Problem Chosen	**2024**	**Team Control Number**
C	**MCM/ICM** **Summary Sheet**	2304217

Mathematical Secrets behind Wordle Reported Results

Summary

Wordle is a compelling online word-guessing game that has recently taken the Internet by storm. Based on the reported results on Twitter, the main emphasis of the paper is placed on the scientific mathematical models to predict the time-varying quantity of reported results and their distribution. Also, we come up with a systematic classification model to evaluate the difficulty of any specific word.

In TASK 1, firstly, the **ARIMA-GRU** model is proposed to predict the number of reported results. We combine the ARIMA model, which can predict well about the linear part, and the GRU model, the opposite of the ARIMA model, which can only capture the information of the non-linear part. The prediction interval on March 1, 2023, is $[9773.514, 10590.548]$. As for the second goal, we are required to find out whether the word attributes can affect the proportion of reported results playing in hard mode, we determine some specific word attributes that may cause effects and use the **Pearson correlation coefficient** to check their correlation. It turns out that word attributes have nothing to do with the hard mode proportion.

In TASK 2, we develop a **Modified Normal Distribution Model** made up of multiple **linear regression models**. We innovatively divide the distribution of the results into normal distribution part and the non-normal distribution part. Firstly, we train two linear regression models I, to predict the parameters of a normal distribution,i.e. μ and σ. Secondly, we train 7 linear regression models II to predict the non-normal part of different tries — the error between the percentage derived from the predicted normal distribution and its true percentage. As for the input of the models, we use **Pearson correlation coefficient** to screen out all the variables possibly related to its output. In addition, we use R^2 and $R^2_{adjusted}$ to evaluate the goodness of fit which reaches 0.774 and 0.762 in the end. Finally, we give the predicted results distribution of the word $EERIE$ on March 1, 2023. Its percentage of 1 try, 2 tries, 3 tries, 4 tries, 5 tries, 6 tries, and 7 or more tries will respectively be 0.70%, 1.21%, 8.95%, 25.27%, 33.48%, 23.62% and 6.76%.

In TASK 3, with **Pearson correlation coefficient**, we filter out factors that have slight impact on the difficulty level. Besides attributes of solution words, we add AI Judging as a new indicator by developing **two AI wordle robots**, **HumanoidBot**(based on word frequency) and **EntroyBot**(based on information theory), to classify the difficulty level of wordles. The first robot is designed to emulate the human mind, and the second robot is to assess model difficulty in a scientific respective. Then we verify the normality of the average-try number and set difficulty levels according to the normal distribution. Finally, we develop a difficulty-classification model using **logistic regression**. Our **classification model** is robust and receives high accuracy.

Finally, to gain a deep understanding of our model, we verify model robustness in many cases and analyze the strengths and weaknesses of our model. When the initial input is randomly distorted, the final convergence distribution of the model has little difference. Moreover, studies of reported results suggest multiple interesting rules.

Keywords: ARIMA-GRU model, Modified Normal Distribution Model, AI wordle-Bot

Contents

1 Introduction

1.1 Problem Background

Wordle is a popular puzzle currently offered daily by the New York Times. Players try to solve the puzzle by guessing a five-letter word in six tries or less, receiving feedback with every guess. Many (but not all) users report their scores on Twitter. The teams are provided with the following tasks:

- Develop a model to explain the variation of reported results and use the model to predict the prediction interval for the number of reported results on March 1, 2023.
- Find out whether there is a relation between the attributes of the word and the percentage of reported results that were played in hard mode.
- Develop a model that predicts the distribution of the reported results.
- Develop and summarize a model to classify solution words by difficulty.

1.2 Our work

1. Firstly, we establish a **ARIMA-GRU model** to explain and predict the variation of the reported result number, which includes an ARIMA model to predict the linear part and a GRU model to predict the non-linear part.
2. We select some word attributes that may influence the proportion of reported results played in hard mode. We calculate the r between them to check their correlation.
3. To predict the results distribution, we build the **Modified Normal distribution Model**, thinking of the results distribution as a combination of a normal distribution and non-normal error. We use two separate **linear regression models** to predict them respectively.
4. Furthermore, we develop **two AI wordle robot–HumanoidBot**(based on word frequency) and **EntroyBot**(based on information theory) to predict the difficulty of wordle.
5. In addition, we develop a model using **logistic regression** to classify wordle difficulty.
6. Last but not least, We conduct **rubustness analysis** to verify the accuracy, stability and robustness of the model.

2 Preparation of the Models

2.1 Assumptions and Justifications

- **Assumption 1**: People who post their game results always tell the truth and don't cheat the game.

- **Assumption 2**: We assume that the population's ability to guess words is relatively stable or in a calculable relationship with time.

- **Assumption 3**: The number of users of the game will not have a sudden change, regardless of the impact of factors such as the running state of the game itself.

2.2 Notations

The primary notations used in this paper are listed in Table 1.

Table 1: Notations

Symbol	Definition
r	Pearson correlation coefficient
R^2	Coefficient of determination
$MAPE$	Mean absolute percentage error
DR	Difficulty ratio
AT	Average Tries
p_i	The proportion of i$^{\text{th}}$ tries of all reported results
N	Number of reported results
H	Number of reported results played in hard mode

2.3 Data Preprocess

I Handling of Word Outliers

Through observing the data, we find that some words are misspelled. Our correction is as follows

Table 2: Word Cleaning

original word	**modified word**
clen	clean
naiive	naive
rprobe	probe
marxh	march
tash	stash

II Handling of Result-Number Outliers

We draw a graph about the ratio of numbers in hard mode to that of reported results in the figure below.

$$ratio = \frac{H}{N} \tag{1}$$

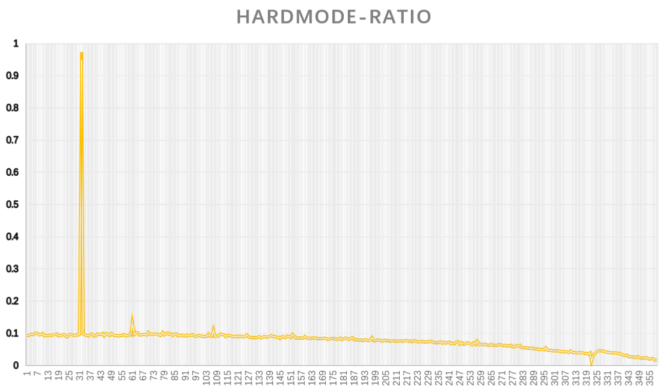


Figure 1: Hard Mode Ratio

We can see that almost all the ratios are below 0.2, however, the ratio on 2022-11-30 is close to 1, so we consider it to be abnormal. Compared with the surrounding values, we change the number of reported results on 2022-11-30 from 2569 to 25690.

3 Task 1: Reported Results Prediction Model

3.1 Prediction on the Number of Reported Results

3.1.1 Initial Observation of the Data

According to the question, we need to develop a model to predict and analyze the variation in the number of reports and use it to make a prediction on March 1, 2023.

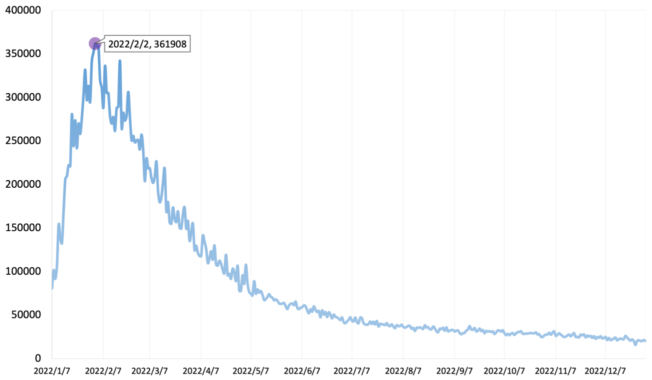


Figure 2: Number of Daily Reported Results

As is clearly illustrated in the chart above, the number of daily reports soars from 8000 to 306356 in the beginning and reaches its peak on February 22, 2022. Then, the number gradually decreases till December 31, 2022. After that, the number is relatively stable.

To predict the number, we decide to use the Autoregressive integrated moving average(ARIMA) model and GRU model, two popular models in time series forecasting. More importantly, we propose **the ARIMA-GRU combined model** to make the prediction more accurate.

3.1.2 ARIMA Linear Prediction Model

The ARIMA model is known as the Autoregressive Integrated Moving Average Model, which uses differencing to convert a non-stationary time series into a stationary one, and then predict future values from historical data. This model is a combination of autoregressive (AR) and moving average (MA), which can transform a non-stationary time series into a stationary time series, and then regress the lagged values of the dependent variable, the present and lagged values of the random error term to the model established.[1]The formula is listed below:

$$y_t = c + \sum_{i=1}^{p} \phi_i y_{t-i} + \varepsilon_t + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} \tag{2}$$

where y_t is the current value; c is the constant term;ϕ_i is the autocorrelation coefficient; ε_i is the error term; θ_i is the coefficient of error term.

3.1.3 GRU Nonlinear Prediction Model

Gated Recurrent Unit(GRU), a modified recurrent neural network, can handle the problem of long-range dependencies. GRU consists of the following steps.

step1: Update Gate

The update gate helps the model to determine how much of the past information (from previous time steps) needs to be passed along to the future. [2]

step2: Reset Gate

Essentially, reset gate is used to decide how much of the past information to forget.

step3: Current memory content

In this step, we introduce new memory content that uses the reset gate to store the relevant information from the past.

step4: Final memory at current time step

In this step, GRU needs to calculate the vector which holds information for the current unit and passes it down to the network. In this process, an update gate is needed. It determines what to collect from the current memory content and what from the previous steps.

3.1.4 ARIMA-GRU Model

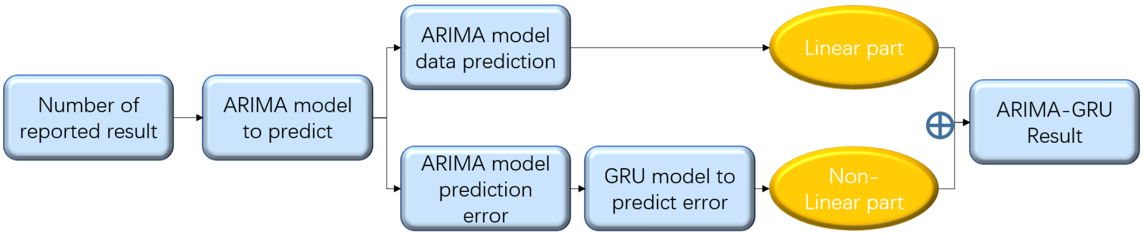

Number of reported result
ARIMA model to predict
ARIMA model data prediction
ARIMA model prediction error
GRU model to predict error
Linear part
Non-Linear part
ARIMA-GRU Result

Figure 3: Flowchart of ARIMA-GRU Model

Because Arima is more suitable for predicting smooth sequences, which leads to the predicted value we get having a small fluctuation, and there is a large gap in the true value of the test set. We take the difference between the ARIMA-predicted result and the real value as the input, use the GRU model to predict the interpolation between the two, and then add the more linear result predicted by ARIMA and the more volatile result predicted by GRU together to get the final prediction result.

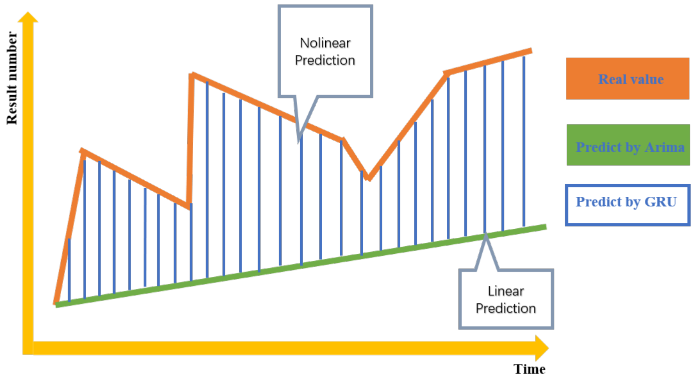
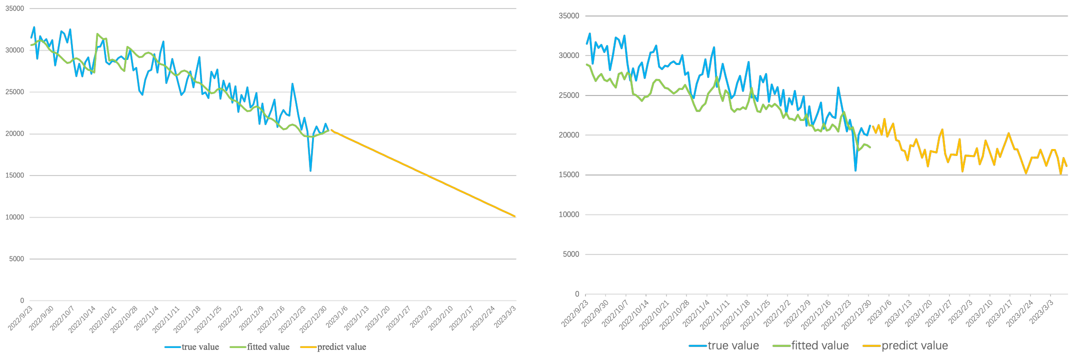


Figure 4: Concept of ARIMA-GRU Model

3.1.5 Prediction Results

The ARIMA model prediction and the GRU model prediction of the number of reported results are performed respectively. The results are obtained as shown below.



(a) ARIMA Model:Report Number Prediction (b) GRU Model:Report Number Prediction

Figure 5: The Prediction Result of Original Models

- **ARIMA Model Prediction Results**

For this problem, we use the one-step prediction method to predict the report number, i.e.,the first i-1 data are used as the training set when predicting the number of reported results at the ith, while the ith sample is added to the training set when predicting the i+1th sample.

The prediction results of the ARIMA model are shown in figure 5(a). From Figure 5(a), we can see that the ARIMA model is not very accurate and realistic in predicting the future number of reported results, which is almost linear to time **with pretty slight fluctuation**. However, the ARIMA model is still able to **capture the trend** of report number, which means that it can predict well about the linear part of the number of daily reports.

- **GRU Model Prediction Results**

The results obtained by using the GRU model are shown in figure 5(b). It can be found that the GRU model can **capture the fluctuation** of the number of reported results well, and it has **slightly improved the prediction accuracy** of the number compared to the ARIMA model with a larger difference between fitted value and real value. Therefore, we can conclude that the GRU model is able to capture the information of the non-linear part of the number relatively well.

- **ARIMA-GRU Model Prediction Results**

To further improve the prediction accuracy, we adopt **an ARIMA-GRU model** for prediction to synthesize the linear and non-linear components of the number of reported results.

For this purpose, firstly, we based on the ARIMA prediction results and the real value of the reported results number to get the residual sequence of the report number, which is used as the expected output of the GRU model; secondly, use the data after reconstructing with the optimal order as the GRU input; thirdly, input the training set to the GRU for learning the model and predicting the residual series test set to obtain the ARIMA residual series prediction; finally, the ARIMA and GRU model prediction results are summed to obtain the final prediction result on March 1, 2023. The prediction results are shown in **Figure 6**.

The predicted number of reported results on prediction interval for the number of reported results on March 1, 2023 is approximately antextbf9993.

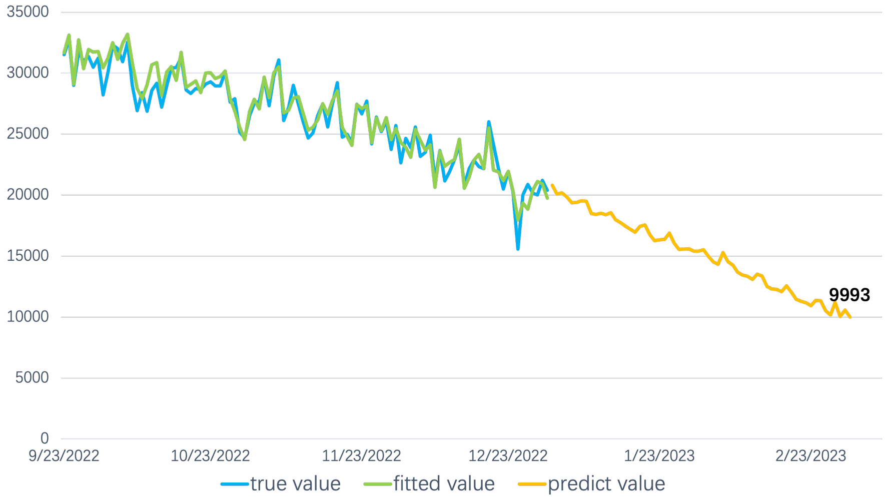


Figure 6: ARIMA-GRU Model: Report Number Prediction Chart

- **Calculate prediction interval**

Since we find that the difficulty of wordle is changing periodically, it is difficult to determine exactly the difficulty of guessing words on March 1^{st}. Here we take the predicted data from Feb 24^{th} to March 9^{th} and calculate the normal distribution of these data(we have verified that the data is in line with normal distribution using the Kolmogorov-Smirnov test). When alpha is 0.95, we calculate the confidence interval as **[9773.514, 10590.548]**.

When we change the value of alpha, the confidence interval will change as shown in the figure below:

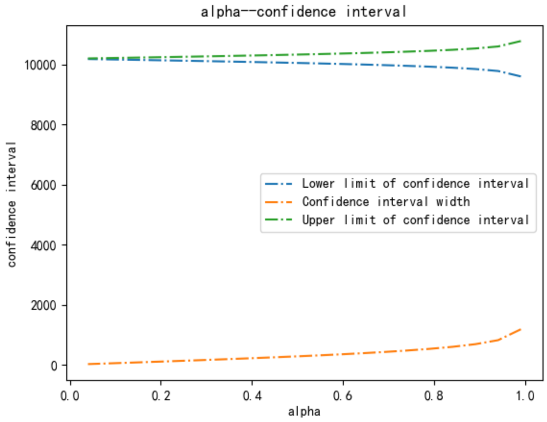


Figure 7: Alpha-Confidence Relation

- **Comparison of Prediction Results**

In order to verify the superiority of the ARIMA-GRU prediction model, we use Mean Absolute Percent Error (MAPE) and coefficient of determination(R^2) as the model performance evaluation indexes. The evaluation results are obtained in the table below.

Table 3: Model Accuracy

Model	MAPE	R^2
ARIMA	0.242	0.534
GRU	0.345	0.329
ARIMA-GRU	0.133	0.965

- **Explanation of the Prediction Results**

We can see that after 2022-12-31, the number of reported results has shown a decline in general, which we believe is due to the decline in people's enthusiasm for Wordle games.

3.2 Relation Between Word and Percentage of Hard Mode

3.2.1 Choice of Word Attributes

Because "wordle" is a riddle about a word, the attributes of a word that may affect the game should have something to do with the difficulty of the game. In this case, we first think of using the average number of tries AT to indicate the difficulty of a word.

$$AT = \sum_{7}^{i=1} p_i i \tag{3}$$

Then we can derive the average number of tries of all words is 4.19. To describe people's performance in the game better, we further use 4 tries as a boundary to define the difficulty ratio DR[3] of a word.

$$DR = \frac{\sum_{7}^{i=5} p_i N}{\sum_{4}^{i=1} p_i N} - 1 \tag{4}$$

We randomly choose some words and show their AT and DR in the following image.

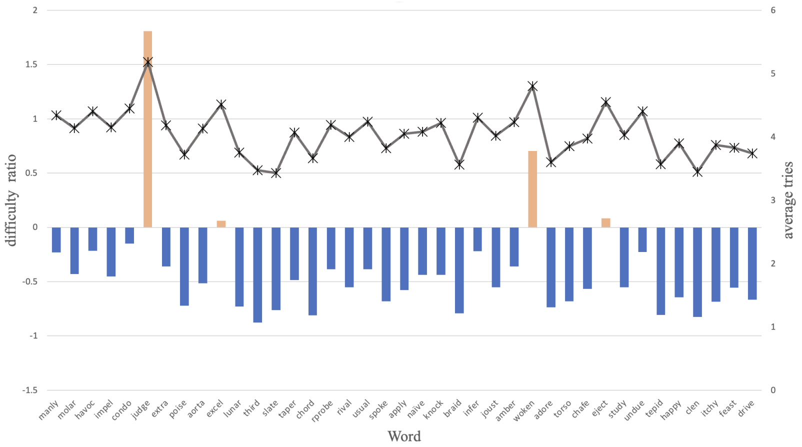


Figure 8: The DT and AT of Some Random Words

From the image, we can see that DR divides the words into two parts intuitively. Words with a DR above 0 can be considered as difficult words to solve and words with a DR below 0 can be considered relatively easy words. Based on this idea, We sort all the words in the data by DR. The results are shown in the image below.

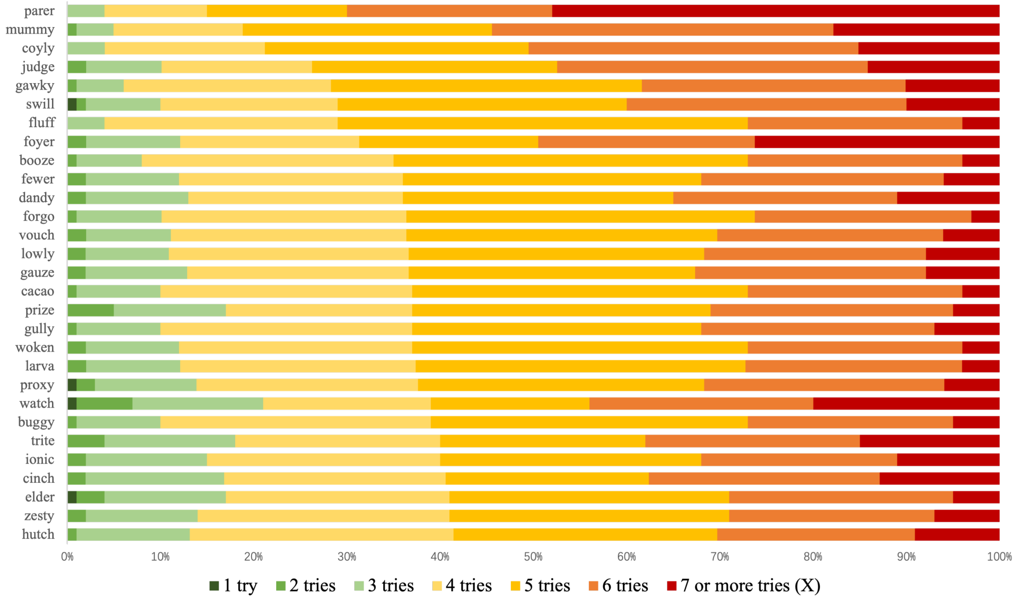


Figure 9: The Top 30 Hardest Words

We can see that words like *parer, mummy, coyly, swill, fluff, booze*, which include duplicate letters made up 63.3% of the top 30 hardest words. Words like *mummy, coyly, judge, gawky, foyer, booze*, which include letters of appearing frequency lower than or equal to 2% in texts, account for 53.3% of these words. Words like *parer, coyly, gawky, swill, forgo*, whose $Zipf frequency$ is less than or equal to 3, account for 36.67 of these words. In particular, $Zipf frequency$ stands for the base-10 logarithm of the number of times it appears per billion words. Words like *mummy, swill, fewer, dandy, vouch, prize* which have at least three common words similar to them in spelling, make up 30% out of these words. Words like *mummy, coyly, gawky, swill, fluff, dandy*, which have only one vowel, make up 50% of these words. In conclusion, we choose **duplicate letters, letter frequency, Zipf frequency, similar words, vowel numbers** as word attributes that may affect the percentage of reported results that are played in hard mode.

3.2.2 Calculation of Word A wordibutes

- **Duplicate letters:** We counted the number of repeated letters in each word.
- **Letter frequency:** We sum up the base-10 logarithm of each letter's appearing frequency to get a number as the word's letter frequency.[4]
- **Zipf frequency:** We use the Python library $wordfreq$ to obtain the Zipf frequency of each word.
- **Similarity:** If words A and B have 4 same letters exactly in the same place, then we call that B is the similar word of A, and vice versa. We use $wordfreq$ to find out 2500 words of length 5 with the highest Zipf frequency, and then count the number of similar words in these common words corresponding to each word in the given data.
- **vowel numbers:** We count the number of "a, e, i, o, u" appearing in each word as its vowel number.

3.2.3 Verify the Correlation

To verify the correlation, we introduce the Pearson Correlation coefficient r, which can describe a statistical relationship between two variables. The formula is shown below.

$$r=\frac{\sum_{i=1}^{n}\left(X_i-\bar{X}\right)\left(Y_i-\bar{Y}\right)}{\sqrt{\sum_{i=1}^{n}\left(X_i-\bar{X}\right)^2}\sqrt{\sum_{i=1}^{n}\left(Y_i-\bar{Y}\right)^2}} \tag{5}$$

After bringing the word attributes and the percentage of hard mode into the equation, we get the following results.

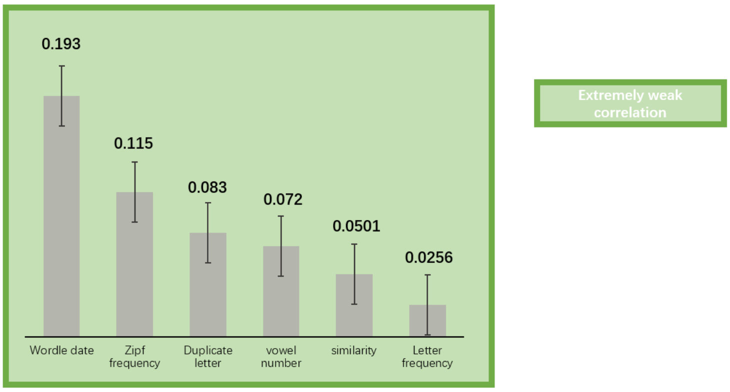


Figure 10: $|r|$ of Word Attributes and Hard Mode Proportion

We can see that the $|r|$ of these words are all smaller than 0.2, which means these word attributes have **little correlation** with the hard mode proportion.

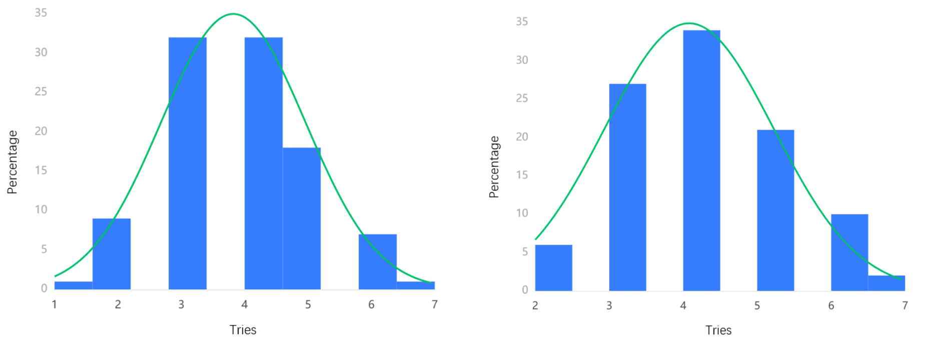
Moreover, we trained a linear regression model to test the correlation between these word attributes and hard mode percentage. Its coefficient of determination R^2 turns out to be -0.1824, which further verifies that the word attributes have nothing to do with the proportion of hard mode.

We believe that the reason why the difficulty of words has nothing to do with the hard mode ratio is that when we choose the play mode, we do not know whether the word of the day is simple or difficult.

4 Task 2: Reported Results Distribution Prediction Model

4.1 Distribution Inference And Establishment of Model

I. Distribution Inference After a preliminary observation of the distribution of the reported results, it's easy to see that results of 3, 4, or 5 tries always made up most of the results, and results of 1, 2, 6, or 7 tries always made up a much smaller proportion, which exactly corresponds to the feature of normal distribution. Moreover, we check the histogram of the distribution of the reported results, and it looks good fit with the curve of normal distribution. We have randomly chosen two histograms and shown them below.



(a) Results Distribution on 2022-1-16 (b) Results Distribution on 2022-5-30

Figure 11: Histogram of Randomly Chosen Results Distribution

II. Establishment of the Modified Normal Distribution Model Although from the histogram above, the distribution of the results seems a good fit with the normal distribution, there is still room for improvement. We choose to follow the idea of the previous task — first, use a **linear regression model** to predict the μ **and** σ of future results distribution, then use another 7 linear regressions to predict the **error** of each tries' percentage and its corresponding percentage of the predicted normal distribution derived from the first linear regression model.

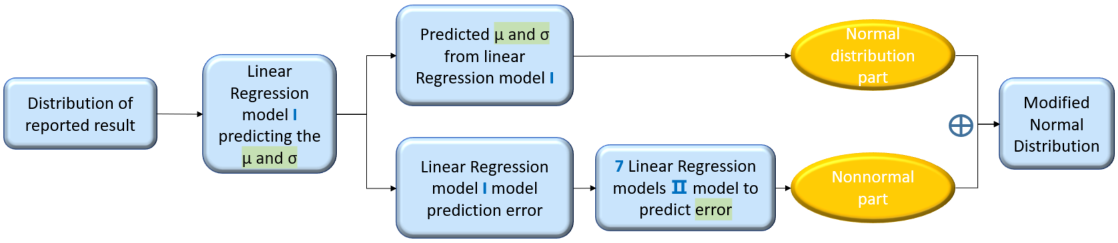

Distribution of reported result
Linear Regression model I predicting the μ and σ
Predicted μ and σ from linear Regression model I
Linear Regression model I model prediction error
7 Linear Regression models II model to predict error
Normal distribution part
Nonnormal part
⊕
Modified Normal Distribution

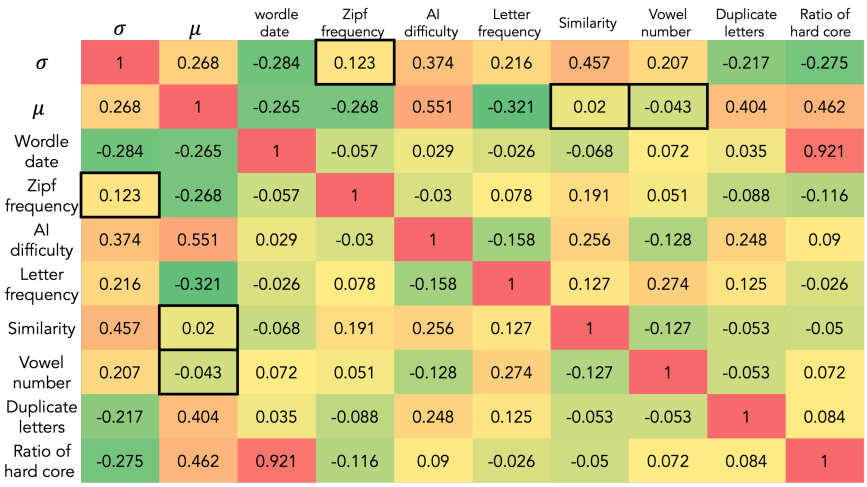
Figure 12: Flow Chart of Modified Normal Distribution Linear Regression Model

4.2 Prediction of μ and σ

First, we need to get the normal distribution part by **predicting its μ and σ**, which describes what a normal distribution looks like.

4.2.1 Factors Related to μ and σ

We calculate the correlation coefficients of the data two by two.



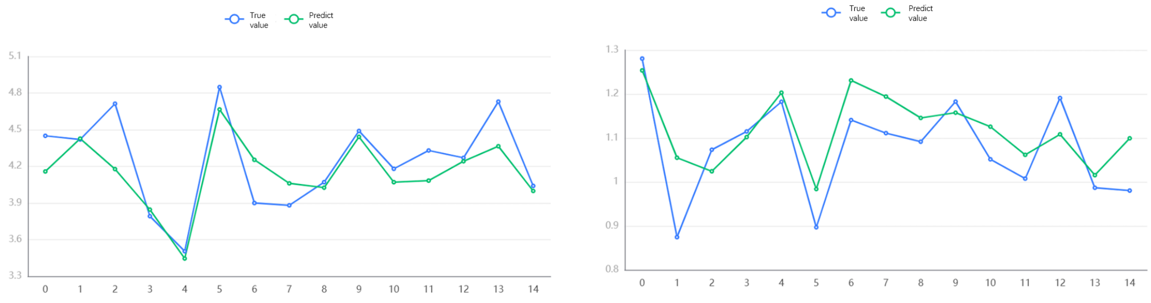
	σ	μ	wordle date	Zipf frequency	AI difficulty	Letter frequency	Similarity	Vowel number	Duplicate letters	Ratio of hard core
σ	1	0.268	-0.284	0.123	0.374	0.216	0.457	0.207	-0.217	-0.275
μ	0.268	1	-0.265	-0.268	0.551	-0.321	0.02	-0.043	0.404	0.462
Wordle date	-0.284	-0.265	1	-0.057	0.029	-0.026	-0.068	0.072	0.035	0.921
Zipf frequency	0.123	-0.268	-0.057	1	-0.03	0.078	0.191	0.051	-0.088	-0.116
AI difficulty	0.374	0.551	0.029	-0.03	1	-0.158	0.256	-0.128	0.248	0.09
Letter frequency	0.216	-0.321	-0.026	0.078	-0.158	1	0.127	0.274	0.125	-0.026
Similarity	0.457	0.02	-0.068	0.191	0.256	0.127	1	-0.127	-0.053	-0.05
Vowel number	0.207	-0.043	0.072	0.051	-0.128	0.274	-0.127	1	-0.053	0.072
Duplicate letters	-0.217	0.404	0.035	-0.088	0.248	0.125	-0.053	-0.053	1	0.084
Ratio of hard core	-0.275	0.462	0.921	-0.116	0.09	-0.026	-0.05	0.072	0.084	1

Figure 13: Correlation Coefficient Chart for μ and σ

As is vividly depicted in the chart above, the magnitudes of r are determined by color shades. Therefore, μ is weakly related to two word attributes,i.e.similarity and vowel number.σ has little correlation with word frequency. Therefore, we only focus on the respective relations of σ and μ with the remaining factors.

4.2.2 Linear Regression Model I of μ and σ

We use 70% of the past data of derived factors as input, μ and σ of corresponding results distributions from the past as the output to train the first linear regression model to predict the future μ and σ respectively. The left 30% of data is used as the test set. The comparison of the predicted values and true values of the test set is shown below.

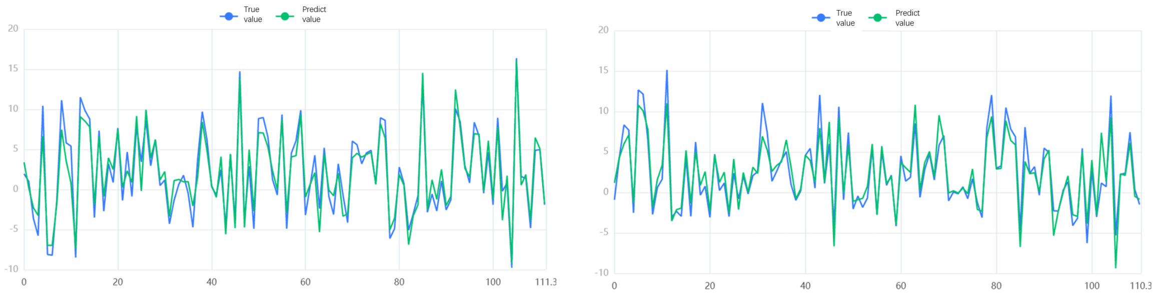


(a) Predicted Values and the True Values of μ (b) Predicted Values and the True Values of σ

Figure 14: The Training Result of Linear Regression ModelI

4.2.3 Linear Regression Models II of Normal Distribution Error

For each number of tries, we trained a linear regression model respectively to predict the error between the true percentage of it and the percentage of the predicted normal distribution derived from linear Regression Model I. Also, we divide the past data into 70% training set and 30% test set. Following are two of the comparisons between the predicted error and true error of the test set.



(a) Predicted Error and the True Error of $3tries$ (b) Predicted Error and the True Error of $6tries$

Figure 15: The Training Result of Linear Regression Models II

4.3 Evaluation of the Modified Normal Distribution Model

We use R^2 and $R^2_{adjusted}$ to evaluate the goodness of fit of the model. Specifically, the R^2 and $R^2_{adjusted}$ of linear regression model II take the average of the model predicting μ and the model predicting σ, and the R^2 and $R^2_{adjusted}$ of linear regression model II take the average of all 7 linear regression models of different tries.II.

Table 4: Model Evaluation

Model	R^2	$R^2_{adjusted}$
Linear Regression I	0.494	0.485
Linear Regression II	0.643	0.632
Modified Normal Distribution Model	0.774	0.762

From the table above, we can see that the final combined Modified Normal Distribution Model improved indeed compared to the initial linear regression models. We are 70% sure of the accuracy of the results.

4.4 Prediction of the Results Distribution of $EERIE$

First of all, we calculate the attributes of $eerie$. In particular, we get the value of N and H using the model in task 1.

Table 5: Features of EERIE

EERIE	AI Judge	Duplicate letters	Letter frequency	Zipf frequency	H/N	date	vowel number	similarity
score	4.250	3	1.965	3.333	0.097	1.167	3	0

After bringing the desired variables into linear regression model I, we can get that $\mu = 4.662$, $\sigma = 1.135949$. Then we bring the variables into linear regression model II, we can get the predicted error of each tries' percentage. Finally, we add up the percentage of the predicted normal distribution and its error to get the distribution of the predicted results, which is shown below.

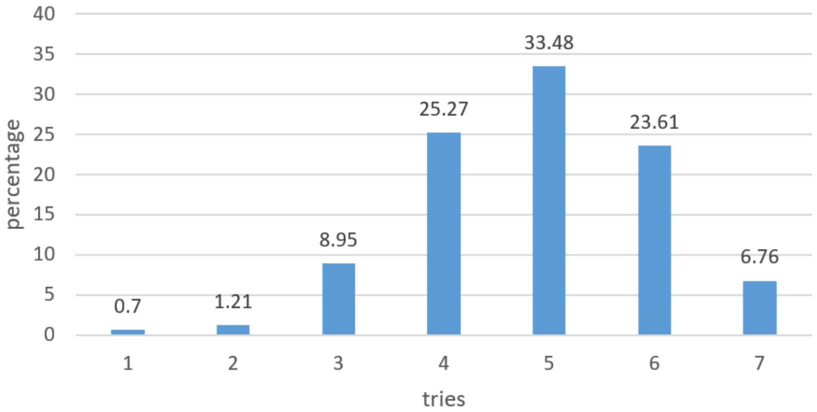


Figure 16: Predicted Results Distribution on March 1, 2023

5 Task 3: Develop a model to classify wordles by difficulty

5.1 Factors Related to Difficulty Level

To further evaluate the difficulty level of wordle, we should select some relevant indicators. In task 1, several word attributes are proposed. Likewise, we assume that these attributes are related to the difficulty level of wordle. We apply AT as the indicator of the difficulty level. Moreover, in task 3, we add a new factor,**AI judging**, to better determine the difficulty. Detailed mechanism of AI judging is discussed in the next section.

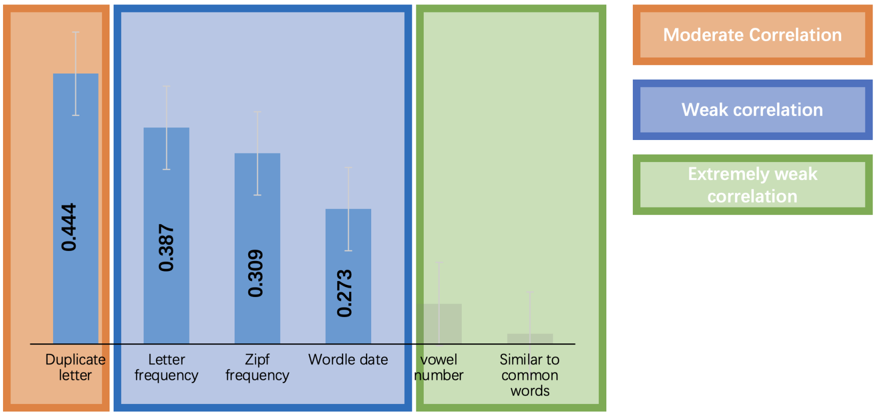


Figure 17: Correlation Coefficient Chart for AT

As shown in figure 17, difficulty level AT is weakly related to vowel number and its similarity to other common words. In that way, the ultimate goal is to classify the difficulty level with the following factors: **AI judging**, duplicate letters, letter frequency, and Zipf frequency.

5.2 AI Difficulty Judging

Table 6: Notations

Symbol	Definition
$Freq_{letter}$	the frequency of letter used
$Freq_{word}$	the frequency of word used
En_{add}	new entropy to add

I.HumanoidBot

1 Inspiration

From the study we do in Task 1, we can conclude that the higher the daily use rate of letters in a word, the easier it will be for people. This is because humans always tend to remember and try popular words. According to human beings' memory and selection characteristics, we designed **HumanoidBot** based on **letter frequency** to imitate the process of human players playing wordle game.

2 Pseudocode

Algorithm 1 HumanoidBot

```
word-scope ← 2500 popular word
wordle-feedback ← green/yellow/grey feedback
$Freq_{letter}$← number of the letter in thesaurus
$Freq_{word}$← sum of $Freq_{letter}$
while input word is not right do
    if Get green block then
        word scope ← words with the same letter in the green position
    end if
    if Get yellow block then
        word scope words ← words with the same letter in the word but not at the yellow position
    end if
    if Get yellow block then
        word scope ← word scope − words with the same letter in the word
    end if
    transverse word scope and get the next input word
    if current $Freq_{word}$ > max $Freq_{word}$ then
        max $Freq_{word}$ ← current $Freq_{word}$
    end if
    next input word ← word with highest $Freq_{word}$
end while
```

3 Running Mechanism

step1: calculate $Freq_{word}$

We traverse all the words in the word scope and record the frequency of each letter. Then we caculate $Freq_{word}$ using the formula below

$$Freq_{word} = \sum Freq_{letter} \tag{6}$$

step2: update word scope depending on wordle feedback

when the block is yellow, we update the word scope by filtering out the word where the letter is in the word but not at the response yellow position; when the block is green, we update the word scope by filtering out the word with the same letters in the response green position; when the block is grey we update the word scope by dropping out words with the same letters in the word.

step3: find next input word

We then transverse all the words in the word scope and find the word with maximum word frequency and let it be the next input word.

4 Accuracy on History Set

We take the word given in "ProblemCDataWordle.xlsx" and get the following result:

Table 7: HumanoidBot Accuracy

Model	AT	success rate
HumanoidBot	4.58	93.72%

II. EntropyBot

1 Inspiration

HumanoidBot is designed with the essential idea of always choosing the words the with highest word frequency. However, in reality, We can figure out which words can help us better narrow the scope of words. The idea has a scientific terminology – **Information entropy**.

definition of information

$$I = \log_2(\frac{1}{p}) = -\log_2(p) \tag{7}$$

the information turns the probability of an occurrence into several bits.

definition of expected information

$$E[I] = H(p) = -\sum_{x} p(x)\log_2 p(x) \tag{8}$$

E[I] tells us the amount of information on average and is known as Shannon entropy. The main idea of Entropybot is to choose a word with the maximum entropy, because it can minimize the probability space[5].

2 Pseudocode

Algorithm 2 EntropyBot

```
word-scope ← 2500 popular word
wordle-feedback ← green/yellow/grey feedback
calculate $Freq_{letter}$ and $Freq_{word}$ in the same way as HumanoidBot.
while word is not right do
    for every word in word scope do
        for every letter in word do
            Fill different colors
            $En_{add}$ ← entropy when in particular color
            entropy ← entropy + $En_{add}$
        end for
    end for
    get the next input word
    if current entropy > max entropy then
        max entropy ← current entropy
    end if
    next input word ← word with highest entropy
end while
```

3 Running Mechanism

step1: calculate $Freq_{word}$

This step is similar to the first step of HumanoidBot, so we do not repeat it here.

step2: calculate entropy

We first transverse all the words in the word scope. Let's take "SHORT" in Fig 18 as an example.

Figure 18: Short

a

Then we draw different colors on the block in Fig19 and calculate its entropy.

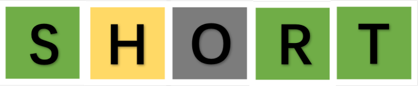


Figure 19: Short with Color

We can figure out the words meeting the above conditions, like a skirt. if n words fit this condition, then the added entropy in these conditions is

$$En_{add1} = -\frac{1}{n}\log_2\frac{1}{n} = \frac{1}{n}\log_2 n \tag{9}$$

Then we change the color of the word in Fig20.

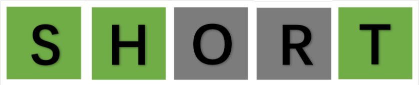


Figure 20: Short with Changed Color

We can also figure out the words meeting the above conditions, like sight. if m words fit this condition, then the add entropy in these condition is

$$En_{add2} = -\frac{1}{m}\log_2\frac{1}{m} = \frac{1}{m}\log_2 m \tag{10}$$

if we have k different color method, then the whole entropy for these word is

$$Entropy = En_{add1} + En_{add2} + ... + En_{addk} \tag{11}$$

step4: find word with maximum entropy

In this step, we transverse the word scope and find word with maximum entropy.

step5: input the word and update word scope

This step is similar to the second step of HumanoidBot, and so we do not repeat it here.

4 Accuracy on History Set

Table 8: EntroyBot Accuracy

Model	AT	success rate
EntroyBot	4.01	100%

5.3 Difficulty Level Classification Model

1 Normality Check

I Kolmogorov–Smirnov Test

We calculate that the p value is 0.310, which is greater than 0.05, indicating that AT meets the normal distribution.

II Normality Test Histogram

We obtain the following normality test histogram Fig21

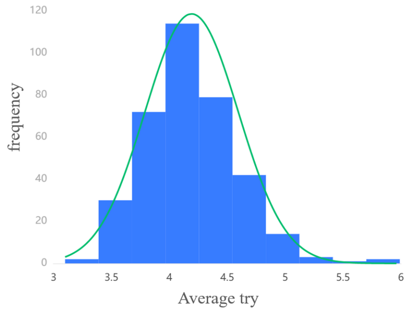
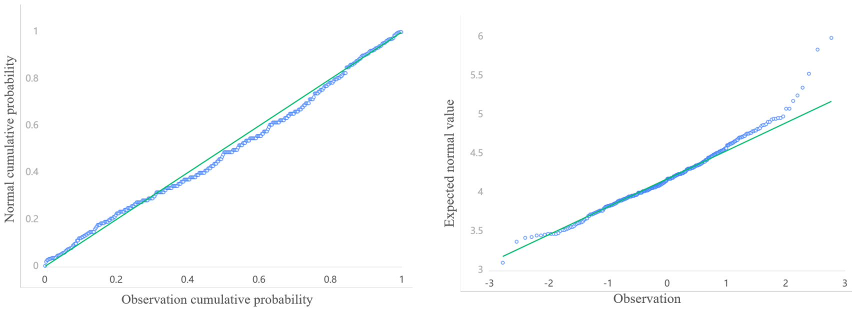


Figure 21: Normality Test Histogram

III P-P and Q-Q Diagram of Normality Test

We draw the P-P and Q-Q diagram as follows.



(a) P-P Diagram of Normality Test (b) Q-Q Diagram of Normality Test

Figure 22: P-P and Q-Q Diagrams

We can see from the graph that the scattered points coincide well with the straight line, which indicates that AT obeys normal distribution.

2 Set Difficulty Levels

According to the characteristics of normal distribution [6], we know that

$$Pr(\mu - \sigma \leq X \leq \mu + \sigma) \approx 0.6827 \tag{12}$$

Table 9: Difficulty Distribution

difficulty level	easy	medium	hard
ratio	0.16	0.68	0.16

We calculate the correlation coefficient of normal distribution as follows

$$\begin{aligned} \mu &= 4.192 \\ \sigma &= 0.404 \end{aligned} \tag{13}$$

Then we can get the dividing line below

$$
\begin{aligned}
line_1 &= \mu - \sigma = 3.788 \\
line_2 &= \mu + \sigma = 4.596
\end{aligned}
\tag{14}
$$

5.4 Use Logistic Regression to Classify

We use a logistic regression algorithm to classify the difficulty level of data according to [AI Judging, Duplicate letters, Letter frequency, Zipf frequency], and Figure 23 is our result on test set.

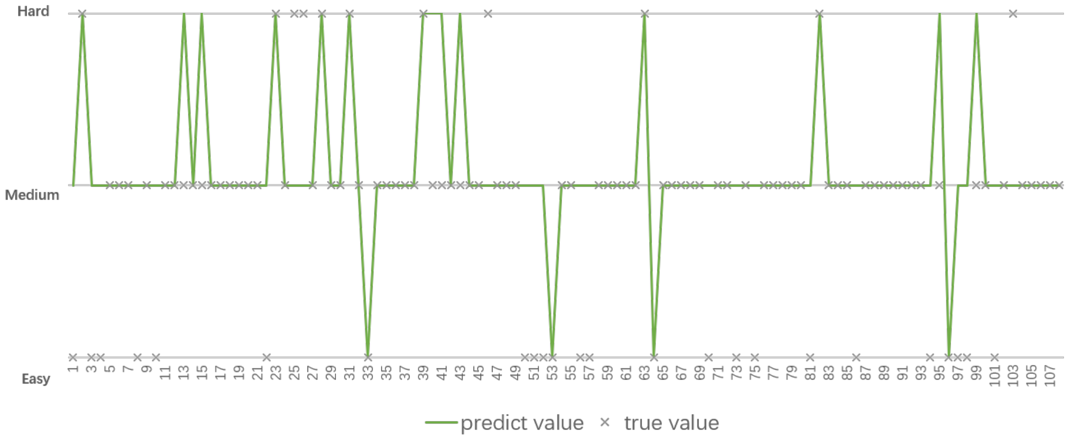


Figure 23: Classified Result on Test Set

The relevant characteristics of our training model are as follows.

Table 10: Feature of Model

features	Accuracy	Recall	Precision	F-score
Train set	0.793	0.793	0.781	0.769
Test set	0.759	0.759	0.804	0.725

5.5 Classification Result of EERIE

We calculate the characteristic values of EERIE as follows:

Table 11: Features of EERIE

EERIE	AI Judge	Duplicate letters	Letter frequency	Zipf frequency
score	4.25	3	1.965	3.33

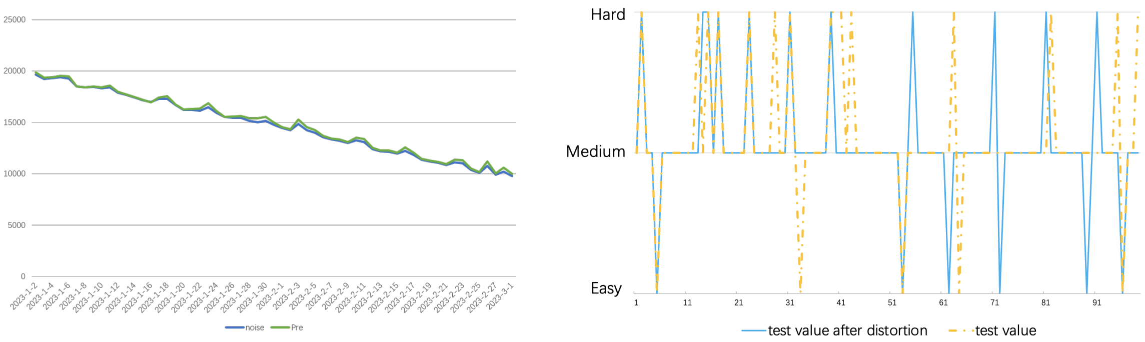
The classification results of EERIE are as follows

Table 12: Classification of EERIE

Predict result	Easy probability	Medium probability	Hard probability
Hard	0.132%	25.3%	74.6%

6 Analysis on Model

6.1 Robustness Analysis



(a) Robustness of ARIMA-GRU

(b) Robustness of Logistic Regression Model

Figure 24: Robustness Analysis

For task 3 and task 1,it is significant to consider whether the ARIMA-GRU model and Logistic Regression(LR) model can be stable under different initial models. Therefore, we add a noise with a normal distribution with $\mu = 0$ and $\sigma = 0.05\mu_{original}$ to the input data of both models.

As for ARIMA-GRU model,our prediction results and original results are shown in Figure 24(a). The difference between the prediction results of the original data and the data with noise is small, so our model has strong robustness. Our test data of LR model before and after distortion is similar and the ultimate classification result of EERIE is the same, i.e.Hard.

6.2 Strengths and Weaknesses

I Strengths

- We combine ARIMA model and GRU model to predict the linear part and nonlinear part of the number of reported results. The advantages of ARIMA in predicting stationary series are fully utilized, and the influence of GRU over-fitting is effectively reduced. Through practical results, our **ARIMA-GRU** model has achieved good prediction results.
- We propose seven hypothetical factors that may affect the difficulty and analyze their correlation. Our selection of influencing factors is comprehensive and detailed.
- We innovatively divide the distribution of the results into a normal distribution part and a non-normal part, which provides a new solution to the distribution of a small range of discrete data.
- We develop two **AI robot**–HumanoidBot and EntropyBot. **HumanoidBot** is based on word frequency, whose **mode of thinking** is similar to that of human beings. **EntropyBot** is based on information theory, which can present word difficulty from a scientific perspective.

II Weaknesses

- The accuracy of classifying difficulty is 75.9%, which is not high enough. This is due to some subjective factors when determining the difficulty of the problem, and at the same time, we have unavoidable errors in the quantitative indicators of the characteristics of words.
- Generally, players with higher scores in wordle will tweet, so there is a certain deviation between the tweet data and the actual data.

7 Other Interesting Features of the Data

- We find that words with **higher letter frequency** are easier than words with **higher word frequency**. This situation indicates that compared with choosing familiar words, people tend to **search by letter** when thinking about wordle answers.
- Duplicate letters in word made the word more difficult to guess, which may result from the position uncertainty of wordle's yellow hint in that it may confuse the player.
- Number of reported results on Twitter has shown a significant decrease in recent months, however, hard mode ratio gradually increases.This shows that with the decline of the popularity of wordle, a large number of people no longer play wordle, but loyal fans and wordle masters tend to remain.
- On average, 95% of wordle puzzles can be solved within 6 guesses, and the game needs approximately 4 tries.
- Overall, people are becoming better at solving wordle. During the first 60 days, people's average score is 4.17 tries, but during the last 60 days, people's average score decreased to 4.11, which means people's overall ability of solving wordle increased by about 3.22%.
- we can see that the number of reported results has a **local peak** in a week. We think this is caused by two reasons:
  1. Every week, there will be a relatively simple topic that people try less, and people will share it through Twitter after getting better scores.
  2. On weekends, people will have more time to play wordle games and share on Twitter.

8 Our Letter

From: Team #2304217
To: the Puzzle Editor of the New York Times
Date: February 19, 2023,
Subject: A Mathematical Analysis of Wordle Based on Reported Results

Dear Sir or Madam:

Congratulations on the success of Worlde! Wordle has reached an unprecedented level of popularity and activeness and gained millions of daily players. As big fans of wordle, we did a mathematical analysis of the reported Wordle game results from 2022-1-7 to 2022-12-31 collected from Twitter, hoping to be of some help to you.

First, we observed the number of people posting wordle scores over time. Also, we predicted the variation in the future. Secoother new modes some attributes of the words themselves, we screened out the attributes that have an impact on the difficulty of the wordle puzzles and tried to find out whether there exists a correlation between these word attributes and the proportion of people playing the hard mode. Thirdly, we thought of an innovative and mathematical, way to describe the results distribution of a certain Wordle puzzle, which could also be used to predict the results distribution of a certain word puzzle in the future date. In addition, we developed a scientific classification system to evaluate the difficulty of a word as a wordle puzzle, which may be a reference for you to determine the answer word of the day.

Overall, we have the following conclusions and suggestions for you.

- The number of daily reports soars in the beginning and reaches its maximum on February 22, 2022, which is 306356, wordle has really caught on and made a big hit. However, when people's temporary enthusiasm faded away, the number of reported results decreased slowly over time. However, after the proportion of people playing hard mode increased at first, it remained rather stable, which indicates that the players like this mode. It's maybe a good idea for you to develop other new modes of wordle.
- The word's difficulty is most affected by the number of duplicate letters, letter frequency, and word frequency. If you want to set an easy puzzle, you can go with words like $arose$, $alien$, and $raise$. If you want to give players a big challenge, it's a nice try to go with words like $hitch$, $gawky$ and $parer$. In addition, the proportion of people playing hard mode has nothing to do with the word's attributes.
- We use our setup model to predict the future results quantity and distribution of the word $EERIE$ on March 1, 2023. It turns out that there will be 9993 people posting their wordle results on twitter, and the percentage of 1 try, 2 tries, 3 tries, 4 tries, 5 tries, 6 tries and 7 or more tries will respectively be 0.70%, 1.21%, 8.95%, 25.27%, 33.48%, 23.62% and 6.76%. It shows that $eerie$ is a very difficult word. Our classify system also got the same result.

If you are interested in our specific mathematical model and algorithms, please feel free to contact us. We are glad to present more details of our thoughts.

Best regards

Team #2304217 of 2022 MCM

References

[1] Wikipedia contributors. *Autoregressive integrated moving average. Wikipedia.* . (2023, February 15). `https://en.wikipedia.org/wiki/Autoregressive_integrated_moving_average`

[2] Simeon Kostadinov. "Understanding GRU Networks" *Towards Data Science* (2017, December 16). `https://towardsdatascience.com/understanding-gru-networks-2ef37df6c9be`

[3] Barry Smyth. "Peak Wordle Word Difficulty." *Towards Data Science* (2022, February 22). `https://towardsdatascience.com/peak-wordle-word-difficulty-64907be4c177`

[4] Wikipedia contributors. *Letter frequency. Wikipedia.* . (2023, February 15). `https://en.wikipedia.org/wiki/Letter_frequency#:~:text=Letter%20frequency%20is%20the%20number%20of%20times%20letters,who%20formally%20developed%20the%20method%20to%20break%20ciphers.`

[5] 3Blue1Brown. "Solving Wordle using information theory" *Youtube* (2022, February 6). `https://www.youtube.com/watch?v=v68zYyaEmEA`

[6] Jay L.Devore, probability and statistics for engineering and the sciences, *Cengage Learning*, Boston, 2012.

Appendices

TASK 2

The linear regression

μ = 3.179 + 8.033*r + 0.189*ai - 0.081*word freq - 0.012*letter freq + 0.259*duplicate - 0.711*date

σ = 0.947 + 0.056*ai - 1.128*r - 0.051*duplicate - 0.029*date + 0.021*similar number + 0.002*letter frequency + 0.033*vowel number

$error_{1try}$ = 0.222 + 0.705*duplicate - 0.069*vowels - 0.09*similar number - 0.056*letter frequency + 11.036*r - 0.03*ai - 1.132*u + 3.808*delta - 0.17*word freq - 0.904*date

$error_{2tries}$ = 14.635 + 2.806*duplicate - 0.273*vowels - 0.147*similar - 0.115*letter freq + 123.94*r + 0.97*ai - 9.374*u + 14.722*sig - 0.87*word freq - 8.352*date

$error_{3tries}$ = 60.353 + 4.17*duplicate - 0.719*vowels - 0.171*similar - 0.154*letter freq + 127.966*r + 3.654*ai - 18.772*u + 3.402*sig - 1.452*word freq - 11.374*date

$error_{4tries}$ = 35.434 + 0.644*duplicate + 0.593*vowels + 0.447*similar - 0.009*letter freq + 13.105*r + 2.452*ai - 3.756*u - 28.124*sig - 0.324*word freq - 4.053*date

$error_{5tries}$ = -27.548 - 3.996*duplicate + 1.618*vowels + 0.413*similar + 0.153*letter freq - 161.554*r - 2.583*ai + 13.038*u - 15.869*sig + 1.486*word freq + 11.571*date

$error_{6tries}$ = -49.818 - 3.19*duplicate - 0.281*vowels - 0.178*similar + 0.124*letter freq - 125.613*r - 3.258*ai + 13.946*u + 9.472*sig + 1.115*word freq + 10.95*date

$error_{7tries}$ = -35.325 - 0.988*duplicate - 0.806*vowels - 0.223*similar + 0.068*letter freq + 10.985*r - 1.011*ai + 5.93*u + 14.201*sig + 0.163*word freq + 1.845*date