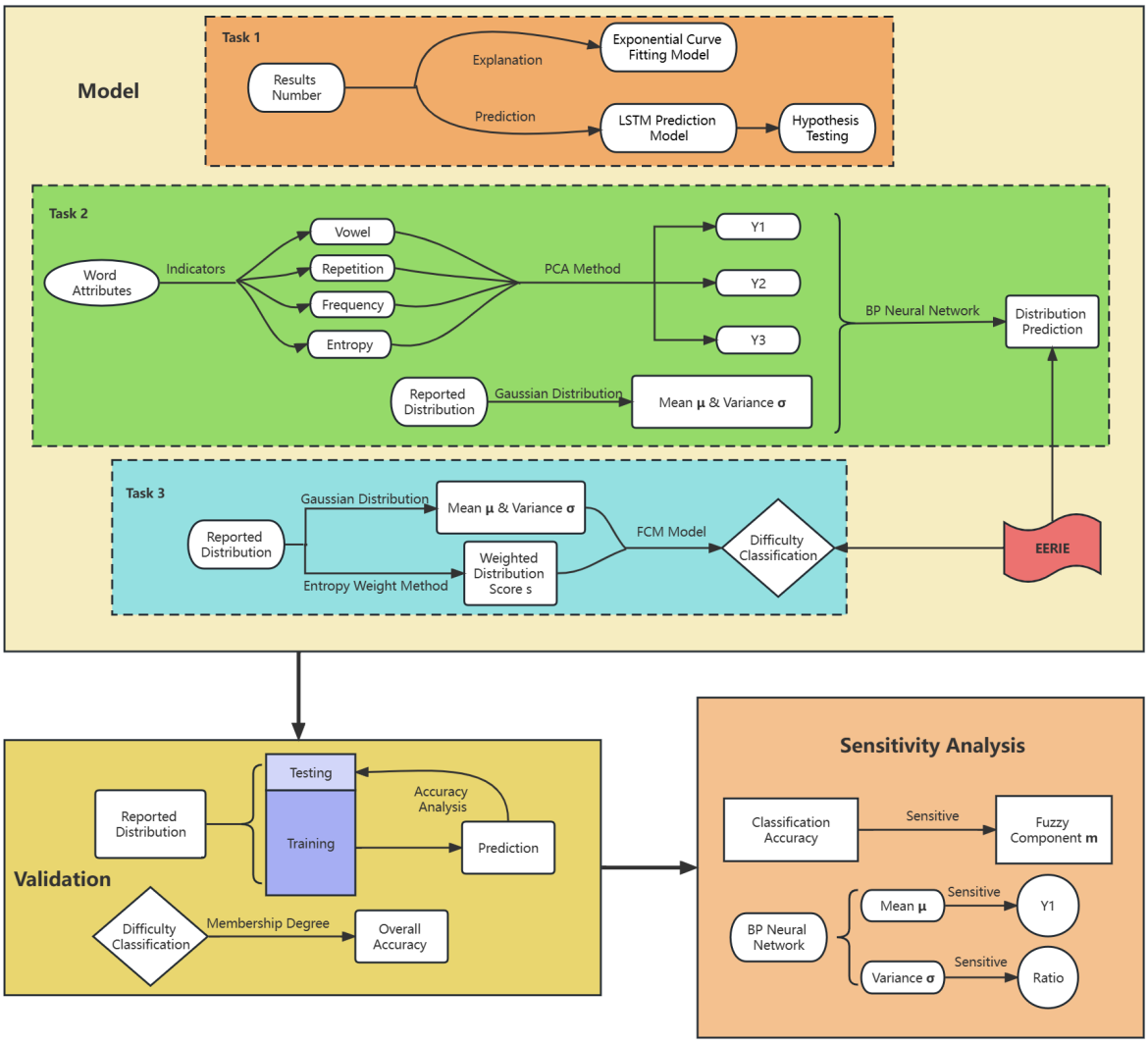


Figure 2: Flow Diagram for Word-attributes Players Game Model (WPGM)

2 Assumptions and Notations

2.1 Assumptions

1. Assume that the results reported from each player were independent to each other, and the trial times distributions for each player were roughly the same.

2. Assume that the trial times larger than 7 can be treated as 7 times for simplification of our model because whether exactly 7 times or more won't affect our model much.

2.2 Notations

Table 1: Notations

Symbols	Description
$N_{total}(t)$	The number of results at the t^{th} day
$N_{hard}(t)$	The number of results in hard mode at the t^{th} day
$N_{Uptrend}(t)$	The number of results for uptrend part at the t^{th} day
$N_{Downtrend}(t)$	The number of results for downtrend part at the t^{th} day
$N_{predict}$	The prediction number of results on March 1, 2023
N	The total number of words in the data set.
word_i	The i^{th} word in the data set.
X_{vow}	The number of Vowel in a word
X_{rep}	The multiplicity of repeated letters
X_{freq}	The frequency of the word in daily use
$X_{entropy}$	Entropy of the word
X_{ratio}	Ratio of Hard-Mode players among all players on that day
$R_{word-ratio}$	Correlation between word attributes and Hard-Mode Ratio
μ_i	The mean value of trial time results of the i^{th} word
σ_i	The variance of trial time results of the i^{th} word
Y_i	The i-th principle component
$t_{i,j}$	The percentage of results of j trial times cases of the i^{th} word
w_i	The weight for t_i when assessing the difficulty level
s_i	The weighted distribution score for the i^{th} word.
x_i	The vector of the i^{th} word projected to $\mathbb{R}^2$ for classification
c_i	The group center for the i^{th} difficulty level.
U_i	The i^{th} difficulty set.
$u_{i\,j}$	The Membership Degree for the i_{th} word to the j^{th} difficulty set.
M	Fuzzy exponent that determines the fuzziness of difficulty boundaries.
level_i	The difficulty level the i^{th} word belongs to.
accuracy_{cls}	The accuracy for difficulty classification

3 Task 1: Time Series and Correlation Analysis

3.1 Results Number Variation Model

3.1.1 Exponential Curve Fitting Model

Observing the results number time series, it's conspicuous that the number experiences a variation that it ascended to its peak and then descended with declining acceleration and persistent fluctuation, which indicated a declining popularity of Wordle Game.

To explain the variation, the results number variation can be divided to two parts:

- Part 1: Gradual and continuous increase of new-coming players.
- Part 2: Players leaving the game or playing the game at lower frequencies.

Therefore the results number can be summarized in one formula:

$$N_{Total}(t) = N_{Uptrend}(t) + N_{Downtrend}(t) \tag{1}$$

For Part 1, the rate of the increase of new-coming players is assumed to be linear proportional to the number of new-coming players:

$$\frac{\mathrm{d}}{\mathrm{d}t}N_{Uptrend}(t) = k_{Up}N_{Uptrend}(t) \tag{2}$$

Then, $N_{Uptrend}(t) = C_1\, e^{k_1\, t}$, where C_1 and k_1 are constant coefficients.

For Part 2, the declining trend of popularity resembled the physical cooling process. Studies have shown that this cooling process could be applied in economics or other fields to describe long term dynamics.[4] **Newton's law of cooling** has it that the speed of cooling is proportional to the temperature variation,

$$\frac{\mathrm{d}}{\mathrm{d}t}N_{Downtrend}(t) = -k_{Down}(N_{Downtrend}(t) - N_{Downtrend}(t_0)) \tag{3}$$

Then $N_{Downtrend}(t) = C_2\, e^{k_2\, t}$, where C_2 and k_2 are constant coefficients.

Therefore, the $N_{Total}(t)$ has the form:

$$N_{Total}(t) = C_1\, e^{k_1\, t} + C_2\, e^{k_2\, t} \tag{4}$$

Fit the curve and then came the results:

$$N_{Total}(t) = 544900\, e^{-0.01755\, t} + 18760\, e^{0.0007413\, t} \tag{5}$$

The **R-square is 0.9873**, which is quite close to 1, suggesting the excellent fitting effect. The fitting curve is plotted in Figure 3.

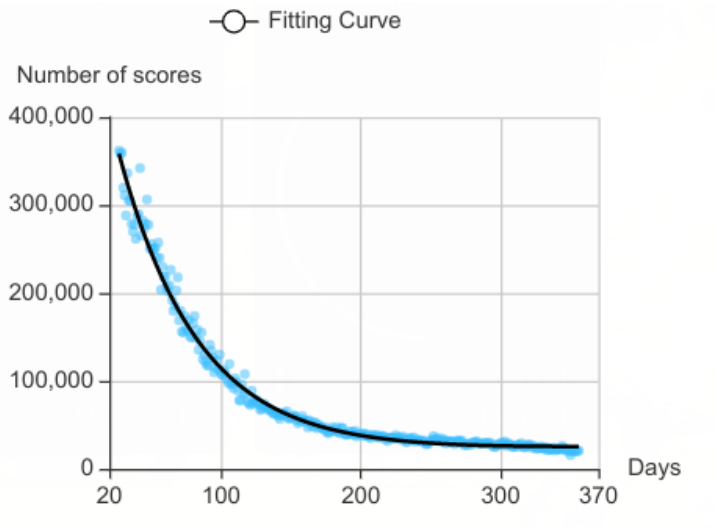


Figure 3: Fitting Curve for Results Number

3.1.2 LSTM Prediction Model

The **exponential fitting curve model** can explain the variation of number of players in a large scale, but the **smooth curve** it gives leaves out the **fluctuation information** of data. Therefore, we resorted to LSTM Prediction Model for more practical predictions, since the machine learning algorithm can study the details of fluctuations and flows.

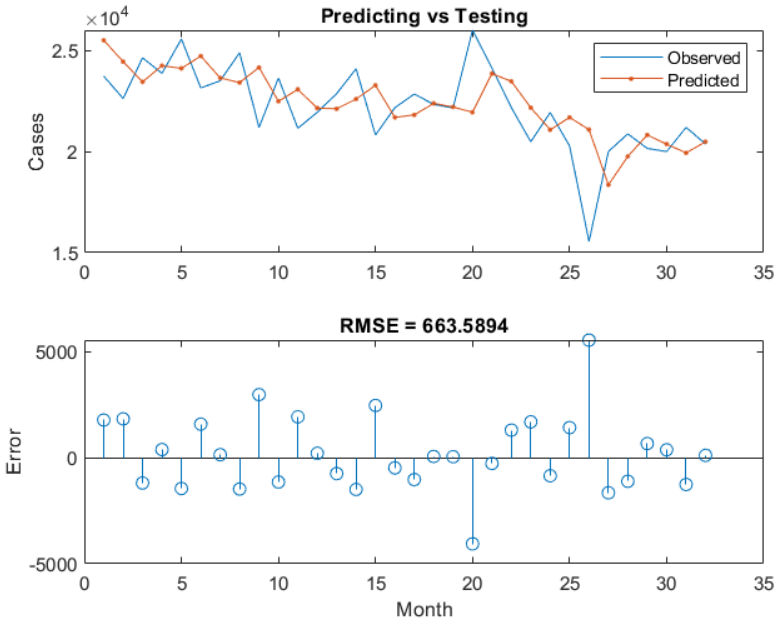


Figure 4: LSTM Prediction: Predicting Results and Testing data

The data of results number was split to 90% training set and 10% testing set. As can be seen from Figure 4, the predicting results were quite consistent with the testing data, and $\mathbf{RMSE} = \mathbf{663.5894}$.

Further predicted the results numbers for next 60 days, and the predicting results number for March 1, 2023 was **15315**.

Then we applied **Hypothesis Testing** for calculating the prediction interval at the $\alpha = 0.1$ significance level.

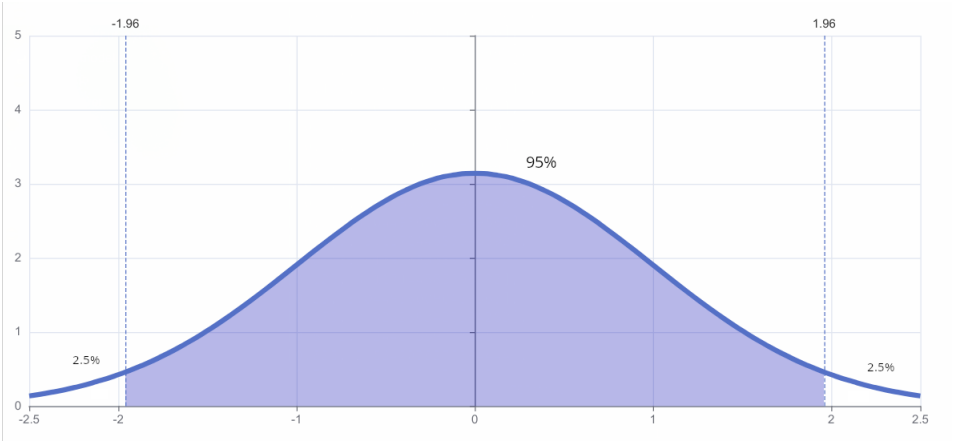


Figure 5: Normal Distribution with Labeled Probability

To accept the hypothesis that $N_{Predict}$ is a possible result, $N_{Predict}$ should satisfy the inequality that

$$\phi(|\frac{N_{predict} - 15315}{RMSE}|) \leq 97.5\% \tag{6}$$

where $\phi(x)$ is the PDF for normal distribution, and then

$$|\frac{N_{predict} - 15315}{RMSE}| \leq 1.96 \tag{7}$$

Therefore, the prediction interval for $N_{predict}$ is **[14014,16616]**.

3.2 Correlation Analysis

3.2.1 Word Attributes Model

In order to explore how **word attributes** affect the percentage of scores played in Hard Mode, we first introduce 4 indications that feature the word. Different words have their own attributes to make them unique. Throughout the paper, we mainly focus on 4 attributes of words that may **influence players' trial times for guessing**.

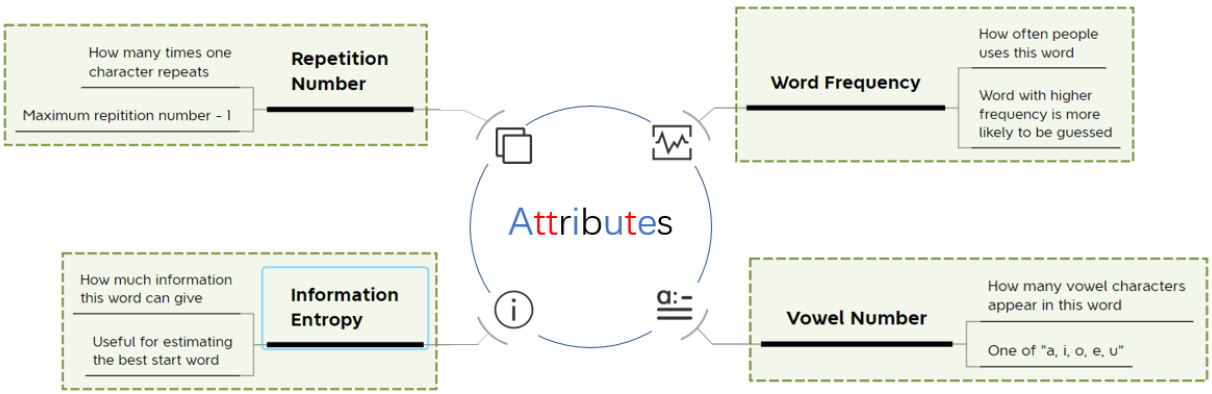


Figure 6: Word Attributes

- Vowel Number (X_{vow}) : The number of vowels in a word will influence its difficulty to be guessed. Since all words have vowels and there are only five of them among the alphabet, players are more likely to guess words based on them.

Figure 7: Illustration for Vowel Number (X_{vow})

- Repetition Number (X_{rep}) : Repeated letters appeared in a single word will impact the amount of information it gives out. X_{rep} was calculated by subtracting 1 from highest repetition times for letters in the word.

Figure 8: Illustration for Repetition Number (X_{rep})

- Word Frequency (X_{freq}) : The frequency of word used in daily life may directly influence the number of attempts to some extent.The data of word frequency could be found on the Internet.[5]

- Initial Information Entropy ($X_{entropy}$) : For this attribute, we learn from the definition of entropy to quantify the information that guessing a certain word can provide. In short, the higher the information entropy of the word is, the easier to guess the word. According to Sanderson's algorithm,[6] the entropy of guessing a word can be represented as the formula:

$$E[I] = \sum_x p(x) \cdot \log_2(\frac{1}{p(x)})$$

Where $p(x)$ refers to the probability of each outcome of guessing. For example, if we guess "crane" and the outcome is green on "c" and yellow on "a", the $p(x)$ in this case refers to all possible words that satisfy this constraint divided by the number of all five-letter words in database.

3.2.2 Word Attributes & Hard-Mode Player Ratio X_{ratio} Correlation Analysis

In addition to word attributes, we also need to name a factor to represent the percentage of hard mode scores among all the results.

- Hard-Mode Player Ratio (X_{ratio}) : We calculate the value of hard-mode results divided by all the results in a day. Our mission is to use models to find how this ration is related to the word attributes mentioned above.

The variables X_{vow},X_{rep},X_{freq},$X_{entropy}$ and X_{ratio} were first standardized to eliminate the influence of units by the *z-score* formula:

$$\tilde{x} = \frac{x - \mu_x}{\sigma_x}$$

Then we calculated covariance and correlation coefficients to evaluate the overall dependency of every two variables by formulas:

$$\text{Covariance: } \mathrm{Cov}(x, y) = E[(X - E[X]) \times (Y - E[Y])]$$

$$\text{Correlation Coefficient: } r(x, y) = \frac{\mathrm{Cov}(x, y)}{\sigma_x \, \sigma_y}$$

The correlation coefficients were compared and displayed in Figure 9.

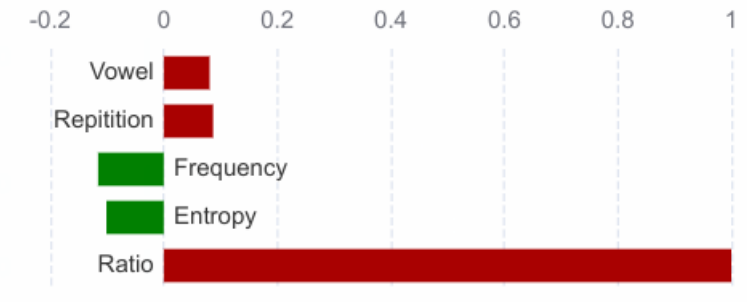


Figure 9: Correlation Coefficients with regard to Hard-Mode Player Ratio (X_{ratio})

It can be noticed that the correlation coefficients of X_{Ratio} with respect to the four word attributes respectively were **around 0**, which indicates that the Hard-Mode Player Ratio are **hardly related** to word attributes.

It can be naturally explained by the fact that players won't be influenced by words in-game when selecting game modes before-game.

3.2.3 Results Distribution Model

To simplify the distribution of trial times varying between words, based on the abundance of data source and the assumption that the trial times between players are independent and followed a roughly same distribution pattern, we applied **Central Limit Theorem** and assumed the distribution of trial times to be a **Gaussian Distribution**.[7]

Therefore, the results distribution can be characterized by two factors: **Trial Times Mean (μ)** and **Trial Times Variance (σ)**.

It is worthwhile noticing that the larger Mean (μ) will imply a **greater difficulty** to guess the word since more trial times are needed.

3.2.4 Word Attributes and Results Distribution Correlation Analysis

The correlation coefficients were calculated again to illustrate the correlation between word attributes and the results distribution.

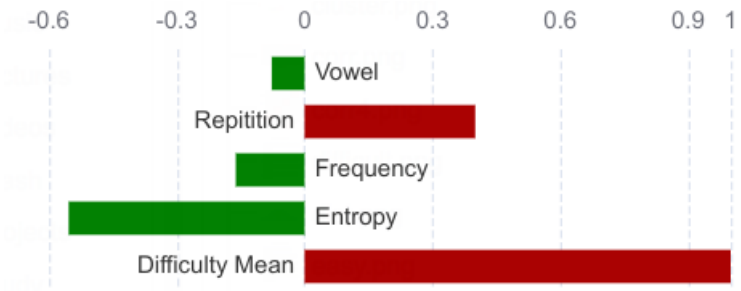


Figure 10: Covariance with regard to Hard-Mode Player Ratio

Little correlation was found related to the variance of distribution, but two important conclusions could be reached concerning the correlation between **word attributes and Trial Times Mean (μ)** from Figure 10.

(1) The Mean value (μ) is positively correlated to X_{rep}, which means that the larger X_{rep} is for a word, the harder it is for players to guess.

Possible Explanation: This coincides with the fact that the more repeated letters a word have, the less probability a player can guess a right letter.

(2) The Mean value (μ) is negatively correlated to $X_{entropy}$, which means that the larger $X_{entropy}$ is for a word, the easier it is for players to guess.

Possible Explanation: The information entropy ($X_{entropy}$) can describe the probability for the word to deduce to other words based on our definition. Correspondingly, it also implies **how much probability other words can deduce this word**. Therefore, greater information entropy can make the word easier to be guessed.

4 Task 2: Results Distribution Prediction Model

4.1 Principle Component Analysis (PCA) for Word Attributes

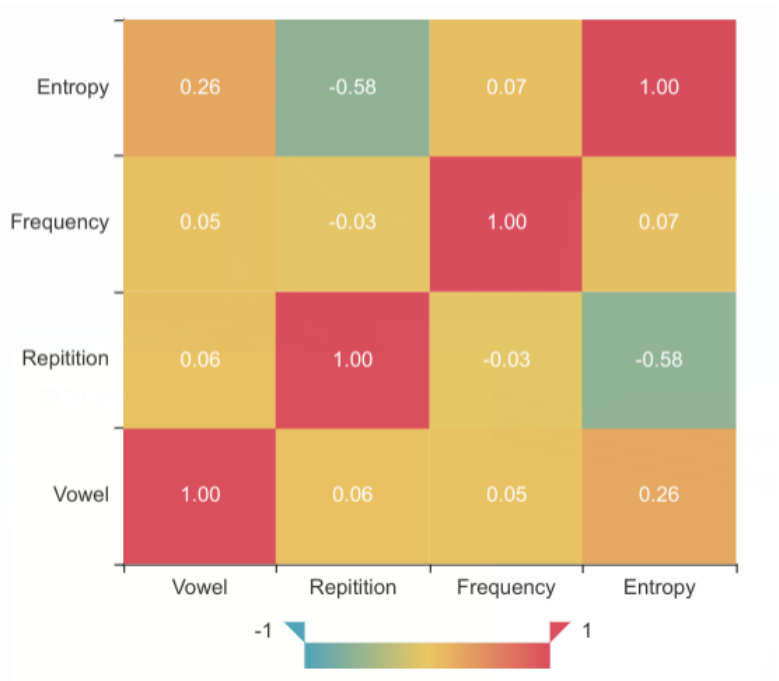


Figure 11: Correlation Matrix for Word Attributes

Observing the correlation matrix constructed by the correlation coefficients between each word attributes in Figure 11, much **correlation** can be noticed between word attributes. For example, the appearance of overlapped letters and numbers of vowels affect may each other, and may also have an influence on the entropy.

Based on these discoveries, we came up with **PCA**[8] for two reasons:

1. Convert these highly correlated variables into independent variables.
2. Project the variables onto lower dimensions to simplify the model and make the problem more solvable.

The eigenvalues and corresponding eigenvectors for the correlation matrix were first calculated and sorted. We selected the first three principle components, which had an accumulated contribution of 91.63%.

The three principle components were labeled as Y_1,Y_2 and Y_3, and the detailed information was recorded in Table 2.

Table 2: PCA for Word Attributes

Eigenvalue	Principle Component	Accumulated Contribution
1.63	$0.25X_{vow} - 0.64X_{rep} + 0.13X_{freq} + 0.71X_{entropy}$	40.76%
1.06	$0.80X_{vow} + 0.41X_{rep} + 0.44X_{freq} + 0.02X_{entropy}$	67.33%
0.97	$-0.44X_{vow} - 0.10X_{rep} + 0.89X_{freq} - 0.10X_{entropy}$	91.63%

4.2 Back-Propagation (BP) Neural Network Model

The results distribution is a vector in seven dimension (from one guess to more than 6 guesses), and the game input is the word and the Hard-Mode Player Ratio. The relationship between them is difficult to find directly. BP neural networks are capable of **modeling non-linear relationships** between the input data and the output. This is important in the Wordle **Result Distribution Prediction Model** because the feedback given by the game is based on complex rules and is not directly related to the solution word. BP neural networks can find those complex characteristics by **Back-Propagation**.[9]

4.2.1 Data Pre-process

After using the PCA for all words, the four word attributes can be converted to **three principle components** Y_1, Y_2, Y_3. The input layer of the BP neural network has four neurons: Y_1, Y_2, Y_3, and the Hard-Mode Player Ratio X_{ratio}.

By fitting the discrete distribution of game result with the continuous normal distribution,we can prepare the samples of the output layer that has two neurons: μ and σ, which can be used to generate the equivalent normal distribution result.

4.2.2 Network Structure

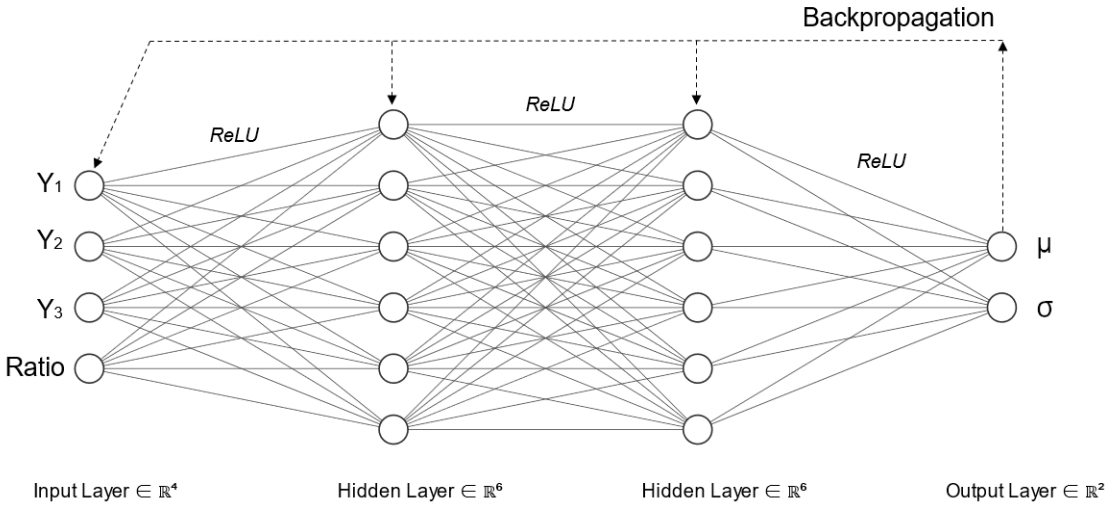


Figure 12: The BP Neural Network Structure

To better fit the sample data set, we tried different study rate, number of layers and number of nodes in hidden layers. We also take the problem of over-fitting into account and finally determined to use the 4-6-6-2 neural network. We choose to use the ReLU activation function : $\mathrm{ReLU}(x) = \begin{cases} x, x > 0 \\ 0, x \leq 0 \end{cases}$ between adjacent linear layers which can improve the nonlinear fitting ability of our model.

We choose to use the *Mean Squared Error* (**MSE**) function as the loss function of our network:

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{8}$$

The Back-Propagation process will use this function to compute the gradient of the error and update the weights and biases of the network by using a gradient descent optimization algorithm.

4.3 Validation of the Model

After training this BP neural network for 10^6 iterations, the total MSE loss has been reduced to about $\mathbf{0.074}$.

To validate the model's robustness, we split the sample data into training set and testing set. The training set contains 80% of the data while the testing set contains 20% of the data. We first train our model on the training set then test it on the testing set and compare the MSE loss of two sets. We randomly choose those two sets for 6 times and the result is shown in figure 13. The difference between the training set and the testing set is about 10% of the training set MSE loss, which is acceptable.

The analysis of the model result will be discussed in the following sections.

Figure 13: MSE Loss of Training and Testing Set

4.4 Results Distribution Prediction for Eerie on March 1, 2023

4.4.1 Input Data

The raw data is shown in table 3. The input data can be summarized as:

1. Word Attributes:

   The model will first convert the attributes of the word to the three principle components Y_1, Y_2, Y_3 before sending it to the BP neural network.

2. Hard-Mode Player Ratio (X_{ratio}):

   Based on the Hard-Mode Player Ratio time series, we applied LSTM to predict the Hard-Mode Player Ratio on March 1, 2023. The final prediction is **0.0969**, with a relatively tolerable RMSE.

Table 3: Attributes of "Eerie"

Attributes	Value	Explanation
Vowel X_{vow}	4	'e' and 'i' are vowels
Repetition X_{rep}	2	'e' appears 3 times
Entropy $X_{entropy}$	2.97	Information entropy
Frequency X_{freq}	772484	Word frequency
Ratio X_{ratio}	0.0969	Prediction of Hard-Mode Player Ratio

4.4.2 Output Result

The BP neural network generated the output as $(5.202626, 1.4668336)$ where the first entry is the mean and the second entry is the variance of the normal distribution.

The distribution from these two parameters is visualized in figure 14.

Try Number	Prob.(%)
1	0.48
2	2.65
3	9.33
4	20.58
5	28.52
6	24.84
≥ 7	13.59

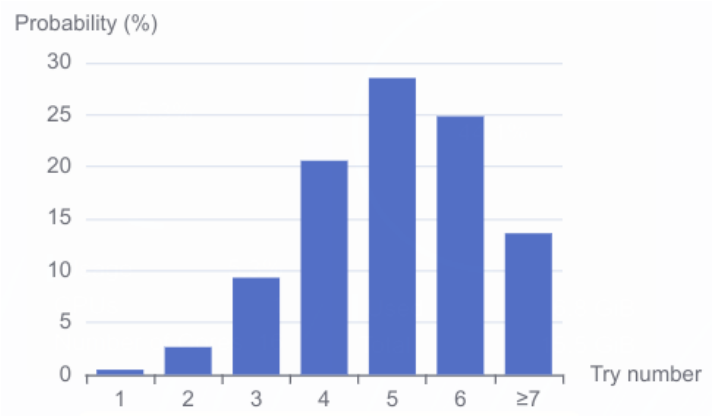


Figure 14: Distribution of game result for "Eerie"

5 Task 3: Word Difficulty Classification and Prediction

5.1 Weighted Distribution Scoring Model

To better measure the difficulty of the word by the result distribution of trial times, it is meaningful to assign different weights to different trial times and calculate weighted trial times for 2 reasons:

1. For most words, most people try 3-5 times to guess, so the trial times between 3-5 usually can't provide valuable information for assessing difficulties.

2. Though relatively few people guess the words for just 1 to 2 times or more than 6 times, those cases are actually most valuable for assessing difficulties.

The goal of **rating information value and assigning weight** for $t_i(i = 1, 2, \cdots, 7)$ reminded us of applying **Entropy Weight Method**. Entropy Weight is found to enhance the function of the attribute with the highest diversity of attribute data (DAD) as well as weaken the function of the attributes with a low DAD in decision-making or evaluation.[10]

Since we have n words and 7 indicators to be weighted, we generated the data matrix:

$$X = \begin{bmatrix} t_{11} & t_{12} & \cdots & t_{17} \\ t_{21} & t_{22} & \cdots & t_{27} \\ \vdots & \vdots & \ddots & \vdots \\ t_{n1} & t_{n2} & \cdots & t_{N7} \end{bmatrix} \tag{9}$$

Then we standardized the data we have and got the standardized matrix $Z = (\tilde{t}_{ij})$,

$$\tilde{t}_{ij} := \frac{t_{ij}}{\sqrt{\sum_{k=1}^{N} t_{kj}^2}} \tag{10}$$

Next, we normalized the data and got the probability matrix $P = (p_{ij})$.

$$p_{ij} := \frac{\tilde{t}_{ij}}{\sum_{k=1}^{N} \tilde{t}_{kj}} \tag{11}$$

Finally, we calculated the information entropy and the entropy weight of t_i $(i = 1, 2, ..., 7)$.

$$e_j := -\frac{1}{\ln N} \cdot \sum_{i=1}^{N} p_{ij} \ln p_{ij} \quad w_j = \frac{1 - e_j}{\sum_{i=1}^{m} (1 - e_j)}, j = 1, 2, ..., 7 \tag{12}$$

The **Weighted Distribution Score** for the j^{th} word is then:

$$s_i = \sum_{j=1}^{7} w_j \, t_{i\,j} \tag{13}$$

Generally, a larger weighted distribution score means a greater word difficulty.

5.2 Word Difficulty Classification Model

5.2.1 Difficulty Level Setting

We set **4** difficulty Levels for Wordle Game, which were respectively **Easy, Medium, Hard, Hell** with increasing difficulty. Therefore, the mission is to classify the words given in data set into 4 **difficulty sets**:

$U_1 = \{\text{words}|\text{difficulty = Easy}\}$ $U_2 = \{\text{words}|\text{difficulty = Medium}\}$

$U_3 = \{\text{words}|\text{difficulty = Hard}\}$ $U_4 = \{\text{words}|\text{difficulty = Hell}\}$

5.2.2 Indicators for Difficulty Classification

We select two indicators for difficulty classification:

(1) **Trial Time Results Mean (μ_i)** (2) **Weighted Distribution Score (s_i)**

Generally, larger μ_i and larger s_i indicate higher word difficulty, we believe that considering the distribution of both of them can make our difficulty classification more comprehensive.

5.2.3 Word Difficulty Embedding Model

The words are embedded into $\mathbb{R}^2$ vectors. For the i^{th} word, the embedded vector is:

$$x_i = (\tilde{\mu_i}, \tilde{s_i}) \tag{14}$$

where $\tilde{\mu_i}$ and $\tilde{s_i}$ are *z-score* standardized μ_i and s_i

5.2.4 Apply Fuzzy C-Means Model (FCM) for Classification

Since the relationship between words and were quite fuzzy, the **boundary between each different difficulty modes should be blurred** to some extent. Therefore, we resorted to **Fuzzy C-Means Model (FCM)** for word difficulty classification.[11]

First, the centers of four difficulty groups were set to be $c_i (i = 1, 2, 3, 4)$, and the detailed values were calculated later.

Then, **Euclidean Distance** was applied for describing the distance between the embedding vectors of words:

$$\text{dist}(c_i, x_i) = \|c_i - x_i\|_2 \tag{15}$$

After that, To characterize the fuzziness, we introduced the concept **membership degree** u_{ij},to describe the probability for the i^{th} word to belong to the j^{th} difficulty level:

$$u_{ij} = P[\text{word}_i \in U_j] \tag{16}$$

Since every word must belongs to one difficulty set, we consequently had:

$$\sum_{j=1}^{4} u_{ij} = 1 \tag{17}$$

Then the **fuzzy partition matrix** was constructed as $U = (u_{ij})$.

Then the target is to minimize the objective function:

$$J(U, C) = \sum_{s=1}^{4} \sum_{i=1}^{n} (u_{is})^m \times \text{dist}(c_i, x_s)^2 \tag{18}$$

where $m = 1.2$ is the arbitrarily-set **fuzzy exponent** that controls how much the clusters overlap with each other, therefore m describes the fuzziness of difficulty boundaries.

Then we applied **Iterative Method** with formulas:

$$u_{ij}^{'} = \frac{1}{\sum_{k=1}^{4} (\frac{\|x_i - c_j\|}{\|x_i - c_k\|})^{\frac{2}{m-1}}}, C_j^{'} = \frac{\sum_{i=1}^{N} u_{ij}^m \times x_i}{\sum_{i=1}^{N} u_{ij}^m} \tag{19}$$

and then the satisfying membership degree matrix $U = (u_{ij})$ and centers C_i (i=1,2,3,4) were ultimately found to let the objective function (18) attain its approximated minimum.

Finally, we classified the words to the difficulty sets to which the words had highest membership degree:

$$\text{level}_i = \arg\max_j u_{ij} \tag{20}$$

5.2.5 Accuracy Analysis

The accuracy for classifying word_i,$(i = 1, 2, ..., n)$ can be described by its membership degree $u_{i\,\text{level}_\text{i}}$.

We calculated the overall accuracy by taking the mean value of accuracy for all the words in the data set:

$$\text{accuracy}_{cls} = \frac{\sum_{i=1}^{N} u_{i\,\text{level}_\text{i}}}{N} \tag{21}$$

After examination, we got that $\text{accuracy}_{cls} = 96.5\%$, which is really satisfying.

5.3 Difficulty Classification for "Eerie"

With the distribution information acquired in Task 2, the Trial Times Mean and Weighted Distribution Score of "Eerie" can be calculated. We standardized μ_{Eerie} and s_{Eerie} with the *z-score* coefficients used in equation (14) and then we got $\tilde{\mu}_{Eerie}$ and $\tilde{s}_{Eerie}$. Therefore, the embedding $\mathbb{R}^2$ vector for "Eerie" is:

$$x_{Eerie} = (\tilde{\mu}_{Eerie}, \tilde{s}_{Eerie}) \tag{22}$$

We applied the **Word Difficulty Classification Model** for "Eerie" and the results were visualized in figure 15.

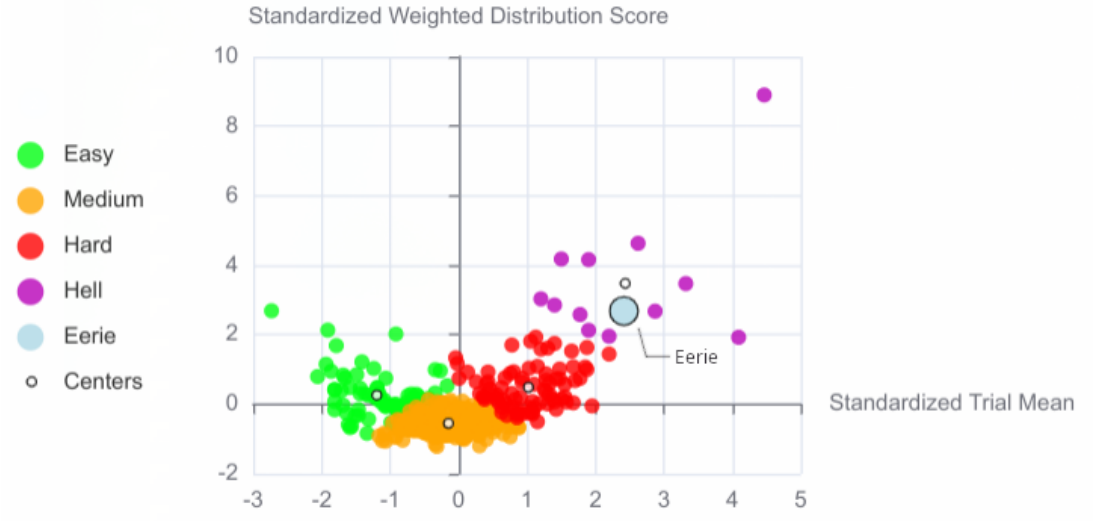


Figure 15: Word Difficulty Classification

It can be clearly seen that "Eerie" is classified into the **Hell Difficulty Category**.

6 Task 4: Data Mining

6.1 Hard-Mode Player Ratio Time Series Analysis

The Hard-Mode Player Ratio is plotted in Figure 16. It can be obviously seen that the Hard-Mode Player Ratio ascends and gradually reaches an equilibrium state, which may have two possible explanations:

1. More and more players have gained experience and strategies and therefore **resort to Hard-Mode for more difficulties and challenges.**
2. As time goes on, **new-coming players decrease** and thus more and more players are experienced players.

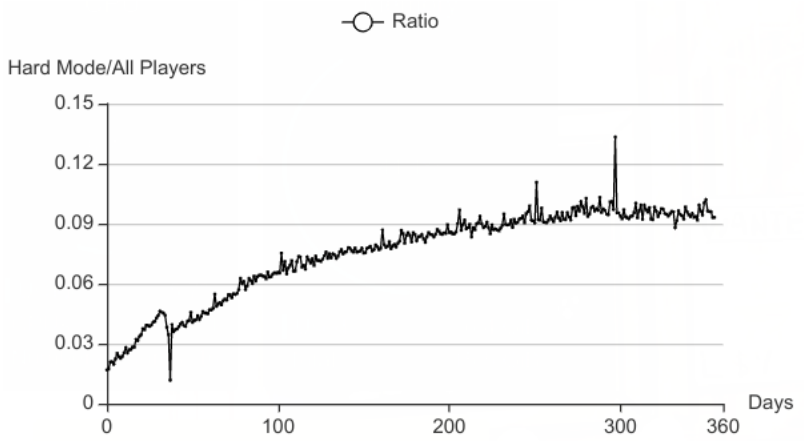


Figure 16: Hard-Mode Player Ratio Trend

6.2 Crucial Properties Influencing the Word Difficulty

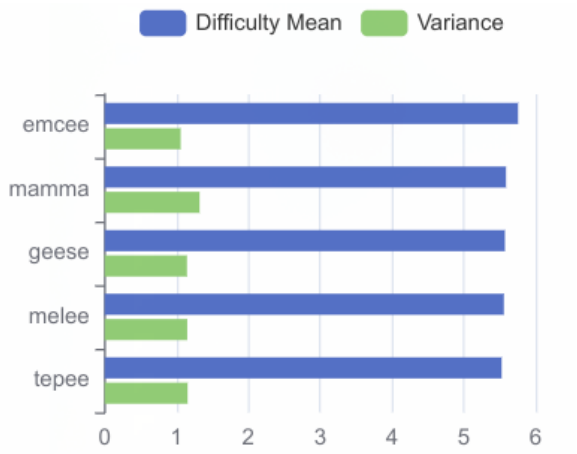


Figure 17: Most Difficult Words

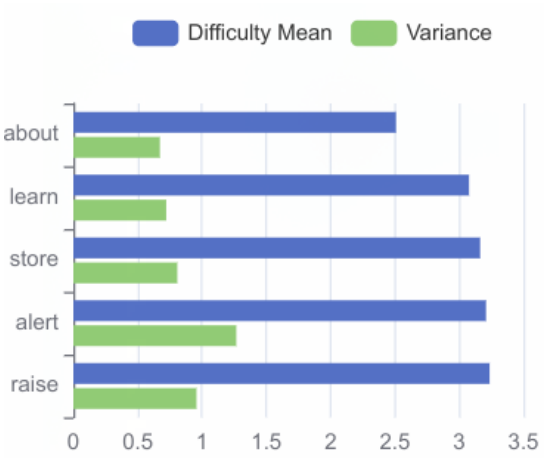


Figure 18: Easiest Words

The most difficult and easiest 5 words are listed in Figure 17 and Figure 18, and the two groups of words share some common patterns.

- **The most difficult words** have several repeated letters especially vowels.
- **The easiest words** have several different vowels and are quite commonly used.

Thus we verify that the **repetition of letters** in a word will increase the word difficulty. Besides, the **variety of vowels** and the **frequency** of a word can lower the word difficulty on the contrary.

7 Sensitivity Analysis

7.1 Sensitivity of Difficulty Classification to the Fuzzy Exponent (M)

The only flexible variable involved in the **Word Difficulty Classification** is the fuzzy exponent (M). The fuzzy exponent is a constant that should be greater than 1.

As is mentioned before, we set M to be 1.2. For the sensitivity analysis part here, we run our algorithms for several times with different M and checked the classification accuracy. The result is shown in Figure 19.

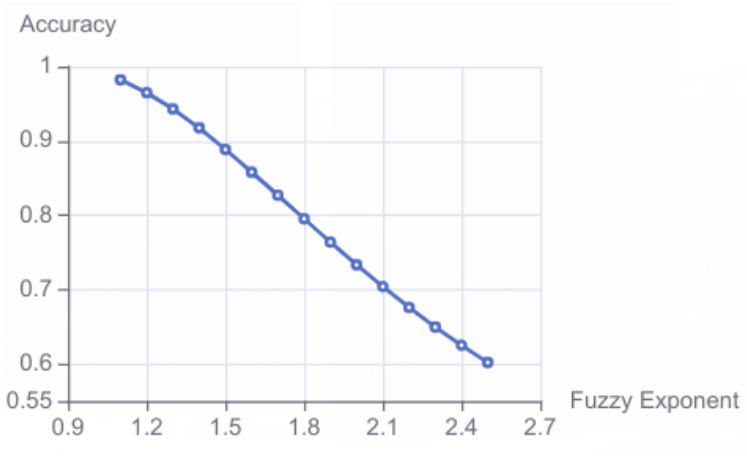


Figure 19: Sensitivity Analysis for Fuzzy Exponent and Classification Accuracy

It turned out that the accuracy of classification is sensitive to the fuzzy exponent (M) to some extent, and it is better to keep M within the range from 1.1 to 1.5 so that the classification accuracy can be fairly satisfying.

In addition, it is worth mentioning that despite the variation of classification accuracy due to changing fuzzy exponent, the predicted difficulty for "Eerie" is **always the hardest "Hell Level"**, which **shows the robustness** to our model and results to some extent.

7.2 Sensitivity of BP Neural Network to Inputs

The inputs contain the three principle components and ratio X_{ratio}. We choose one sample word “manly” to test the sensitivity of the model. We give ± 0.1 change to each input and calculate the floating range of the result Δ_μ and Δ_σ.

The result shows that the output μ is most sensitive to Y_1 while the output σ is most sensitive to X_{ratio}. However, the changes are relatively small to the result, therefore in general the model passes the sensitivity analysis and **shows the robustness**.

Figure 20: Δ_σ

Figure 21: Δ_μ

8 Strength and Weakness

8.1 Strength

- Our model has both the capability to explain the variation of results number variation by curve fitting as well as the ability to predict the results number by LSTM.
- Our model converts 4 word attributes to 3 **relatively independent** variables to improve the Machine Learning Learning Effects.
- Our model considers both the results distribution attributes and the information weight of each trial time result to give a **more comprehensive difficulty classification**.
- Our model allows the **difficulty level boundary to be blurred**, which takes the fuzzy relationship between words into consideration.
- Our model achieves great accuracy and can **stand the tests of simulation**.
- Our model features great **robustness** so that our results and conclusions are not sensitive to changes.

8.2 Weakness

- Our model simplifies the results distribution into Gaussian Distribution by Central Limit Theorem. Though the data **manifests strong Gaussian Distribution properties indeed**, the arbitrary substitution can **still leaves out some details** and leads to inaccuracy to some degree.
- Our model uses the BP neural network that has two hidden layers and 6 nodes each hidden layer. Although the trained result **works well with the 20% testing set**, it may still **over-fit the data** due to **the complexity of the characteristics and the scarce** of the sample data to some extent.

9 Future Updates for the Model

9.1 Taking Players' Strategies into Considerations

In this study, the individual difference between players' strategies are overlooked, while some players may first input several fixed words every time to test the letter information, and others may just simply guess the word every time based on the former trials. The overlook of players' strategies will certainly effect our predictions of game results distribution to some degree.

9.2 More Scientific Analysis for Game Results Distribution Description

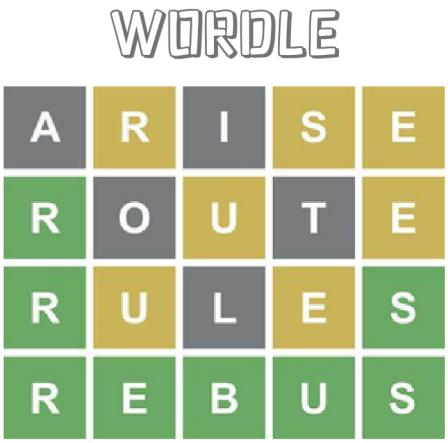
In this model, we simplify the game results distribution for all the words as Gaussian Distribution, which overlooks the influence each specific word might impose on. Therefore, to reveal a more accurate relation between game results and word attributes and forecast a better prediction, the improvement of results distribution simulation is indispensable.

9.3 Apply the Neural Network that can better Extract Features

In this model, the Back-Propagation Neural Network may not be powerful enough for extracting features sometimes, and the arbitrary machine learning may lead to over-fit problems, which means that the results may seem perfect for the data in the training set, but may not work well when facing new data for prediction, and the prevention for less over-fit will sacrifice the accuracy for prediction to some extent. Therefore, it is imperative that a better Neural Network for extracting features should be applied for improvement.

10 Conclusion

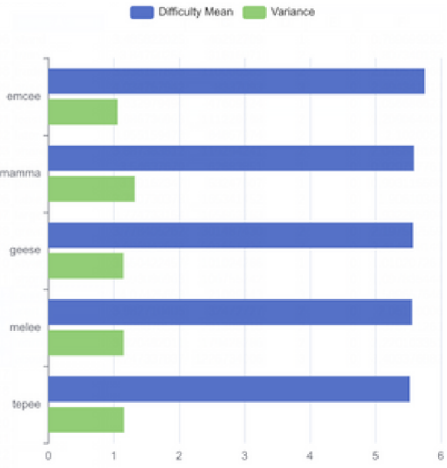
To study the relations between word attributes, players number and results distribution, we propose Word-attributes and Players Game Model (WPGM) to quantify their abstract relations and apply them for future prediction. Our model can explain and predict the results number with the aid of curve fitting and LSTM, depict the correlation between various variables with statistic calculations, predict the results distribution by BP Neural Network, and classify the word difficulty by FCM. We also further explore the trend of Hard-Mode Player Ratio and the common properties easy and difficult words share in data mining. We not only discern the features but also try to analyze and explain them. Our Model successfully passes sensitivity analysis and manifests great robustness. Despite the slight weakness and limitations, we believe that model can present satisfying results to some extent. In the future, we seek for more complicated mathematical modeling and advanced machine learning technology to improve our model for the better.



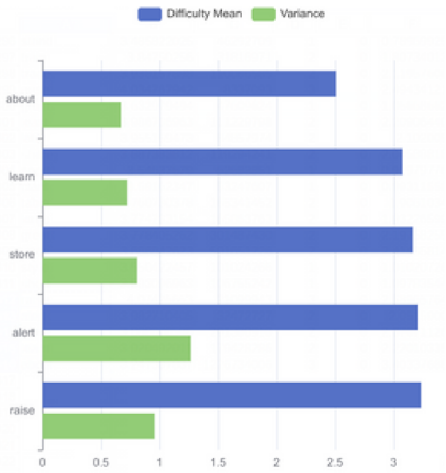
TO: PUZZLE EDITOR OF THE NEW YORK TIMES
FROM: TEAM 2311153
DATE: FEBRUARY 20, 2023

Dear Puzzle Editor,

Greetings! The game Wordle has been popular throughout the world. As a fan of Wordle, it is our pleasure to share with you some interesting results of this puzzle game based on the data from your website. In this letter, we will cover the variation of number of players within time, how word attributes affect players' trails of guessing, our own word difficulty classification and other useful conclusions. We sincerely hope that our analysis can bring you some insights into this game, and we also hope Wordle to expand its influence and bring fun to more people.



Most Difficult Words



Easiest Words

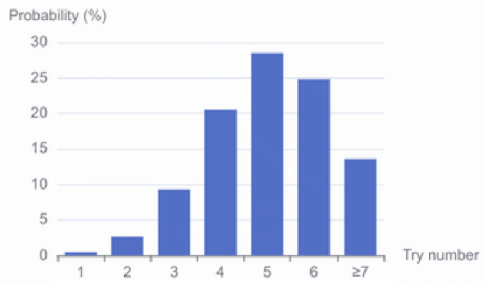
To begin with, we studied the trend of the number of players and made predictions to it. We found that the players can be grouped into two parts. The majority of them are long-time players who are leaving the game or playing it at lower frequency, while the rest are newcomers that grows gradually and continuously. Since the population and speed of gamers leaving the game are both greater than those coming to it, the overall trend of population presents a declining trend. Therefore, we suggest that the New York Times should consider how to retain players and keep their passion for it. For example, we can make the UI more attractive, develop other difficulty modes or even broad the length of words to offer players various experiences.

What's more, we set four attributes of words as indicators to depict how they affect the difficulty of guessing, which are the number of vowels, the number of repeated letters, the frequency of the word in daily use and the information entropy (the ability of the word to provide information). In fact, these four indicators have correlations themselves. For example, we found that the repetition of letters in a word is negatively correlated to its entropy, while the number of vowels has a positive relation with its frequency in use. By math modeling, we managed to deduce their contributions to the difficulty of guessing the word. The most inspiring conclusions are listed as follows:

1. More repetition in letters indicates add difficulty to guessing.

2. Distinct vowels in a word provides more information, thus reducing the trials to guess the word out.

3. Frequency of word use do not have a significant impact on the difficulty compare with the composition and structure of it.



Predicted Distribution of "Eerie"

In addition, we found that the attributes of words hardly affect the percentage of scores reported that were played in Hard Mode. It can be naturally explained by the fact that players won't be influenced by words in-game when selecting game modes before-game.

We further classified the word difficulty into four degrees: Easy, Medium, Hard and Hell. As is expected, the most difficult words are ***emcee***, ***mamma***, ***geese***, ***melee*** and ***tepee***, which all had several repeated letters especially vowels. On the other hand, the easiest words are ***about***, ***learn***, ***store***, ***alert*** and ***raise***, which all possess several different vowels.

We assume that the visualized difficulty of words can be applied in the game Wordle. The New York Times can rate the difficulty by scores and show it to players together with the solution. The scores may help gamers build confidence in the long run. They will not feel defeated when they come across difficult words and get more satisfaction when they guess them correctly in only a few turns. The score also enables the players to evaluate their improvement in their skills, which also serves as a big motivation to keep playing it.

Using the model we develop, we label the difficulty of the word "***Eerie***" as "Hell" and predicted the result distribution on March, 1. This is a challenge for us but also a chance to show you how powerful our models are. However, chances are that the result distribution will be "eerie" on that day since many of us will try this word as the first attempt.

In a word, Wordle is not only a puzzle game but also an excellent educational tool to help people know better of words. We genuinely hope that the New Yorks Times can develop it by heart instead of utilizing it only to attract people for newspaper subscriptions.

Warm regards,

TEAM 2311153

References

[1] "Wordle." Wikipedia, Wikimedia Foundation, en.wikipedia.org/wiki/Wordle.

[2] Knight, Ste. “How and Why to Play Wordle in Hard Mode.” Make Use Of, https://www.makeuseof.com/how-to-play-wordle-hard-mode-and-why.

[3] *Wordle*[Online image]. https://sites.google.com/site/eskolatics/wordle.

[4] Jadranda D.T.(2018). Applicability of Newton’s law of cooling in monetary economics. Physica A: Statistical Mechanics and its Applications, vol 494, 209-217.

[5] Rachel T. English Word Frequency, Kaggle, https://www.kaggle.com/datasets/rtatman/english-word-frequency.

[6] Sanderson. G (2022). Solving Wordle using you keep your information strcak theory. https://www.3bluelbrown.com /lessons /wordle.

[7] "Central Limit Theorem." Wikipedia, Wikimedia Foundation, en.wikipedia.org/wiki/Central_limit_theorem.

[8] "Principal Component Analysis." Wikipedia, Wikimedia Foundation, en.wikipedia.org/wiki/Principal_component_analysis.

[9] Hao L.(2022), Optimal selection of control parameters for automatic machining based on BP neural network, Energy Reports,vol 8, 7016-24.

[10] Chen, P. (2021). Effects of the entropy weight on TOPSIS. Expert Systems with Applications, 168, 114186.

[11] Karim E. M.(2022) Optimal Entropy Genetic Fuzzy-C-Means SMOTE (OEGFCM-SMOTE),Knowledge-Based Systems,Vol.262.