The Research on Word Game based on SIRS-ARIMA Model and Machine Learning Algorithm

Junjun Hu
College of Computer, Huainan Normal University, Huainan, 232000, China

Xiaoyan Li
College of Computer, Huainan Normal University, Huainan, 232000, China

Yongkuo Zhang
Jilin University of Finance and Economics; Changchun 130000, Jilin, China

Xiajie Ai
College of Computer, Huainan Normal University, Huainan, 232000, China

Lei chen
College of Computer, Huainan Normal University, Huainan, 232000, China

Correspondence author: Xiaoyan Li

ABSTRACT
Various word game software is becoming more and more popular, such as the recently popular "Wordle" crossword game, which can entertain, develop intelligence, and improve word learning ability. However, there are little research on how to improve the challenge and innovation of word games. For this challenge, this paper focuses on the research of word games based on SIRS-ARIMA model and machine learning algorithm. The SIRS-ARIMA model is an innovative approach that combines the SIRS propagation model and the autoregressive integrated moving average model (ARIMA) to analyze and predict dynamic changes in the word game by taking into account factors such as social media propagation. This paper also uses the entropy method of machine learning algorithm and SVC model to classify the difficulty of words, so as to optimize the design and play of word games. By analyzing player behavior and word attributes, it can personalize the game experience and provide players with precise feedback mechanisms. This research provides new theories and methods for the development of word games and provides strong support for the design of more engaging and innovative games.

Keywords
SIRS Model, ARIMA, machine learning, Wordle Game, SVC

INTRODUCTION
Word game software has become increasingly popular today, and many parents use word game software to enhance their children's interest in language learning during the language learning phase, thereby improving their children's learning skills. Teachers use crossword puzzles as a means of digital instruction, and students find it more helpful and fun to work collaboratively on crossword puzzles in the classroom to learn concepts and prepare for tests than to do them individually outside of class [1-2].Word game software has also become a means of raising awareness and memory skills in older adults, and studies have shown a direct correlation between the frequency of word game use and cognitive function in adults aged 50 and older. Word games can stimulate brain activity and contribute to the prevention of diseases such as dementia [3]. Given the growing popularity of word game software among various populations, this paper focuses on the well-known online word game Wordle which was developed by Josh Wardle, which was released online in January 2022, it has gained a lot of popularity [4], millions of people play this game every day. In "wordle", players must guess a five-letter word in six attempts. This hidden word, also known as a "mystery word", is selected daily from a list of 2,315 words based on a certain distribution and is posted on the New York Times website. Among the published studies are reports of optimal and suboptimal calculations. Some scholars have used Wordle to analyze the word cloud of verbal and written responses of informants, it is a special kind of textual visualization in which words that are used more frequently are effectively highlighted by occupying more prominent positions in the expression [5] and [6].In this paper, SIRS model was used to predict and analyze the results of wordle word game player calculations and machine learning algorithms were used to classify wordle word difficulty. The contributions of the work in this paper include the following sections:

(1) The prediction and validation on the number of reported results for the wordle word game in this paper is beneficial to wordle developers in optimizing word difficulty in a timely manner, thus improving player's entertainment and learning experience.

(2) This paper uses SVC to classify word difficulty, which is conducive to developers to adjust the word bank and design to help players improve their vocabulary and language ability.

(3) The exploration of player's learning behavior in this paper can understand players' habits of using words so as to design and provide a more personalized experience.

2. RELATED WORKS
Wordle word game software has attracted the attention of many scientists around the world since its first online release in 2022 [4]. With the popularity of machine learning algorithms, many scientists have been using machine learning algorithms to study the wordle word game software more and more. Bhambri et al., used a reinforcement learning approach to train an intelligence to select the best strategy for word guessing in a Wordle game, where the state of the intelligence was represented as a tuple (containing the number of green and yellow letters) and a sequence of word guesses. The experimental results show that the reinforcement learning algorithm significantly improves the performance of the Wordle game [7]. Benítezet et al., used a Monte Carlo tree search algorithm to solve the Wordle game. The authors modeled the Wordle game as a partially observable Markovian decision process and used sequences of guessed words as actions of the intelligences. Experimental results show that excellent performance can be obtained using the Monte Carlo tree search algorithm in Wordle games [8].

Anderson et al., provided some heuristics, such as choosing the next word guess based on letter probability and the principle of minimum entropy. The authors studied a variety of word guessing strategies and compared their performance in the Wordle game [9]. De Silva applied statistical analysis to assess the effectiveness of the strategy and found that using letter probabilities and picking words containing known letters significantly improved the success rate of the Wordle game [10]. Zeng designed four strategies that both computers and human players can learn and trained a Q-learning-based AI to automatically solve this game, which can give suggestions to players and provide useful combinations in various situations. However, it still cannot effectively solve some rare situations and may need more training or new ways to set the game state [11]. Short, M.B. proposed two heuristics ("Fastest Descent" and "Maximum Expected Probability") for computational experiments, and they introduced a solution for Wordle games that can significantly improve their performance [12]. Selby [13] For the first time, the optimal strategy for playing Wordle is implemented and the corresponding optimal scores are given. The average of these scores corresponds to a randomly selected mystery word from a mystery list according to a uniform probability distribution.Bertsimas and Paskov [14] verified the optimal scores of [13] using a dp-based solution method, and the authors noted that with respect to wordle puzzles, DP computation is very fast for online optimal solutions of individual puzzle words. Ho A proposed RL solution methods such as Deep-Q Learning and Advantage Actor Critic, with considerably worse results to the maximum information gain heuristic [15]. [16] and [17] give suboptimal strategies for solving puzzles online and give research work to obtain theoretical complexity guarantees.

3. METHODOLOGY

In this paper, SIRS model and ARIMA model are used to simulate and predict number of reported wordle results and explore the influencing factors affecting the number of difficult mode passers, and finally classify the word difficulty based on the entropy value method as well as SVC.

3.1. SIRS model and ARIMA model assumptions.

For the SIRS model and ARIMA model, there has the following model assumptions:

(1) The total number of Twitter users N remains unchanged.

(2) The population is divided into three categories: general users, game players and players who have abandoned the game. The number of people in the three categories at time t is denoted as s(t), i(t) and r(t) respectively.

(3) The probability of an ordinary user playing the game is λ, and the probability of a gamer giving up the game is μ.

(4) Game players may still play the game again after giving up the game, and there will be a probability Y of game players will play the game again after giving up the game every day.

(5) The total number of players in a period of time is relatively stable, not considering the large number of game players joining only the number of set.

(6) The player n(t) who is playing the game at moment t is a continuously varying, differentiable function with time.

(7) The number of effective exposures (sufficient to cause disease) or new games per player per unit of time is λ ($\lambda \geq 0$ is a constant).

(8) The number of players as a percentage of the total number of patients is μ.

(9) Players can still be converted into regular users after giving up the game.

(10) It is assumed that the encoding of words represents the variation in structure between words.

(11) It is assumed that characteristics such as frequency of use represent how often people use them in their lives.

The model symbol description can be shown in Table 1.

Table 1. Model symbol description

Symbols	Description
N	Total number of Twitter users
t	t moment
s(t)	Number of Twitter users who did not play games on day t
i(t)	Players who played the game among Twitter users on day t
r(t)	Players who abandoned the game among the users of Twitter on day t
λ	Probability of the average user playing the game
μ	Probability of gamers abandoning the game
T	Probability of players posting daily reports on Twitter
m	The percentage of people who can be influenced to play the game by an average article
X_i	Represents the percentage of the ith passing case
$\in_i$	Mean white noise sequence
B	lag operator
$(b_{ij})_{m\times n}$	A matrix of i samples and j indices

V_{ij}	indicator
e_i	entropy
W_i	Weighting
σ	The probability of spreading the game

3.2. SIRS model.

The infectious disease model was first proposed in 1927 by Kermack and McKendrick for their study of the scale of the Black Death epidemic in London. The SIRS model of infectious disease divides the population into three categories: Class S. Susceptible, refers to people who do not have the disease but lack immunity and are susceptible to infection after contact with an infected person. Class I. Infectious, refers to people who have contracted an infectious disease that can be transmitted to members of Class S. Class R. Recovered, refers to a person who has been isolated or has become immune as a result of recovery from illness. If immunity is limited, a member of category R can become a member of category S again.In this paper, ordinary users are denoted as s(t), game players are denoted as i(t), players who give up the game are denoted as r(t), N is the total number of people, and the relationship between these three categories and the total number of people N is N=s(t)+i(t)+r(t), where t represents the moment t.

Considering the reality, this paper argues that Twitter posts affect the motivation of existing players to play the game and the possibility of ordinary users to become players, and that some players who have played the game may still play the game again after giving up on it, the process figure is shown by Figure 1.

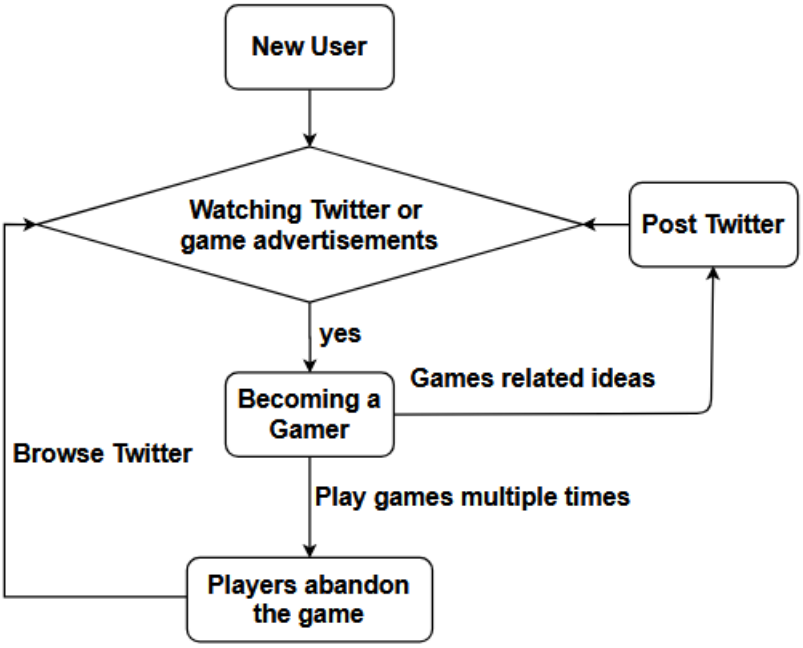

Fig 1: Wordle word game flow

Based on the Figure 1, the SIRS model is considered more suitable for simulating in this paper.SIRS model building and solving: The analytic equation of the SIRS model as can be expressed as

$$\begin{cases} \frac{di(t)}{dt} = \lambda i(t)(1 - i(t)) - \mu i(t), t \geq 0, \\ i(0) = i_0. \end{cases} \quad (1)$$

$$i(t) = \begin{cases} \left[\frac{\lambda}{\lambda - \mu} + \left(\frac{1}{i_0} - \frac{\lambda}{\lambda - \mu}\right) e^{-(\lambda - \mu)t}\right]^{-1}, & \lambda \neq \mu \\ \left(\lambda t + \frac{1}{i_0}\right)^{-1}, & \lambda = \mu \end{cases} \quad (2)$$

Order:

$$\sigma = \lambda / \mu \quad (3)$$

Where: σ is the intensity of infection using the definition of σ, equation (1) can be rewritten as

$$\begin{cases} \frac{d_i}{d_t} = -\lambda_i \left[i - \left(1 - \frac{1}{\sigma}\right)\right], t > 0, \\ i(0) = i_0 \end{cases} \quad (4)$$

Correspondingly, the model resolution can be expressed as

$$i(t) = \begin{cases} \left[\frac{1}{1 - \frac{1}{\sigma}} + \left(\frac{1}{i_0} - \frac{1}{1 - \frac{1}{\sigma}}\right) e^{-\lambda\left(1 - \frac{1}{\sigma}\right)t}\right]^{-1}, & \sigma \neq 1, \\ \left(\lambda t + \frac{1}{i_0}\right)^{-1}, & \sigma = 1. \end{cases} \quad (5)$$

Results and analysis: From equation (4), when $t \to \infty$ w, there has

$$i(\infty) = \begin{cases} 1 - \frac{1}{\sigma}, & \sigma > 1, \\ 0, & \sigma \leq 1. \end{cases} \quad (6)$$

From the above equation, σ = 1 is a threshold value. If σ ≤ 1, then i(t) becomes smaller over time when $t \to \infty$ tends to zero. This is due to the fact that the cure rate is greater than the effective addition rate and eventually all players are dropped from the game.σ ≥ 1,then i(t) tends to the limit when($1 - \frac{1}{\sigma}$) $t \to \infty$ which indicates that a certain percentage of the total population will always be infected and become sick when the probability of abandoning the game is less than the game addition rate. SIRS model is a more typical system dynamics model, and its salient feature is that the model is characterized by a system of ordinary differential equations in the form of a model about multiple interrelated system variables. Such problems are difficult to find analytical solutions to and can be solved numerically using python.

Since not every player will upload his report on Twitter after finishing the game, this paper sets Y to denote the probability that a game player posts a report on Twitter every day, and the number of players posting reports on Twitter every day is i(t) * Y

3.3. ARIMA model.

In this paper, the ARIMA model was used to predict the "number of reported wordle results" interval. The Auto-regressive Integer Moving Average (ARIMA) is a statistical analysis model that can predict future trends using time series data. The basic idea is to consider the data series as a random sequence formed over time, and after identifying this sequence, the future of it can be predicted. For example, this model can be used to predict the number of reports on the day of wordle. The ARIMA model consists of an autoregressive (auto-regression, AR) model and a moving average (moving average, MA) model, which predicts the later values by measuring the effect of the previous value and uses a linear combination of the general equation of the ARIMA model is as follows.

$$w_j = \frac{1-e_j}{\sum_{i=1}^{m}(1-e_j)}\left(1 - \sum_{j=1}^{p} n_i B^i\right)(1-B)^{dY_t} \quad (6)$$

Parameters: L is the lag operator, d is the integer difference order, c is the constant term, $\emptyset_i$ is the autoregressive coefficient, α_j is the moving average coefficient, $\in_t$ is white noise. ARIMA stands out by integrating Autoregressive (AR), Integrated (I), and Moving Average (MA) components, using lag operators and coefficients to accurately predict future data points based on past values and errors. Implementing ARIMA involves parameter estimation through methods like least squares or maximum likelihood estimation, and a crucial step of residual testing to ensure no information is overlooked and to prevent overfitting or underfitting. ARIMA is a versatile tool across fields such as economics, finance, and environmental science, aiding in precise estimations and strategy.

3.4. Entropy value method modeling.

The entropy method is the first method to calculate the corresponding information entropy based on the relative range of variation of each index, and then calculate the weight of each method, which is more objective and scientific than the analytic hierarchy process (AHP). This paper extracts the features of words, and finally gets the features such as the frequency of word usage, word coding, whether it is repeated, and the proportion of difficult patterns.

The combined scores of words are calculated and an index of the difficulty of word differentiation is derived using the combined score system of words. The data are pre-processed and divided into test and training sets. The entropy method is used to calculate the corresponding information entropy based on the relative change magnitude of each index value, and then calculate the weight of each index. The calculation steps are as follows:

(1) Build the initial matrix. The matrix consisting of i samples and j indicators can be expressed as

$$B = (b_{ij})_{m\times n} = \begin{pmatrix} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{m1} & \cdots & b_{mn} \end{pmatrix} \quad (7)$$

(2) Data standardization. Data standardization is necessary to process index values in a dimensionless way, since inconsistent orders of magnitude and dimensions of each index can affect calculation results. The formula for dimensionless processing of positive indices is as follows.

$$v_{ij} = \frac{max(b_{ij}) - b_{ij}}{max(b_{ij}) - min(b_{ij})} \begin{matrix} i = 1,2,\cdots,m \\ j = 1,2,\cdots,n \end{matrix} \quad (8)$$

(3) Calculate the entropy value of this index, the entropy value of j.

$$v_{ij} = \frac{max(b_{ij}) - b_{ij}}{max(b_{ij}) - min(b_{ij})} \begin{matrix} i = 1,2,\cdots,m \\ j = 1,2,\cdots,n \end{matrix} \quad (9)$$

(4) Calculate the entropy parameter of this index.

$$e_j = -\frac{1}{ln\, m}\sum_{i=1}^{m} p_{ij}\, ln\, p_{ij} \begin{matrix} i = 1,2,\cdots,m \\ j = 1,2,\cdots,n \end{matrix} \quad (10)$$

3.5. SVC model.

Support Vector Machine (SVM) is a set of machine learning algorithms categorized into SVC (classification) and SVR (regression). SVM serves for classification tasks while SVC handles regression tasks. Support vector classification (SVC) is a classification machine learning model found in the Scikit-learn library. It is used to fit a hyperplane or multiple hyperplanes in a high-dimensional space to separate and classify data points. To choose the optimal decision boundary, SVC selects the hyperplane with the largest boundary. For better classification accuracy, the model can use various types of kernels to transform the input data into a higher dimensional space. To create a SVC model, the SVC function of Sklearn can be used and the labels are predicted using the coefficients of the model (w) and the intercept (b) term obtained from the coef_ and intercept_ properties of the model. Since a single hyperplane can only classify two classes, the parameters of the model include the kernel type, the regularization parameter C, and the gamma of certain kernels. After training on the training data, the model can predict the class labels of new data using prediction methods, and the scoring method can evaluate the

accuracy of the model by returning the average accuracy of the test data and labels. Using the coef_ and intercept_ properties of the fitted model, the equations of the decision boundary can be obtained

4 ANALYSIS OF EXPERIMENTAL RESULTS

4.1. Datasets

The data in this paper are derived from data from 4800000 Twitter users published by the New York Times from January 17, 2022 to December 31, 2022. Including the number of reports posted daily by game users in Twitter, the word of the day, the number of reports of difficult mode, the number of daily reports of difficult mode, the number of times the daily game needs to be successful (x = 1 to 7) as a percentage of the number of people.

4.2. Data pre-processing

First, dataset should be processed by removing invalid and missing data, correcting the incorrect word data, and changing abnormal values. Resampling as well as smoothing of the data is performed. Since ARMA and ARIMA models require that the time series must meet the requirements of smoothness and non-white noise, the use of difference and smoothing (rolling average and rolling standard deviation) is required to achieve smoothness operation of the series. Usually, the first-order difference method is sufficient to achieve the smoothness of the series, but sometimes second-order differences as well as smoothness tests are required.

The word attribute values are obtained by the following methods:Coding of words: These words are coded according to their composition. In this article, the code for a is “01” and the code for b is “02”. By analogy, the codes of the 26 letters of the alphabet can be obtained. For example, the code of the word “eerie” is “0505180905”. Add “0” to the code and convert the string to a floating point number. This gives a decimal number that reflects the composition of the word.Word Sentiment Score: Sentiment score of words in the data based on the NRC Word-Sentiment Association Dictionary produced by Dr. Saif M. Mohammad and Dr. Peter Turney (Senior Research Scientist, National Research Council of Canada) to obtain the sentiment attributes of the words.Word composition case: determine if there are duplicate letters in the word. If there is, it is 1, if not, it is 0. Frequency of word usage: This paper searches the frequency of each word on Google Ngram Viewer for 2019.

4.3. Prediction and validation on the player’s reported results using SIRS model and ARIMA model

In this paper, we established SIRS model and ARIMA model to predict the total number of users posting game results on Twitter every day by iterating, and simulated and predicted using SIRS model by continuously adjusting parameters such as the probability of players posting game results on Twitter, the probability of players giving up playing games and the probability of new users starting to play games every day, and used ARIMA model was used to verify the prediction results of the SIRS model. The model parameters are set as shown in Table 2. The parameter y here is the probability that the old player will restart the game. The parameter P(t) here is the number of Twitter blog posts by players who played the game on day 0. The explanation of other parameters can be found in Table 2

Table 2. The model parameters setting

Parameters	Value
N	4800000
λ	0.2
μ	0.0662
ν	0.248
m	3.6
y	0.11
P(t)	203680

In this paper, we use R^2as the evaluation index of the model and R^2as the coefficient of determination, which takes a value between 0 and 1. The closer this value is to 1, the better the fit of the model is, and the formula for R^2 is shown below:

$$R2=\frac{SSR}{SST}=1-\frac{SSE}{SST} \quad (11)$$

The SIRS (Susceptible-Infected-Recovered-Susceptible) model simulates the user dynamics in a virtual social network playing the Wordle game. In this model, the players are vulnerable to the infection, they spread the virus easily. People who belong to an active category are communicators in this model (I - Infected). The passive players in this model are those who got recovered from game addiction (R - Recovered). The model simulates the transitions between these player states by defining a set of parameters and methods. The ‘__init__’method initializes fixed values such as the number of users, Twitter effect rate, and the rate of new players. It also defines internal state variables that should keep track of the users' counts and their current states. The ‘initialize’ function sets the number of players at the beginning and resets the game variables to the beginning. The ‘update_status’ method checks the current model state based on the given structure, separating the number of initial new players, returning ex-players, and the number of attrition participants. Then it updates the corresponding state variables. The ‘twitters’ method returns the current Twitter propagation count. In the main program, we have a ‘WordleDynamic’ instance copy. The model is being stocked up, and multiple state update iterations are performed, recording changes of Twitter propagation, which are visualized as a trend on the plot. This model simulates the dynamic changes in player states, similarly to the SIRS effect, to better understand and predict the user population evolution in the Wordle game. Game researchers adjusting the model in a way that simulates different user behaviors give the field of game analysis wide possibilities and several forecasts for game operators.The experimental results are shown in Figure 2 and Figure 3.

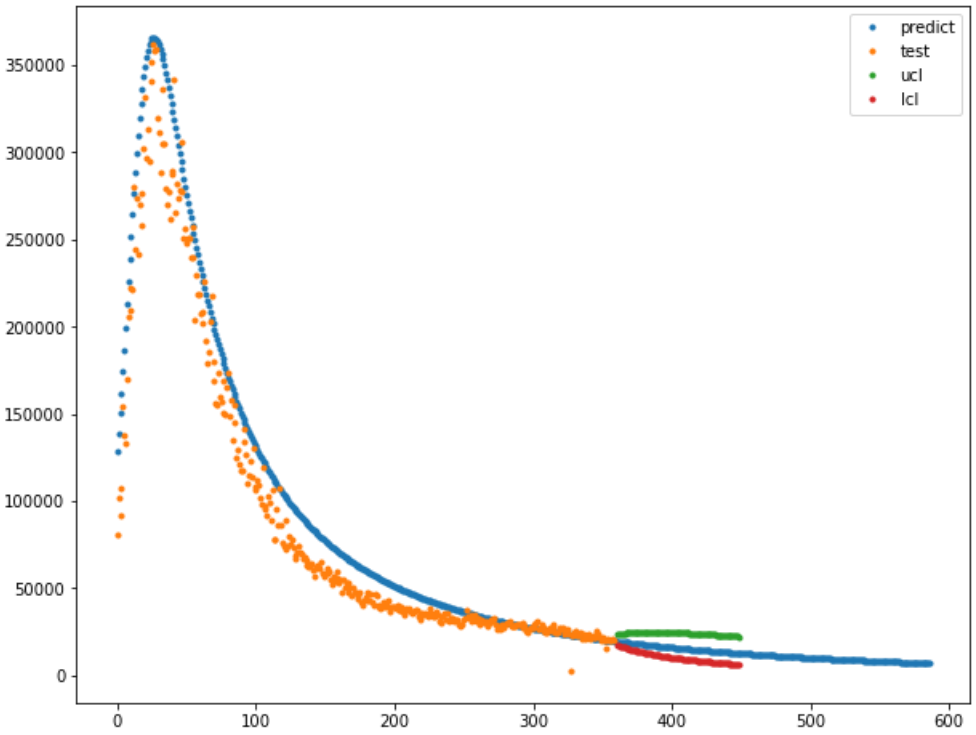

Fig 2: SIRS simulation prediction results

The write-up showcases a forecasting approach that combines ARIMA (Autoregressive Integrated Moving Average) and SIRS (Susceptible-Infected-Recovered-Susceptible) methods. It predicts the Twitter user count for Wordle game enthusiasts over the next two months. The trend line (predict) shows a downward trajectory, suggesting a decline in participant numbers. This could result from waning interest and player attrition as the game's novelty fades. The ARIMA model provides a confidence range, with upper (ucl) and lower (lcl) bounds, indicating forecast uncertainty.

However, the observed values (test) reveal substantial fluctuations, highlighting the volatility of the Wordle participant count. By integrating SIRS and ARIMA techniques, the model aims to capture the dynamic user behavior more e-ffectively.

To summarize, the forecast employs a hybrid approach, combining time-series analysis (ARIMA) and epidemiological modeling (SIRS), to project the future trajectory of Wordle's Twitter following. The declining trend line reflects anticipated interest waning, yet the observed data underscores the inherent unpredictability of such social phenomena.

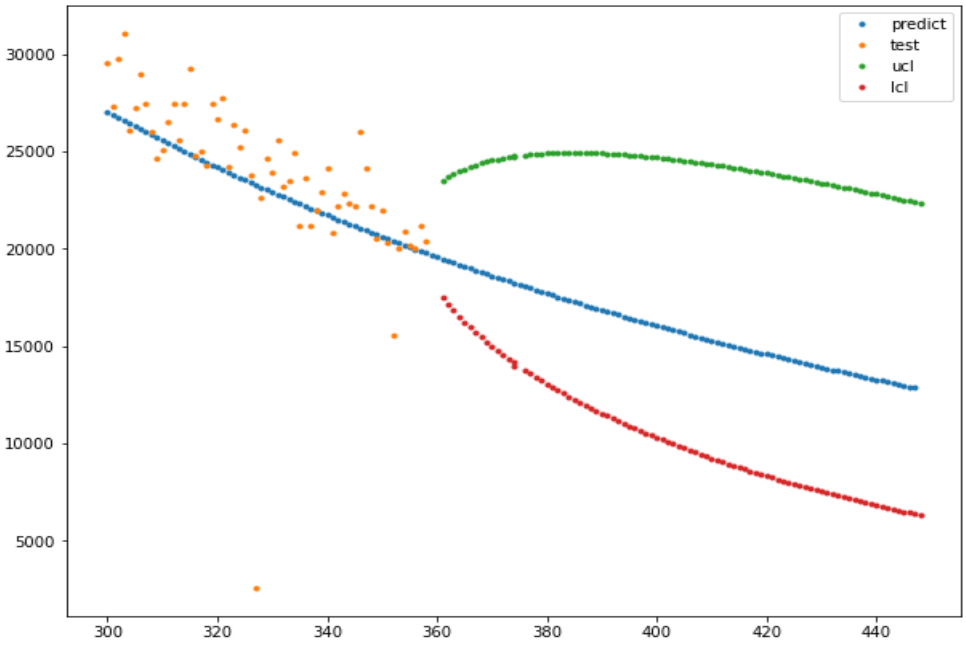

Fig 3: The validation about simulation predition

The horizontal coordinates in Figures 2 and 3 indicate the period from January 7, 2022 to the next 600 days, i.e., December 31, 2023, and the vertical coordinates indicate the number of published reports. The yellow line indicates the original values, blue indicates the simulated test results, and the red and green lines indicate the upper and lower limits of the predicted values, respectively. In this paper, the data of the number of published reports within 348 days from January 7, 2022 to December 31, 2022 are collected for the simulation test, and the data within 349 to 600 days are used for the forecast. By calculating the model evaluation index R^2, the value of R^2 here is 0.9415. The value of R^2 and Figure 2 show that the model simulation value is close to the original real value and has a good simulation effect.

This work used the ARIMA model to validate the prediction results of the SIRS model, and confidence interval need to be calculated first by the ARIMA model to get a better range of predicted values. The method for calculating the confidence interval was based on the standard error and the t-distribution. The standard error was a measure of error and uncertainty that represents the average error between the predicted and true values. t-distribution was a probability distribution that was used to calculate the confidence interval for a given confidence level. The upper and lower values of the confidence interval depended on the confidence level and the magnitude of the standard error. Here "ucl"denoted the upper limit of the confidence interval and "lcl" denoted the lower limit of the confidence interval. From Figure 2 and Figure 3, they showed that the future forecasts were within the confidence interval of ARIMA model, and the forecasts were very reliable after the robustness analysis. And we calculated the ARIMA model evaluation index R^2 and got the value of R^2 which was 0.985.

The values of the evaluation indicators of the models SIRS and ARIMA are shown in the following Table 3:

Table 3. The value of R^2

model	R^2
SIRS	0.985
ARIMA	0.942

The method in this paper combined machine learning and system dynamics approaches. The more accurate prediction of future results was very beneficial for predicting game trends and thus making timely adjustments. In the modeling process, it shows that the settings of the contagion model parameters had a large impact on the final prediction results, such as the daily probability of game players giving up the game, the daily probability of Twitter users playing the game, the total number of Twitter users, etc. The settings of these parameters affected the final prediction results of the contagion model.

4.4. Descriptive analysis of learning outcomes

In this paper, the frequency of word use, repetition or not, coding and sentiment scores were extracted and correlated with the proportion of reported difficulty patterns. By calculating their correlation coefficients, and heat maps were drawn. By observing the heat map which is shown in Figure 4, the frequency of word occurrence shows a positive correlation with the difficulty mode pass, indicating that the number of difficulty mode passers increases when the frequency of word occurrence is higher.

Fig 4: Correlation analysis

This paper has divided the roots that make up words, analyzed the compositional relationships between words by using the Apriori algorithm, and explored the effect of the relationships between roots on the pass rate of difficult patterns.The Table 4 demonstrates the impact of the root of some words: According to the results of association analysis, it was found that the association between words was low, and some results of root frequency statistics are shown in the Table 5.

Table 4. Roots of some word

slump	sl	um	mp	slu	lum	ump	slum	lump
crank	cr	an	nk	cra	ran	ank	cran	rank
gorge	go	rg	ge	gor	org	rge	gorg	orge
query	qu	er	ry	que	uer	ery	quer	uery
drink	dr	in	nk	dri	rin	ink	drin	rink
favor	fa	vo	or	fav	avo	vor	favo	avor
abbey	ab	be	ey	abb	bbe	bey	abbe	bbey
tangy	ta	ng	gy	tan	ang	ngy	tang	angy
panic	pa	ni	ic	pan	ani	nic	pani	anic
solar	so	la	ar	sol	ola	lar	sola	olar
shire	sh	ir	re	shi	hir	ire	shir	hire

According to the results of association analysis, it was found that the association between words was low, and some results of root frequency statistics are shown in the Table 5.

Table 5. The root and frequency

Root of word	Frequency of occurrence	Root of word	Frequency of occurrence	Root of word	Frequency of occurrence
er	28	ine	6	brin	2
in	27	sha	5	Show	2
st	22	tra	5	trai	2
lo	21	flo	5	floo	2
al	20	oke	5	prim	2
ar	19	dge	5	live	2

It was found through the research of this paper. The roots of a word are not related to each other. The roots of “er” and “in”, “st” and “lo” are all high-frequency off-roots, which indicates that several roots of words can be guessed in the game can guess several roots of words related to this.

4.5. Classification of word difficulty based on entropy value method and SVC

a. Select appropriate word attribute metrics, such as word repetition, word usage frequency and word encoding, for maximum and minimum processing of the data.

b. The entropy method was used to determine the weights of each index, and the weighted scores of each word were made according to the weights of each index.

c. Use whether the word is repeated, the frequency of using the word, and the word code as parameters.

d. The difficulty in classifying data based on the median of the composite score of a word is that words larger than the median of the composite score are more difficult, and vice versa.

e. The data is segmented using the model selection library and the training and test sets are segmented. The ratio between the training and test sets is 3:1.

4.6. Calculation results

The weight calculation results are shown in Table 6.

Table 6. The weight calculation results of Entropy method

Projects	Information entropy value	Information utility value	Weight (%)
frequency	0 665	0 335	55 886
coding	0.952	0.048	8.078
repeating	0.785	0.215	35.946
Porporting	0.999	0.001	0.088

The results of the entropy method of weight calculation are given in the Table 6 and analyzed to determine the weights of various indicators. Among the results of the calculation method based on the weights of entropy, the weight of frequency is 55.886%, the weight of coding is 8.079%, the weight of repeated sequences is 35.946%, and the weight of difficult mode ratio is 0.088%. Among these indices, frequency has the highest weight (55.886%), while difficult mode ratio has the most (0.088%). It can be seen in Figure 5.

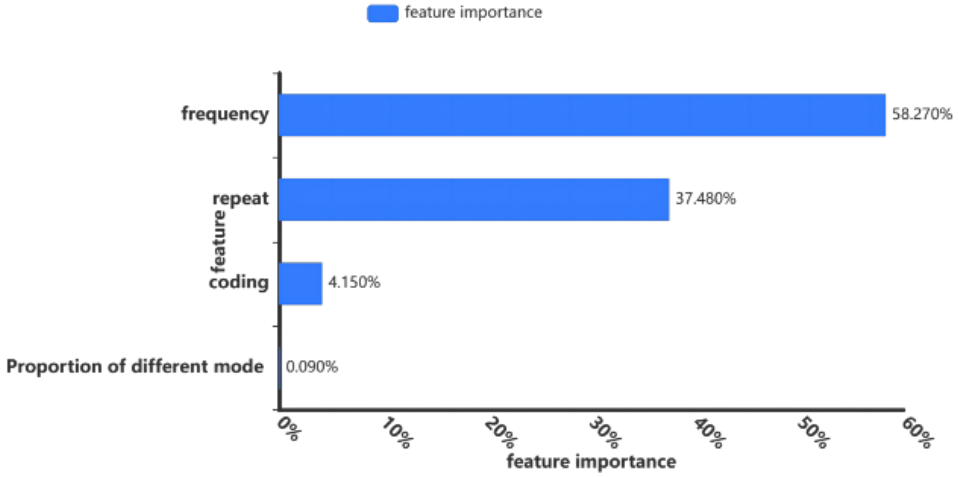

Fig 5: Feature importance

In this paper, each word is finally scored and the words are rated and ranked according to the scoring results.

4.7. Classification of scores using SVC

In this paper, the penalty coefficient parameters of the SVC kernel function, kernel function and objective function are adjusted by using the grid search method. The modeling process of SVC can be divided into the following steps:

(1) Prepare the data: load and pre-process the data. Split the data into training set and test set.

(2) Create an instance of the SVC model: Initialize the SVC model object with the required hyperparameters, such as the kernel type and regularization parameters.

(3) Fitting the model to the training data: Using the fitting method to train the model on the training data.

(4) Predicting the label of the test data: Using the prediction method to predict the label of the test data.

(5) Evaluating the model: The performance of the model is evaluated using metrics such as accuracy, precision, completeness and f1 score.

(6) Tuning hyperparameters: Try different hyperparameters to improve the performance of the model. 7. Repeat steps 3-6 until satisfactory performance is achieved.

(7) Use the training set to train the model.

(8) Test the model using the test set and calculate the accuracy of the model.

(9) The weights given by the entropy method were used to sum the scores based on the prediction results of the first problem. The difficulty of odd words was classified using SVC.In this paper, the best model parameters are determined by using grid search, and the model optimization steps are as follows:

In this paper, the best model parameters are determined by using grid search, and the model optimization steps are as follows:

(1) Compare the prediction performance of SVC models based on different kernel functions and select the optimal kernel function.

(2) Initialize the grid size, set the step size, and define the initial parameter values.

(3) Loop through each set of parameter combinations, combined with the five-fold cross-validation training SVC model, to get the value of the model evaluation index, the best performance of the evaluation index of the parameter combination value as the optimal.

(4) The optimal parameter combination is used as a parameter term to train the final model and obtain the SVC-based classification model.

The SVC-based classification model is obtained by the above training and optimization, and the median of the composite score is set as the threshold, and the data exceeding the threshold are designated as simple and those not exceeding the threshold as difficult.

In this paper, the accuracy rate is chosen as the evaluation index of the model, which will be obtained by dividing the number of correct samples by the number of all samples, since the binary classification model is built in this paper.

Using the trained SVC model to classify words, the accuracy rate is 94.44%, which indicates that the model can fit the original data better and predict more accurately.

The SVC model was developed to classify the difficulty of the "eerie" words as “difficult”, and the accuracy of the model classification was 94.44%. Some of the scoring table is shown in Table 7.

Table 7. The word’s Score

Number	Comprehensive evaluation	Word	Ranking
0	0.058199701	slump	184
1	0.008141981	crank	322
2	0.37345669	gorge	68
3	0.054169186	query	202
4	0.026658707	drink	265
5	0.024766508	favor	272
6	0.354135238	abbey	102
7	0.060773442	tangy	160
...	...	...	...
356	0.023522896	havoc	279
357	0.039844371	molar	238
358	0.039377053	manly	243

5. CONCLUSION

This paper uses the SIRS model to simulate and predict the learning reports of players in Wordle games. By using the SIRS model, it can simulate and predict players’ learning progress in the game, providing valuable insights into their performance and growth over time. The SIRS model can consider various factors that contribute to player learning, including the probability of players posting game reports and the influence of external factors. By incorporating these factors into the model, this paper can generate predictions that are closely related to actual player learning reports. To further verify the accuracy of the SIRS model predictions, an ARIMA model is used in this paper. This model is widely used for time series prediction and provides an additional validation layer for the simulation results of the SIRS model. By comparing the predicted values of the SIRS model with the actual data and applying the ARIMA model, we were able to assess the reliability and accuracy of our predictions in this paper. The results of this study highlight the potential of using the SIRS model to predict players' learning reports in Wordle games. In addition, this paper provides an attribute analysis of the words that appear in the game. Information entropy and SVC models were used to classify the difficulty of different words, and the difficulty of words that did not appear was evaluated. In conclusion, by providing insight into players' learning trajectories, the findings of this paper provide valuable insights for game developers, educators, and researchers, paving the way for enhanced game design, personalized learning experiences, and the advancement of learning analytics in the field of text-based games.

6. ACKNOWLEDGMENTS

Author Contributions: The authors of this research paper contribute as follows: Conceptualization, Xiajie Ai and Yongkuo Zhang.; methodology, Junjun Hu and Yongkuo Zhang; software, Junjun Hu and Yongkuo Zhang.; validation, Yongkuo Zhang and Junjun Hu.; formal analysis, Xiaoyan Li.; investigation, Junjun Hu.; resources, Junjun Hu.; data curation, Yongkuo Zhang.; writing—original draft preparation, Junjun Hu, Xiajie Ai, Xiaoyan Li.; writing—review and editing, Junjun Hu, Xiaoyan Li and Yongkuo Zhang.; visualization, Yongkuo Zhang.; supervision, Xiaoyan Li and Xiajie Ai.; project administration, Junjun Hu, Xiaoyan Li and Lei Chen.; funding acquisition, Xiaoyan Li. All authors have read and agreed to the published version of the manuscript.

Data Availability: The datasets used to support the results of this study are available from the corresponding author upon request.

Conflicts of Interest: The authors declared that there is no conflict of interest in publishing this paper, and the authors confirmed that there is no conflict of interest for author and the co-authors.

Funding: This research was supported by 2023 Scientific research project of Anhui Provincial Education Department (Grant No. 2023AH051551) and the Opening Foundation of State Key Laboratory of Cognitive Intelligence, iFLYTEK (COGOS-2023HE02).

7. REFERENCES

[1] Baralt, M., Pennestri, S., & Selvandin, M. (2011). Using wordles to teach foreign language writing.

[2] Brown, K. A. (2022). MODEL, GUESS, CHECK: Wordle as a primer on active learning for materials research.npj Computational Materials, 8(1), 97.

[3] Havukainen, M., Laine, T. H., Martikainen, T., & Sutinen, E. (2020). A case study on co-designing digital games with older adults and children: game elements, assets, and challenges. The Computer Games Journal, 9, 163-188.

[4] Wardle J., 2022. "Wordle is a love story." Available at URL. (Accessed: 14 November 2022).

[5] Wang, Y., Chu, X., Bao, C., Zhu, L., Deussen, O., Chen, B., & Sedlmair, M. (2017). Edwordle: Consistency-preserving word cloud editing. IEEE transactions on visualization and computer graphics, 24(1), 647-656.

[6] Liu, C. L. (2022, August). Using wordle for learning to design and compare strategies. In 2022 IEEE Conference on Games (CoG) (pp. 465-472). IEEE.

[7] Bhambri, S., Bhattacharjee, A., & Bertsekas, D. (2022). Reinforcement Learning Methods for Wordle: A POMDP/Adaptive Control Approach. arXiv preprint arXiv:2211.10298.

[8] Benítez Gómez Á, Cavanillas Puga E. Wordle solving algorithms using Information Theory[J]. 2022.

[9] Anderson B J, Meyer J G. Finding the optimal human strategy for wordle using maximum correct letter probabilities and reinforcement learning[J]. arXiv preprint arXiv:2202.00557, 2022.

[10] de Silva, N. (2022). Selecting seed words for wordle using character statistics. arXiv preprint arXiv:2202.03457.

[11] Zeng, H. (2023). Finding Wordle Strategies That Can be Mastered by Humans. Highlights in Science, Engineering and Technology, 39, 714-719.

[12] Short, M.B., 2022."Winning Wodrle Wisely," arXiv preprint arXiv:2202.02148.

[13] Selby A., 2022. "The best strategies for Wordle (last edited on 17 March 2022)." Available at URL. (Accessed: 14 November 2022).

[14] Bertsimas D. and Paskov A., 2022. "An Exact and Interpretable Solution to Wordle." Available at URL. Preprint, received June 2022. (Accessed: 14 November 2022).

[15] Ho A., 2022. "Solving Wordle with Reinforcement Learning." Available at URL. (Accessed: 14 November 2022).

[16] Lokshtanov D., and Subercaseaux B., 2022. "Wordle is NP-Hard," arXiv:2203.16713.

[17] Rosenbaum W., 2022. "Finding a Winning Strategy for Wordle is NP-complete," arXiv:2204.04104.