Journal of Interdisciplinary Science Topics

How Good Is Your First Wordle Guess?

Sophia Bennett
Natural Sciences (Life and Physical Sciences), School of Biological Sciences, University of Leicester
03/03/2025

Abstract
Wordle is a 5-letter word guessing game used by millions daily. Debates have been sparked as to which word is the most effective first guess. This paper poses a simpler new score-based model that can mathematically judge how good your guess is. Have you been fooled by an incorrect suggestion online?

Keywords: *Viral Game; Computer Game; Probability; Python; Wordle*

What is Wordle?
Wordle is a web-based game designed to utilise the player's spelling and mental dictionary [1]. A five-letter word must be guessed within six tries. Each guessed letter results in three different colours:

- ***Grey:*** the letter is not in the word
- ***Yellow:*** the letter is in the wrong location
- ***Green:*** the letter is in the correct position

There has been a debate between players regarding which word is the optimum to start with since the game began. Researching online will result in multiple suggested words, but a lot are based on the percentage chance of solving before all the tries are taken (Table 1). This is not necessarily the best way to measure as every player will guess different following words that result in different outcomes.

Crane	Based on an incorrect model due to code bugs that went viral. This model results in an average score between 1 and 6 where crane achieved [2]
Tales	Based on the percentage chance of solving by an individual (success rate 95%) [3]
Slate	Best Based on Wordlebot AI that works out percentage chance of solving (success rate 99%) [4]

Table 1 – Popular and viral starting Wordle words

The Scoring System
This paper proposes a simple new scoring system based on how common a letter is in each place of a word in all previous Wordles to decide how good a starting word really is. It is based on the most green and yellow letters in the guess and will not be related to the ability to solve. The reasoning behind focusing on letter frequency in each position is to increase the likelihood of correctly guessing letters early in the game. Higher frequency letters in specific positions will likely appear in more words. By assigning a score to each letter based on its frequency at each position, the system can rank potential starting words. Using Python allows for efficient processing and searching within the large valid Wordle dictionary [5].

For this study, the probability of each letter being in each position of each word is based on the first Wordle to 31/12/2024 [6]. It is assumed the valid word dataset will not change as of 31/12/2024. This model is based on non-contextual usage, meaning the frequency of letters is the sole determining factor for the optimal starting word, without considering any other factors that may affect letter probabilities in words. For example, ignoring any potential letter pairings that might be more commonly found together, such as 'qu'. The frequency of each letter in each position was calculated (Appendix A) by utilising the COUNTIF(*****) command in Excel. This number represents the letter score (Table 2). This data was then transported into a dictionary within Python, along with the Valid-Words dataset (Appendix B).

Example Word List	1st letter Score
Brown Flake Sheep Blame	COUNTIF(B****) This results in B=2 as B appears twice at the 1st letter.

Table 2 – Example word list including 4 5-letter words to demonstrate the method of counting to give a first position letter score for B.

Functions were created to calculate word scores based on letter scores. Scores were then sorted to find the highest-ranking words.

Results

Based on the scoring system, 'crane', one of the most viral suggested words, scored 825. This was originally stated by Grant Saunderson, who used a model based on only a first guess with no reliance on following guesses [2]. Later, he corrected himself and stated 'soare' (with his score of 5.89/6) after coding errors were fixed. The highest scoring word using this new model is also 'soare' with 908. This confirms the method is reliable. Multiple other reliable programmers also listed 'soare' as the best, but all have different subsequent answers to the ones generated by this new method [4]. Comparatively, the lowest scoring word was 'imshi' with a score of 98. This word was also suggested by an online user's model after alteration [7]. This further confirms the new scoring system works. However, no single model has confirmed 'soare' and 'imshi' as the definite best and worst guesses, suggesting this is a new unique model. This less complex model is not only easier for the average player to understand, but they can also work out the score of any word they wish to try (Table 3). The mean score of a valid Wordle word is 445.41 (Appendix B). Using this scoring system, players can see if they have fallen for invalid viral suggestions and even see what score their unique go-to starting word is.

A website made by @dharmesh allows users to put in their chosen word and receive a rank for that input ('percentile score') [8]. However, the website does not base the system on any mentioned source. Additionally, the website also gives an incorrect lowest word as 'imshi' (the mathematically backed up worst word) only places 12926 out of 12974. This backs up the idea that the scoring system created in this paper is unique and more reliable.

Letter	First Letter	Second letter	Third letter	Fourth letter	Fifth letter
A	83	167	196	92	37
B	102	9	26	9	8
C	107	24	37	86	22
D	51	8	42	42	63
E	44	136	96	166	259
F	79	4	11	19	14
G	67	12	34	41	24
H	45	90	6	18	81
I	19	106	146	89	4
J	11	2	2	1	0
K	13	7	6	37	57
L	50	109	54	92	87
M	66	20	27	39	22
N	18	56	79	109	72
O	20	158	155	67	38
P	76	31	33	24	27
Q	11	4	1	0	0
R	52	154	94	98	112
S	197	12	37	87	19
T	90	52	59	72	157
U	23	83	89	48	2
V	23	9	23	28	0
W	42	25	12	16	10
X	0	10	8	2	4
Y	5	7	19	2	175
Z	2	1	4	12	2

Table 3 – Scoring sheet for each letter in each position of a 5-letter word.

Conclusion

This new simple scoring method can be used not only to judge incorrect viral suggested first words but is accessible enough for anyone to use to judge any word. Other models available online which may be correct, are also complex and hard for a non-programmer to understand. This new model seeks to overcome this. Scores using this model will be in the ranges of 98 to 908 and a mean score of 445.41.

References

[1] The New York Times (2021) *Wordle*, [Computer game] Web-based, The New York Times Games, New York City

[2] 3Blue1Brown (2022) *Oh, wait, actually the best Wordle opener is not "crane"*, YouTube Video, Available at: https://www.youtube.com/watch?v=fRed0Xmc2Wg [Accessed: 28 February 2025]

[3] Smyth, B. (2022) *What's the best word to start Wordle? Data Analytics can tell you.* Available at: https://www.sfi.ie/research-news/news/wordle-data-analytics/ [Accessed: 28 February 2025]

[4] McLaren, M. (2023) *Best Wordle start words to help you keep your streak*, Toms' Guide, Available at: https://www.tomsguide.com/news/best-wordle-start-words [Accessed: 28 February 2025]

[5] Dracos (2023) *valid-wordle-words.txt*, GitHub Gist, Available at: https://gist.github.com/dracos/dd0668f281e685bad51479e5acaadb93 [Accessed: 28 February 2025]

[6] Fiveforks.com (2025) *Past Wordle Words – Alphabetical Answers List*, *No Spoiler*, Available at: https://www.fiveforks.com/wordle/ [Accessed: 28 February 2025]

[7] Nguyen, J. (2022) *You're not a true 'Wordle' connoisseur until you start with the scientifically proven worst word*, Mashable, https://mashable.com/article/worst-wordle-word [Accessed: 28 February 2025]

[8] Dharmesh (2024) *FirstWord For Wordle*, WordPlay. https://wordplay.com/firstword [Accessed: 28 February 2025]

Appendix A:

	Frequency					
Letter	**First Letter**	**Second letter**	**Third letter**	**Fourth letter**	**Fifth letter**	**Total each letter**
A	83	167	196	92	37	575
B	102	9	26	9	8	154
C	107	24	37	86	22	276
D	51	8	42	42	63	206
E	44	136	96	166	259	701
F	79	4	11	19	14	127
G	67	12	34	41	24	178
H	45	90	6	18	81	240
I	19	106	146	89	4	364
J	11	2	2	1	0	16
K	13	7	6	37	57	120
L	50	109	54	92	87	392
M	66	20	27	39	22	174
N	18	56	79	109	72	334
O	20	158	155	67	38	438
P	76	31	33	24	27	191
Q	11	4	1	0	0	16
R	52	154	94	98	112	510
S	197	12	37	87	19	352
T	90	52	59	72	157	430
U	23	83	89	48	2	245
V	23	9	23	28	0	83
W	42	25	12	16	10	105
X	0	10	8	2	4	24
Y	5	7	19	2	175	208
Z	2	1	4	12	2	21
Total no. words	1296	1296	1296	1296	1296	
Most Frequent Letter	S	A	A	E	E	
2nd Most	C	O	O	M	Y	
Least Frequent Letter	X	Z	Q	V/Q	V/Q/J	
2nd Least	Z	J	J	J	U/Z	

	Highest Scoring Word...Letters			
Most common Letters	..with top 5..	..with top 6..	..with top 7..	..with top 8..
E,A,R,O,T, L, I, S	RAREE (738)	LEARE (739)	TIARE (749)	SOARE (908)

Top 5	**Used before?**	**Bottom 5**	**Used before?**
SOREE	0	OGHUZ	0
SLANE	0	IMSHI	0
SLARE	0	XYSTI	0
SHANE	0	ENZYM	0
SEINE	0	OXBOW	0

0 = has not been a wordle before

Table A – Data table showing the frequency of each letter in each position, best and worst letters for each position, most common letters overall and the top 5 and bottom 5 Wordle valid words based on the scoring system.

Appendix B:

Python Code used to search the dictionary of Wordle valid words with annotations '#' to describe the main steps and python response.

```
#Dictionary of letter scores

import pandas as pd
letter_scores = {
    "A": [83, 167, 196, 92, 37],
    "B": [102, 9, 26, 9, 8],
    "C": [107, 24, 37, 86, 22],
    "D": [51, 8, 42, 42, 63],
    "E": [44, 136, 96, 166, 259],
    "F": [79, 4, 11, 19, 14],
    "G": [67, 12, 34, 41, 24],
    "H": [45, 90, 6, 18, 81],
    "I": [19, 106, 146, 89, 4],
    "J": [11, 2, 2, 1, 0],
    "K": [13, 7, 6, 37, 57],
    "L": [50, 109, 54, 92, 87],
    "M": [66, 20, 27, 39, 22],
    "N": [18, 56, 79, 109, 72],
    "O": [20, 158, 155, 67, 38],
    "P": [76, 31, 33, 24, 27],
    "Q": [11, 4, 1, 0, 0],
    "R": [52, 154, 94, 98, 112],
    "S": [197, 12, 37, 87, 19],
    "T": [90, 52, 59, 72, 157],
    "U": [23, 83, 89, 48, 2],
    "V": [23, 9, 23, 28, 0],
    "W": [42, 25, 12, 16, 10],
    "X": [0, 10, 8, 2, 4],
    "Y": [5, 7, 19, 2, 175],
    "Z": [2, 1, 4, 12, 2],
}

# Importing Wordle valid words

file_name = "Z:/My Documents/ValidWords.txt"
with open(file_name, "r") as file:
       word_list = file.read().splitlines()  # Reads and processes within the
with block
print(f"Number of words in the file: {len(word_list)}")

#function to calculate word score

def calculate_word_score(word):
    return sum(letter_scores[letter.upper()][i] for i, letter in
enumerate(word))

# Calculate scores for all valid words as a new dictionary

word_scores = {word: calculate_word_score(word) for word in word_list}
```

```
# Find the highest scoring word

highest_scoring_word = max(word_scores, key=word_scores.get)
highest_score = word_scores[highest_scoring_word]
print("highest score is", highest_scoring_word, highest_score)

#Find the lowest scoring word

lowest_scoring_word = min(word_scores, key=word_scores.get)
lowest_score = word_scores[lowest_scoring_word]
print("lowest score is", lowest_scoring_word, lowest_score)

#ADDITIONAL CODE TO FIND TOP 5 AND BOTTOM 5 WORDS
# Sort the words by score in descending order for finding highest scoring words

sorted_words_descending = sorted(word_scores.items(), key=lambda x: x[1],
reverse=True)

# Sort the words by score in ascending order for finding lowest scoring words

sorted_words_ascending = sorted(word_scores.items(), key=lambda x: x[1])

second_highest_scoring_word = sorted_words_descending[1][0]
second_highest_score = sorted_words_descending[1][1]
print("second highest is", second_highest_scoring_word, second_highest_score)

third_highest_scoring_word = sorted_words_descending[2][0]
third_highest_score = sorted_words_descending[2][1]
print("third highest is", third_highest_scoring_word, third_highest_score)

fourth_highest_scoring_word = sorted_words_descending[3][0]
fourth_highest_score = sorted_words_descending[3][1]
print("fourth highest is", fourth_highest_scoring_word, fourth_highest_score)

fifth_highest_scoring_word = sorted_words_descending[4][0]
fifth_highest_score = sorted_words_descending[4][1]
print("fifth highest is", fifth_highest_scoring_word, fifth_highest_score)

second_lowest_scoring_word = sorted_words_ascending[1][0]
second_lowest_score = sorted_words_ascending[1][1]
print("second lowest is", second_lowest_scoring_word, second_lowest_score)

third_lowest_scoring_word = sorted_words_ascending[2][0]
third_lowest_score = sorted_words_ascending[2][1]
print("third lowest is", third_lowest_scoring_word, third_lowest_score)

fourth_lowest_scoring_word = sorted_words_ascending[3][0]
fourth_lowest_score = sorted_words_ascending[3][1]
print("fourth lowest is", fourth_lowest_scoring_word, fourth_lowest_score)

fifth_lowest_scoring_word = sorted_words_ascending[4][0]
fifth_lowest_score = sorted_words_ascending[4][1]
print("fifth lowest is", fifth_lowest_scoring_word, fifth_lowest_score)
```

```
if word_scores:  # Ensure the dictionary is not empty
    mean_score = sum(word_scores.values()) / len(word_scores)
    print(f"Mean word score: {mean_score:.2f}")  # Rounded to 2 decimal places
else:
    print("No words found to calculate the mean score.")


#ADDITIONAL CODE TO FIND HIGHEST SCORING WORD USING ONLY THE TOP LETTERS
(EAROTLIS)
#Allowing words only containing the top identified letters. This was done with
the top 5 to top 8 letters.

allowed_letters = set("earot")
filtered_words = [
    word for word in word_list
    if set(word).issubset(allowed_letters)]

# Calculate scores for all filtered words

word_scores = {word: calculate_word_score(word) for word in filtered_words}

# Sort the words by score in descending order and get the highest scoring word

sorted_words = sorted(word_scores.items(), key=lambda x: x[1], reverse=True)
filtered_highest_scoring_word, filtered_highest_score = sorted_words[0]
print("filtering with the top 5 letters, the highest scoring word is",
filtered_highest_scoring_word, filtered_highest_score)Appendix C: Responses
given from Python Code

#Python Response
Number of words in the file: 14855
highest score is soare 908
lowest score is imshi 98
second highest is seare 886
third highest is saree 883
fourth highest is saine 878
fifth highest is sease 875
second lowest is emmys 112
third lowest is umphs 113
fourth lowest is xysti 120
fifth lowest is enzym 128
Mean word score: 445.41
filtering with the top 5 letters, the highest scoring word is raree 738
```