Problem Chosen	2023 MCM/ICM Summary Sheet	Team Control Number
C		2321753

Summary

In order to model the difficulty of a given Wordle solution, a metric, called the Combined Normalized Difficulty Metric (CNDM), was created from the given data classifying past solutions as “Easy,” “Medium,” or “Difficult.” This metric ranges from 0.3 to 8.00 and is normalized around 1. Words with a score of 0.55 or below are classified as “Easy,” while any word above 1.29 is classified as “Difficult.” Any other value results in a classification of “Medium.” Then, a series of 6 attributes were defined and quantified that predict word difficulty. They are as follows: 1) the frequency of vowels, 2) the commonality of the word, 3) the commonality of the letters, 4) a word’s similarity to other possible solutions, 5) the number of unique letters, and 6) the frequency of bigrams. Once defined, 5 neural networks were created to predict each difficulty metric and its associated error, to predict the CNDM and its error, and to classify it as “Easy,” “Medium,” or “Difficult.”

While the neural network modeled the CNDM of an individual word, an algorithm was created to model player strategy and predict the distribution of guesses required to solve a given word. It used vocabulary size as the key parameter to test and validate

Next, a statistical model was created to predict player population and the proportion of players reporting in Hard Mode to the total number of player submissions. An exponential regression was fitted to a plot of total submissions over time and Hard Mode submissions over time. By examining outliers to the exponential regression, an inverse relationship between word difficulty and total player submissions was found. However, it was also found that word difficulty had a direct relationship with the number of players reporting in Hard Mode.

Finally, by examining the raw data, several interesting facts can be gleaned. One of which is that as time progressed, the words seemed to decrease in difficulty. This appears to reflect an increase in player skill rather than a decrease in word difficulty. Second is two outliers corresponding to major sporting events. One of which, the Super Bowl, saw a vast decrease in Hard Mode submissions without a decrease in total player submissions. The second is the Fifa World Cup, which saw a similar drop in total submissions without the expected Hard Mode decrease.

Table of Contents

Developing a Wordle Model

I. Word difficulty

In order to assess and predict the difficulty of a given contest based on the attributes of its associated word, we first had to define how to measure the difficulty of the contests in the dataset using the statistics that were provided. The three metrics we determined were the best measures of a contest's difficulty were the average length of the contest across all respondents, the proportion of long finishes, calculated by dividing the number of respondents that took 5 or 6 tries (above average) to get the correct word by the number of respondents that took 1-3 tries (below average), and the proportion of unfinished games. These metrics were normalized across all 359 contests in the dataset, added together, and divided by three in order to get the combined normalized difficulty metric (CNDM) centered around 1 and ranging between 0.3 and 8. By this metric, contests with greater CNDM values were considered to be more difficult based on our three difficulty metrics. The CNDM was then used to assign difficulty classifications to all 359 contests. Those in the bottom 20% of difficulty (<0.55 CNDM) are classified as "Easy", contests in the top 20% of difficulty (>1.29 CNDM) are classified as "Difficult", and all other contests are classified as "Medium".

We next had to determine which word attributes we would be measuring to predict difficulty and create metrics that could be used to quantitatively measure these attributes. After consulting some literature, we came up with six attributes that would have an effect on difficulty: the number of vowels in the word, the relative frequency of the word as it is used in the English language, the relative frequency of the letters as they appear in the list of approved Wordle responses, the similarity of the word to other words in the approved Wordle responses, the number of unique letters, and the relative frequency of the bigrams (pairs of letters) compared to other words in the approved Wordle responses [Smyth]. We also consulted the article Big Data in Little Wordle by Barry Smyth to get the formula for quantifying some of these attributes.
We hypothesized that each of these attributes would be negatively correlated with difficulty with the possible exception of similarity. Our reasoning was as follows: When a word contains a high number of vowels, there is an increased chance that any given guess will provide a clue, players will be more likely to guess a common word than an obscure word, letters that appear more frequently have an increased chance of being guessed correctly, players will be less likely to

guess a word if it contains doubles or triples of the same letter, and players will have an easier time figuring out a word if it contains common bigrams (e.g. if a word contains an “h”, players will reason there will be a “t” in front of it). We also predicted high similarity would be associated with a higher difficulty as it means that there are more words that could obstruct a player from getting the correct answer (e.g. a player with _rain would have to guess randomly between words such as brain, train, drain, grain, etc) Since word similarity correlates with letter frequency, it is possible that a higher similarity between words could lead to an easier solution.

Aside from the obvious methods (i.e. number of vowels and the number of unique letters), we will now describe the methods used to quantify each attribute. A word’s relative frequency score is the total usage of the word online (pulled from Kaggle) divided by the sum total use of every word and normalized to 1. The letter frequency score of a word is the average of the relative frequencies of each letter in the word normalized to 1. We calculated the similarity score using the Damerau-Levenshtein distance, a measure of the total number of substitutions or transpositions it takes to change an input word to a target word. While we quickly determined that having many adjacent words would make a word more challenging, we ultimately settled on the DL distance as a measure of similarity as it measures distance in a manner analogous to both types of Wordle clues (i.e. substitution, green/gray squares: transposition, yellow squares). The similarity score of a given word was equal to the number of words in the approved Wordle list which was a DL distance one or two away. This was divided by the sum total of similarity scores and normalized. Finally, the bigram frequency score was calculated in the same manner as the letter frequency score, but with letter pairs instead.

Figure 1: Attribute Calculations for Word (w) in Wordle list (W)

Relative Word Freq.:
$$\frac{WordCount(w)}{\sum_{i}^{|W|} WordCount(i)} * |W|$$

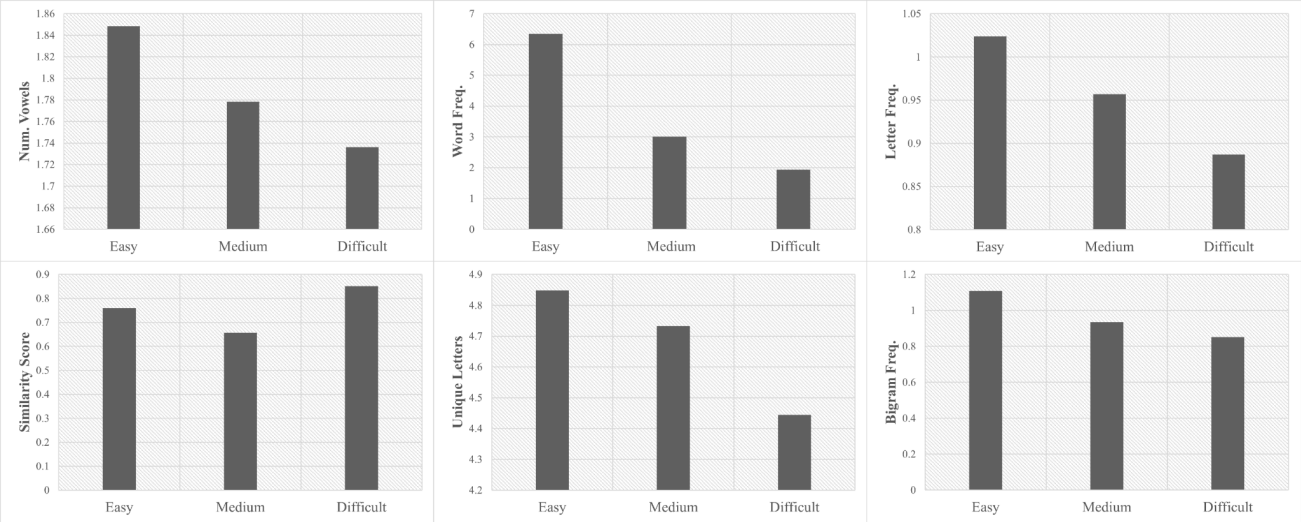
Relative Letter Freq.:
$$\frac{\sum_{l}^{|w|} LetterFreq(l)}{|w|} * \sum_{i}^{|W|} \frac{|i|}{\sum_{l}^{|i|} LetterFreq(l)} * |W|$$

Similarity Score:
$$\sum_{i}^{|W|} DLdist(w,i) * \frac{|W|}{\sum_{a}^{|W|}\sum_{b}^{|W|} DLdist(a,b)}$$

Bigram Freq.:

$$\frac{\sum_{l}^{|w|} BigramFreq(l)}{|w|} * \sum_{i}^{|W|} \frac{|i|}{\sum_{l}^{|i|} BigramFreq(l)} * |W|$$

Table 1: Average Vowel Count, Relative Word Freq., Similarity Score, Unique Letter Count, and Bigram Freq for each Difficulty Classification



These charts (Table 1) for each word attribute graph its average value among all members of each difficulty classification (e.g. letter frequency score of all “Easy” words), and they seem to indicate a strong negative relationship between most of the attributes and word difficulty. It is notable that there is a slight positive relationship between similarity score and difficulty. This lines up with our hypothesis that it would be only slightly positively related due to its collinearity with letter frequency.

We initially attempted to model the CNDM as a function of our six attributes using a multiple linear regression model generated in XLSTAT. While we were able to glean some useful information from this model, such as the coefficients of determination (R^2) for each attribute as well as their degree of collinearity (Table 2a), it was not accurate enough to give a reasonable prediction window for a word’s CNDM. We also generated regressions for the individual difficulty metrics comprising the CNDM in order to get more data on which attributes were most strongly correlated with each metric. From this, we were able to confirm our assumption that

similarity score and letter frequency were collinear, and also discovered that letter frequency and bigram frequency were collinear as well. We decided to keep these measures in the model as they still contributed to its predictive capabilities. We also made the discovery that, despite being weakly correlated with both average game length and long game proportion, the similarity score is strongly correlated with the proportion of unfinished games (Table 2b). This lends credence to the idea of the "one gray square effect" in which a large portion of the player base will fail a contest because they did not have enough information to determine the last letter in a word. All other attributes had very weak correlations with the proportion of unfinished games except for " number of unique characters", which had fairly strong correlations with every metric but had a notably weaker correlation with "unfinished games". We also found that our simple linear regression model was most accurate at predicting the proportion of long games, being able to explain ~36% of the variance

Table 2a: Correlation Matrix for CNDM

	Num Vowel	Relative Freq	Letter Freq	Sim Score	Unique Chars	Bigram Score	Wordle Number	CNDR
Num Vowel	**1**	0.017	0.445	-0.027	-0.052	0.156	0.072	-0.072
Normal Relative Freq	0.017	**1**	0.033	-0.010	0.041	0.057	-0.089	-0.091
Normal Letter Freq	0.445	0.033	**1**	0.321	-0.049	0.618	0.001	-0.205
Normal Sim Score	-0.027	-0.010	0.321	**1**	0.109	0.548	-0.051	0.213
Unique Chars	-0.052	0.041	-0.049	0.109	**1**	0.075	-0.039	-0.338
Normal Bigram Score	0.156	0.057	0.618	0.548	0.075	**1**	-0.028	-0.095
Wordle Number	0.072	-0.089	0.001	-0.051	-0.039	-0.028	**1**	-0.029
CNDR	-0.072	-0.091	-0.205	0.213	-0.338	-0.095	-0.029	**1**

Table 2b: Correlation Matrix for Proportion Unfinished

	Num Vowel	Normal Relative Freq	Normal Letter Freq	Normal Sim Score	Unique Chars	Normal Bigram Score	Wordle Number	Prop. Unfinished
Num Vowel	**1**	0.017	0.445	-0.027	-0.052	0.156	0.072	-0.058
Normal Relative Freq	0.017	**1**	0.033	-0.010	0.041	0.057	-0.089	-0.044
Normal Letter Freq	0.445	0.033	**1**	0.321	-0.049	0.618	0.001	-0.081
Normal Sim Score	-0.027	-0.010	0.321	**1**	0.109	0.548	-0.051	0.334
Unique Chars	-0.052	0.041	-0.049	0.109	**1**	0.075	-0.039	-0.184
Normal Bigram Score	0.156	0.057	0.618	0.548	0.075	**1**	-0.028	0.059
Wordle Number	0.072	-0.089	0.001	-0.051	-0.039	-0.028	**1**	-0.039
Prop. Unfinished	-0.058	-0.044	-0.081	0.334	-0.184	0.059	-0.039	**1**

in this metric, while it was least accurate at predicting the proportion of unfinished games, explaining only ~21% of the variance. This squares nicely with our previous observation that very few attributes showed a strong correlation with unfinished games.

To increase the accuracy of our model, we decided to create five neural networks: 3 predicted each individual difficulty metric and calculated the associated error, 1 predicted the combined difficulty metric and calculated the associated error, and 1 categorized words directly into their associated difficulty classifications. These neural networks were based on code written by Jason Brownlee and shared in the articles *Prediction Intervals for Deep Learning Networks* and *Neural Network Models for Combined Classification and Regression*, which was then altered to optimize and fit our needs. Each neural network contained one input layer with six nodes for our six attributes, one hidden layer with four or five nodes for the numerical and qualitative predictors respectively, and one output layer with one or three nodes for the numerical and qualitative predictors respectively. The numerical models were trained on 95% of the Wordle data given, while the qualitative model was trained on 80% of the data given. The learning rate was kept at 0.01, while the batch size and the number of epochs were determined via trial and error in order to produce the most accurate model (estimated by minimizing the mean absolute error (MAE). Ultimately, 800 epochs with a batch size of 4 were settled on for qualitative predictions, while 300 epochs with a batch size of 16 were settled on for numerical predictions. 95% confidence intervals were generated for quantitative outputs using a Gaussian predictive interval determined by generating 30 distinct models and taking the mean and standard deviation over the predictions from all models [Brownlee]. The interval was calculated as $\mu \pm 1.96\sigma$. The average mean absolute errors among all 30 models for each of the numerical models were 0.18 for length, 1.4 for the proportion unfinished, 0.55 for the proportion of long finishes, and 0.29 for CNDM. Their average interval sizes were 0.42, 2.58, 1.36, and 0.61 respectively. The prediction intervals are all reasonable compared to the range of their corresponding metrics which are 2.9 for length, 48 for the proportion unfinished, 17 for the proportion of long finishes, and 7.7 for CNDM. It is still far too broad of a prediction window to predict all but the most extreme cases confidently as these interval sizes are greater than the standard deviations of the values they are trying to predict. Ultimately, we must conclude that more training data would be required to generate quantitative prediction models with a reasonable degree of precision.

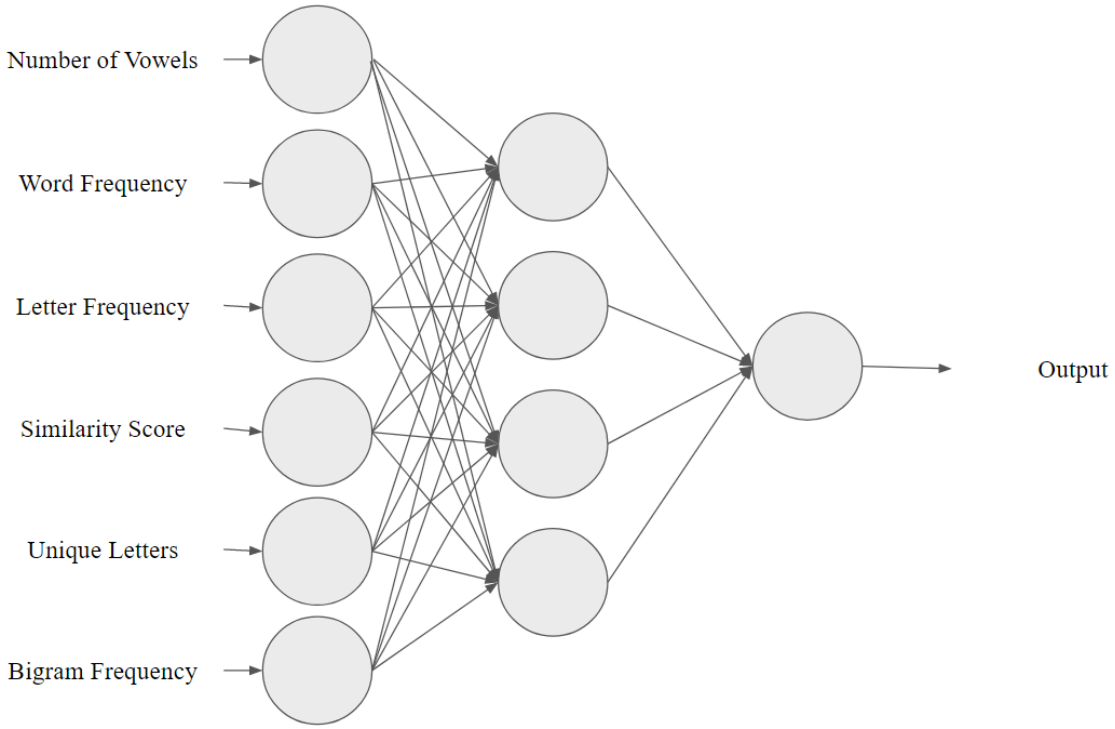


Figure 2: Neural Network Structure for Quantitative Predictions

The model for predicting the qualitative classification of each word as either “Easy”, “Medium”, or “Difficult” was more promising, displaying an average 70% success rate for classifying the 70 target words over 40 distinct models. This is significantly more accurate than guessing the classification randomly which would give a 33.3% success rate, and it is also far more accurate than guessing strategically. Because guessing “Medium” all the way down would yield a 60% success rate, the Z-value of this success rate is $Z = \frac{.7 - .6}{\sqrt{\frac{.6*(1-.6)}{50}}} = 1.44$. This gives a p-value of 0.075, meaning that, at an alpha-value of 0.1, there is a statistically significant advantage of our model over any method of guessing which uses no information about the attributes of the word. Thus we can reject the null hypothesis that a word’s difficulty classification is not related to its attributes, which means we conclude that our hypothesis that one can predict a Wordle’s difficulty with its word’s attributes was correct.

II. Guessing Algorithm

Developing a wordle-solving algorithm that achieves perfect performance is relatively straightforward; however, creating an algorithm that mimics human play is more complicated. By interpreting human play patterns into a simulation and adjusting the parameters of the size of its vocabulary and starting word selection, our algorithm can predict the distribution in a way that explains 89% variance in the data.

To better simulate human behavior in Wordle, our algorithm is designed to choose its guess like a human rather than resorting to the perfect implementation of information theory. Our algorithm determines its next guess by removing every word it knows that fails the four basic deductions that can be made about the solution word based on the clue it is given, and then selects the most common word. This process is analogous to a human picking the first word that comes to mind that fits.

Competence at Wordle is largely determined by a player's vocabulary. Players who have a smaller diction resort to less optimal play as they are unable to think of a word that fits all the constraints of the clue. We interpreted this in our algorithm by having a parameter that limits the word bank the simulation gets before it goes into frustration mode. When there is no guess that satisfies all the logical constraints and the solution word is not in the algorithm's vocabulary, it enters frustration mode. Entering this mode it gains access to the entire word bank but loses some ability to evaluate a word by its clues. This mode mimics a human player resorting to guessing words it knows aren't correct in a desperate attempt to gain new information. When entered, it will always get to the solution word given enough time. However, it will seldom reach a solution within 7 guesses if it enters frustration mode too early or it does not receive the necessary clues.

Simulations that include every possible start word with the same number of trials resulted in players outperforming the simulations in the first couple of turns. This conclusion makes sense for two reasons. The first reason is that players who get especially impressive scores are more likely to want to show their friends, so there is a survivorship bias for lower scores. However, in addition to that, players tend to choose highly effective starting words, as it is not difficult to look up the mathematically determined best-starting words. To account for this, we evaluated every word and ran more trials for the starting words we believed would be more popular.

Rather than random starting words, we selected 2 criteria for how humans choose their starting words: word commonness and letter frequency. These two metrics are combined to create a word's starting score.

Figure 1: Calculation for Starting Score

$$Starting\ score = \frac{1}{\sqrt{Word\ commonness}} \cdot \frac{1}{\sqrt{Letter\ frequency}}$$

The word commonness value is determined by dividing a word's rank in descending order of commonness by the total number of words in the list. (so *about* has a score of 1 and *toits* has a score of .0001)

We determined the letter frequency value of a word by the summation of every letter's probability of showing up in the answer word divided by the theoretical best letter frequency value. (in this case, *soare*) As per the New York Times's official list of most common starting words, the two most common words are adieu and audio. We see that players over-evaluate the information gathered from vowels, and so under-evaluate the frequency of the most common consonants. Also, players are aware that repeated letters do not convey as much information as the first letter, so there is also a penalty for having repeated letters.

By taking the square root of and multiplying the two values, we have an evaluation that synthesizes the two considerations into a singular number. Note that due to the properties of fraction multiplication, this formula outputs a normalized number that punishes words that only do well in one of the two criteria. Through this process, words given a high starting score are determined to be both highly effective and common.

starting word	word commonness score	letter frequency score	starting score
Toits	.0001	.7199	.0084
Mamma	.7317	.2533	.4305
Raise	.9362	.9774	.9566

Table I

After determining the starting score for every word, then we multiplied the starting score by 3, rounded the resulting number, and then ran that number. The resulting scores of the formula align with our intuition about what words we can expect players to use. For example, *toits* is a great starting word but it never gets trial because the word is too obscure for a player to ever use it. Mamma is a poor starting word but because it is a common word, it is given 1 trial. The word *raise* is exactly the sort of word we expect players to use as their starting word. This aligns with what the new york times posted as the most common starting words, as *raise* is fourth on the list. Even though there are more than 3 times as many players who use *raise* than players that use *mamma*, this equation accounts for starting word preference better than every word given the same amount of trials.

Training The Model:

After running simulations at various vocabulary levels, we can calibrate our model to approximate the skill of the player base. Differences in success between vocabulary completeness tend to be more exaggerated at higher guess counts. Simulations at 70%, 90%, and 100% vocabulary completeness, found an incredibly close distribution between the player base and 90% vocabulary completeness. The R^2 between the player data and the 90% vocabulary completeness was at **95.80%**

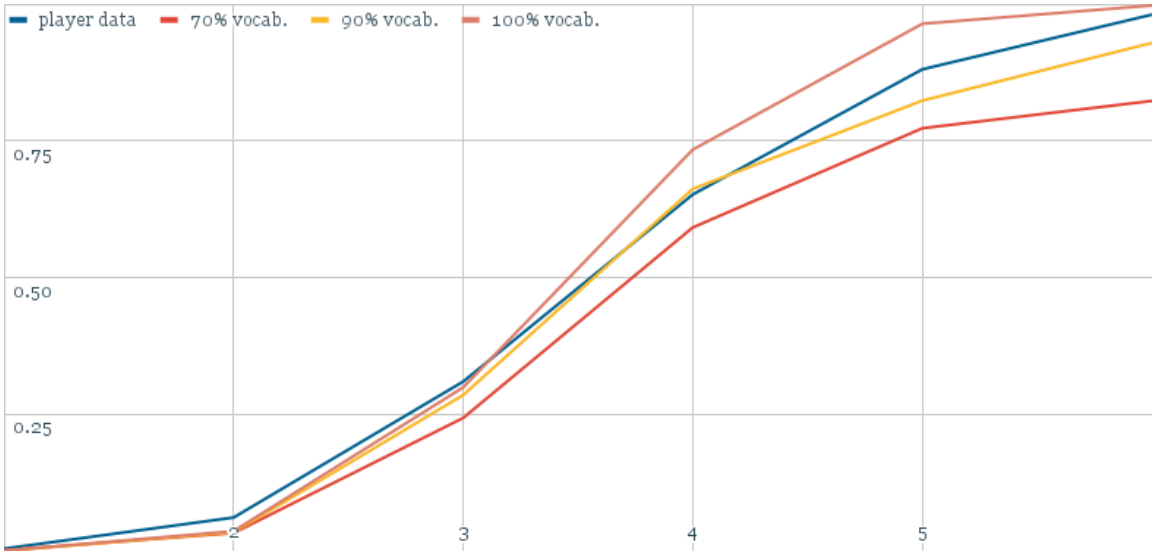


Figure 2: Model Calibration via Vocabulary Completeness Paratermeter

Te

Validating the model:

Even at optimal play by our simulation, we have found that player data outperforms our model. As mentioned before, our speculation is that players tend to be more inclined to post their game if they guess the correct word in one or two attempts. With similar reasoning, we also believe that game losses are unreported as players might not want to share their unimpressive results. Focusing on the section of the guess distribution between 3-5 where we can expect little influence of survivorship bias, we have found a **99.8% R^2** between our model and the player data. Even more incredibly, as you will see later in this paper, the high explanation in variation does result in an unbelievable predictive power.

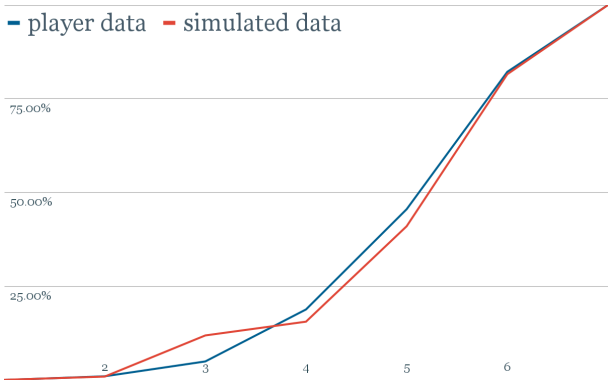
III. "Eerie" Predictions

A cursory glance at the word "eerie" with the knowledge of how each attribute is related to difficulty seems to indicate that it should be above average in difficulty as the number of unique letters is by far the highest-correlated attribute and "eerie" only has 3. Granted, it also has a very high relative letter frequency score, but due to the fact that this attribute is generally less impactful than unique letters, we hypothesize that it would be at least above average in difficulty. Using the quantitative neural network models to predict the difficulty of the word "eerie", we obtained its associated length, the proportion of respondents who did not guess correctly, the proportion of long responses, and CNDM. Respectively they are 4.36 guesses, 2.674%, 2.585, and 0.967. These results are somewhat strange as the average length, proportion incorrect, and proportion of long responses are all greater than the average for their corresponding metrics, indicating that "eerie" should be more difficult than the average Wordle target word, but the CNDM is lower than average, seemingly contradicting the metrics from which it is derived. This can be explained when looking at the 95% confidence intervals generated by these models, which are [3.575, 5.097] for length, [-0.672, 6.019] for the proportion incorrect, [0.684, 4.486] for the proportion of long games, and [0.144, 1.791] for CNDM. The ranges of these intervals are over double the average confidence interval ranges for their corresponding models, meaning it is not possible to determine the difficulty of "eerie" from the quantitative neural networks. One reason for the increased uncertainty in these predictions is because of the unique nature of the word. Given that "eerie" is a statistical outlier in both its number of unique letters and letter frequency score, it seems likely that the word was so far out of the models' training dataset that the precision in its prediction went drastically down. This would be analogous to extrapolating a line of best fit far outside of the known data range. However, we already knew that the numerical models would probably not be precise enough to classify words accurately based on their CNDM, so we primarily chose to generate results for eerie to see if their 95% prediction intervals contain the values/implied values generated by the other models.

After running the word through the classification neural network, which has an accuracy of 70%, we found that "eerie" should be classified as "Difficult." Due to the concern that this model would face a similar precision issue as the numerical models, a 95% confidence interval was generated for the classification prediction using a Gaussian predictive interval determined by using the 40 distinct models and taking the mean and standard deviation of the predictions from

all models (this is possible because the difficulty is classified by predicting a numerical value associated with difficulty classification and rounding to the closest integer– 0 = ”Easy”, 1 = ”Medium”, 2 = ”Difficult”). Our confidence interval for “eerie” ranged from 1.523 to 2.377, both of which round to 2. This means we are 95% confident that our model captured the true difficulty classification for the word. Furthermore, when testing five other words (leave, parer, class, mummy, and fluff), all of which were either outliers in unique letter count or letter frequency score or a combination of the two, the model correctly identified 4 out of the 5, which further indicates that it is still precise when working with words in this range. Additionally, the cutoff for a “Difficult” classification is contained in our confidence interval from the numerical models, so these models agree in that regard even though the classification model is far more precise.

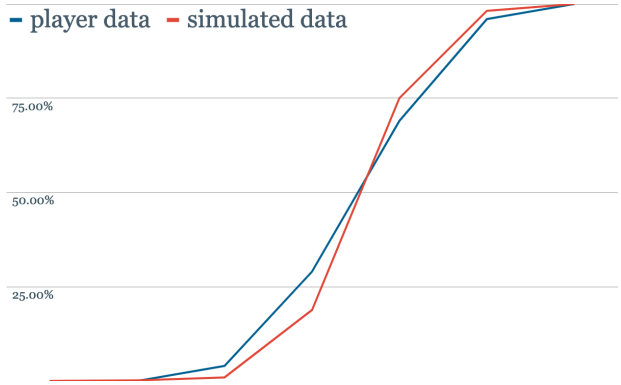
When these outlier words, *mummy* and *fluff,* were run through our distribution algorithm, we found that it corroborated the conclusion we made with the classification neural network.



Mummy

R^2 = 0.9166055327

Figure I



Fluff

R^2 = 0.943507971

Figure II

At 90% vocabulary and our focused starting word, our model is able to explain 95.80% of the variation in the data and can accurately recreate the distribution of a given word in the player data. Figure I shows its results when run 2134 times attempting to guess the word “eerie.”

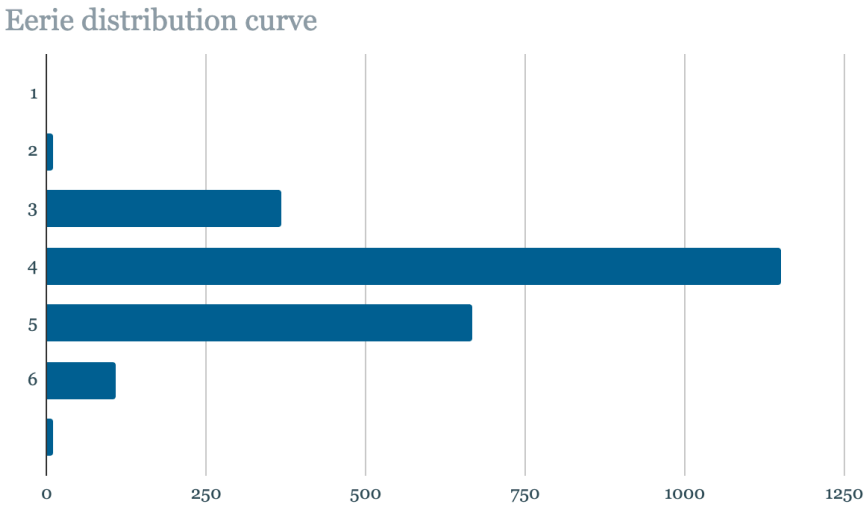


Figure I

Twitter Data Analysis

I. Player Variation

Plotting the number of self-reports day by day, it is easily seen that Wordle’s novelty led to a sharp increase in the player base during its first few months of publication. The ease of reporting one’s score to social media was an incredibly effective marketing strategy that led to an all-time peak of 360,000 players reporting their results. Once the novelty wore off, one can observe the number of players reporting their scores decreasing exponentially. Time is easily the most important factor in determining how many scores will be reported to Twitter, however, there are points where the number of reports suddenly increases before rejoining the standard curve. Examples include February 22, 2022, where 22,000 more results were reported than either surrounding day, and May 4, 2022, which saw a similar increase of 15,000 compared to the days around it. The word for each day was “thorn” and “train” respectively. Both of these words are classified as “easy” with a CNDM score of .34 for “thorn” and .29 for “train.”

While the number of players reporting in Hard Mode follows a similar decay pattern to the total number of players, its own outliers follow a different pattern. While the total number of

Twitter reports increase on easier words, those reporting specifically in Hard Mode reported more often on difficult words. Of note are the days of March 11, 2022, and September 16, 2022. “Watch” was the word of the day on March 11th and led to nearly 3000 extra Hard Mode reports despite 20% of players being unable to solve it leading to a CNDM score of 3.17. For September 16th, the word “parer,” with a CNDM score of 8.00, left 48% of players without a solution, and yet over 1,000 more players reported their Hard Mode score. It appears that while standard players are more likely to report a successful score, players in Hard Mode report more often during especially challenging days.

In order to predict how many players will report on any given day, an exponential regression would be most effective. The data provided was imported into MATLAB, where the total number of reports was plotted in figure A and the number of submissions in Hard Mode was plotted in figure B. Using MATLAB’s inbuilt trendline function, an exponential regression equation was obtained for each plot, displayed in Figures Aa and Ba. The data’s formatting results in the first point, January 7th to correspond to 3 on the x-axis and the final point, December 31st, to be 361 on the x-axis. Thus in order to predict the number of player reports for a given day, January 1st, 2023 should correspond to $x = 362$ and so on.

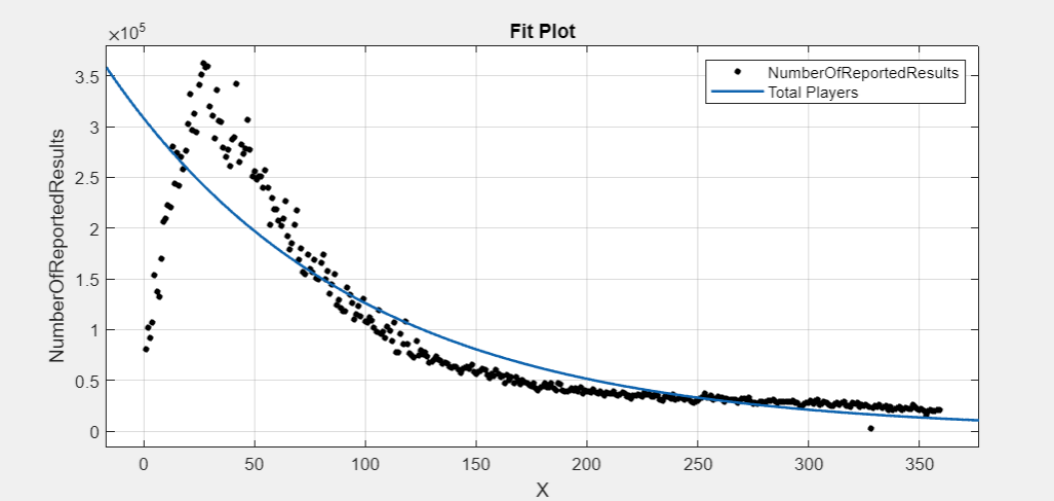


Figure A

$$y = 360000e^{-0.008x}$$

Figure Aa

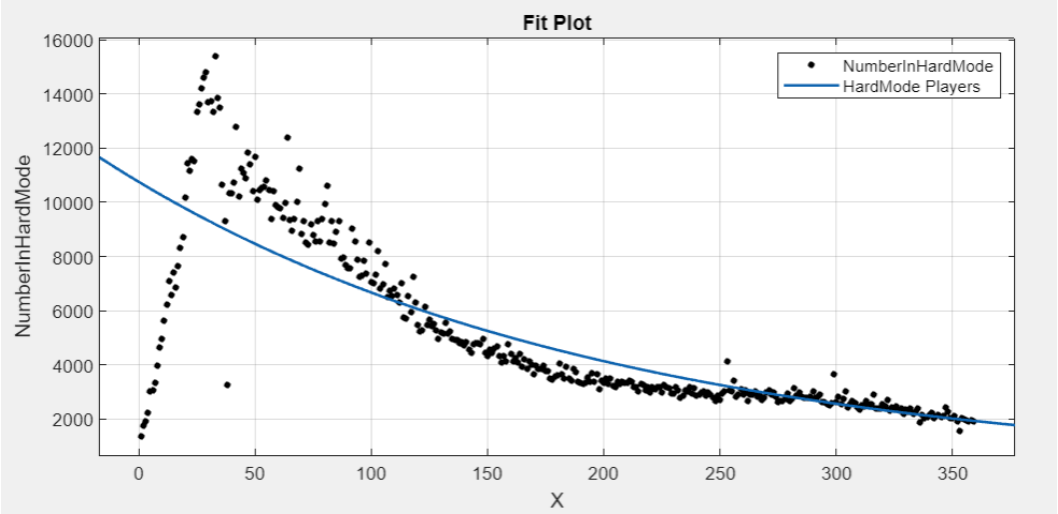


Figure B

$$y = 10750e^{-0.00476x}$$

Figure Ba

March 1st, 2023, would correspond with $x = 421$ and following the regression equation, approximately 12,405 players will report their scores, and 1,449 of those will be reporting in Hard Mode, assuming the word of the day will be of average difficulty. Due to both the high number of data points and the exponential regressions' inability to account for Wordle's rising popularity in its first two months, the Sum of Squared Estimate of Errors and the Root Mean Square Error are both quite high, at 5.61(10^11) and 3.964(10^4) for the total number of reports and 1.019(10^9) and 1882 for the number of reports in Hard Mode. However, these trendlines do account for much of the variation in the data with an Adjusted R-square value of 0.8028 for the total number of reports and 0.647 for the number of reports in Hard Mode. As such, these models can be trusted with a 95% confidence bound.

II. Noteworthy Insights

One noteworthy data insight comes from our linear regression analysis of the relationship between specific word attributes and difficulty. In this analysis, we included one factor that was not a word attribute and this was the contest number. We included this to make sure that the popular notion that Wordle has gotten more difficult over time did not impact our model. However, what we found was that not only did the contest number have very little correlation with difficulty, but what little correlation it did have was negative, meaning that Wordle has actually gotten ever so slightly easier over time. This is likely due to the player base becoming more familiar with the game and developing effective strategies rather than the words themselves becoming easier.

Another insight comes from the two largest outliers in Twitter's submitted results. On November 30, 2022, there were 20,000 fewer results than the two surrounding days. However, this same submission drop-off is not reflected in the Hard Mode players. In fact, they submitted 100 more submissions than the preceding day and 200 more than the following day. Observing cultural events surrounding the 30th most notably is the 2022 Fifa World Cup. Although the World Cup took place from the 20th of November to the 18th of December, most major matches took place between the 30th and the 1st. Accounting for the time difference between Qatar and the United States, this lines up with the massive decrease in player submissions. A similar incident occurred on February 13, 2022, but the positions were reversed. Although the total number of submissions went up by approximately 10,000, the number of submissions in Hard Mode decreased by 6,000. Examining the 13th for cultural events reveals that the Superbowl took place that same day. This leads to an interesting correlation between soccer fans and total Wordle submissions and American football fans and Hard Mode Submissions.

Conclusion

By using data acquired from Twitter, a neural network was created that was designed to predict how difficult any given word should be. It was based upon a Combined Normalized Difficulty Metric created from the ratio of “long” games of Wordle to “short” games. This metric was then compared against 6 quantified word attributes relating to an individual word’s difficulty. They are as follows: 1) the frequency of vowels, 2) the commonality of the word, 3) the commonality of the letters, 4) a word’s similarity to other possible solutions, 5) the number of unique letters, and 6) the frequency of bigrams. The neural network was able to correctly identify a word difficulty rating with a 70% success rate.

Similarly, the number of guesses a player will take to solve any given Wordle can be modeled by an algorithm trained to simulate human habits and tendencies. With metrics based on the commonality of starting words, the effectiveness of starting words, and the common player’s vocabulary. Using these parameters, the algorithm was able to model human guess distribution with 95.8% accuracy

Finally, the player population was modeled using an exponential regression. It was discovered that time was the greatest factor in the variation in player submissions. However, the difficulty of the word inversely affected the total number of submissions while increasing the proportion of Hard Mode submissions.

The models were tested to discover the difficulty of the word “eerie.” The neural networks correctly identified it as “Difficult,” and the player simulation algorithm presented a guess distribution in line with our hypothesis. When compared to the results of the population model, these results will provide an above-expected number of Hard Mode submissions while the number of total submissions will be less than our predicted value.

Letter to the Editor

To the Puzzle Editor of the New York Times:

I hope this letter finds you well. The research team I am a part of has found your work especially invigorating over the past year. It is our hope that the findings of our research help you to further improve the quality of your already fine work. Specifically, our team has been researching the different factors that influence Wordle's difficulty as well as modeling the variation in the player base over the past year.

Firstly, as to what makes a word difficult, we determined there were 6 major factors that attributed to a word being difficult: 1) the number of vowels, 2) the commonality of the word, 3) the commonality of the letters, 4) the number of unique letters, 5) the number of bigrams, and 6) its similarity to other possible answers. After deciding upon these factors, we created a metric to judge how difficult any given word should be based on submission data acquired from Twitter. This metric is called the Combined Normalized Difficulty Metric or CNDM. Using the CNDM, we determined which words in our data set should be classified as “Easy,” “Medium,” or “Difficult.” Then it was a simple matter of comparing the qualities of each word with its corresponding difficulty rating. It should come as no surprise that more common letters and words lead to “Easy” words as well as a high frequency of vowels and bigrams. When it comes to what makes a word difficult, its similarity to other possible solutions is correlated with higher difficulty as is the number of repeated letters.

Once we had a measure of what made a previous word difficult, we created an algorithm to predict how difficult players would find future words. For example, how difficult would players find the word “eerie.” With a highly calibrated player simulator, we ran over two thousand times, we feel confident that 37% of the player base will take more than four attempts to guess “eerie.” This is in sharp contrast to the expected portion of players for an average difficulty word at only 12%. Due to its unique characteristics and overlapping difficulty metrics,

“eerie” was not only classified as a difficult Wordle solution, but also a difficult test of our algorithm’s capabilities.

Finally, we wished to see how players responded to words of varying challenges. I mean no offense as this is a fact for any game, but as time progresses, the player base decreases and Wordle is no exception. However, by modeling the variation in the number of scores submitted to Twitter, we were able to determine what caused different player demographics to either report or withhold their scores. I am sure you are well aware that an easier solution leads to an increased number of Twitter reports, however, it is worth noting that when the Wordle solution was particularly difficult, we saw an increase in the number of players reporting their Hard Mode scores. Also worth noting is the demographic of players reporting in Hard Mode as opposed to the total player demographic. Two extreme outliers were detected on February 13th and November 30th which saw a sharp decrease in Hard Mode players and the total player base respectively. However, it appears these decreases were independent of one another as the decrease in Hard Mode players saw no decrease in the total player base and vice versa. Upon looking into these dates further, we saw that they corresponded with the Super Bowl and the Fifa World Cup. This led us to believe that there is a strong correlation between soccer fans and total submission and between American football fans and Hard Mode submissions. Thus in order to optimize Wordle players reporting their scores to social media, we believe that you should aim to post high-difficulty words during major soccer events and similarly, during major football events you should aim to post easier solutions.

Once again, I would like to offer my appreciation for the work you do and all of us on the research team hope that you will find our data just as enlightening and captivating as we did ourselves.

Sincerely,
MCM Team 2321753

Appendix A

Python Code:

https://www.dropbox.com/scl/fo/epfbw1bxg83kka2u953ty/h?dl=0&rlkey=pd432fanzqcbe693yexsnahvt

References

Brownlee, J. (2021, January 31). *Prediction intervals for Deep Learning Neural Networks*. MachineLearningMastery.com. Retrieved February 20, 2023, from https://machinelearningmastery.com/prediction-intervals-for-deep-learning-neural-networks/

Brownlee, J. (2021, March 28). *Neural network models for combined classification and regression*. MachineLearningMastery.com. Retrieved February 20, 2023, from https://machinelearningmastery.com/neural-network-models-for-combined-classification-and-regression/

Smyth, B. (2022, June 1). *Big Data in little wordle*. Medium. Retrieved February 20, 2023, from https://towardsdatascience.com/big-data-in-little-wordle-306d5502c4d9

Tatman, R. (2017, September 6). *English word frequency*. Kaggle. Retrieved February 20, 2023, from https://www.kaggle.com/datasets/rtatman/english-word-frequency

Hoffman, H. (2022, December 23). *Finding the best Wordle opener with Machine Learning*. Medium. Retrieved February 20, 2023, from https://towardsdatascience.com/finding-the-best-wordle-opener-with-machine-learning-ce81331c5759

Amlen, D. (2022, September 1). *Few wordle players use consistent starting words, but when they do, it's adieu*. The New York Times. Retrieved February 20, 2023, from https://www.nytimes.com/2022/09/01/crosswords/wordle-starting-words-adieu.html