

How it plays Wordle

1. Get the wordle library from online using web page data
 - a. There are about 13000 acceptable guesses
 - b. Of those 13000, there are about 2300 words that can be solutions
2. The bot chooses a word to guess from the list of 13000 acceptable words.
3. It submits a guess and gets the colors for each letter. Then it removes words from the possible solutions that cannot be the answer.

In this case:



It removes words that don't start with M. It removes words with A, F and I.

In this case:



It removes words without T or O. It removes words with T in position 1 and O in position 3. It removes words with R or L.

4. After each guess, the list shrinks. Eventually you have one word left that has to be the answer.

How it chooses good guesses

You want to shrink the list as much as possible with each guess. The algorithm calculates how much each possible guess would reduce the solution list on average. Here is an illustration of the concept:

Say the bot narrowed the list of possible solutions to this:

lakes
rakes
takes

Guessing **hippo** would be useless

the letters will all be gray _____ (eliminates 0 words)

Average reduction = 0

Guessing **loopy** would be good because it has an **l**

if the **l** becomes green, the answer is **lakes** — (eliminates 2 words)

if it becomes gray, the answer is **rakes** or **takes** — (eliminates 1 word)

Average reduction = 1.5

Guessing **blurt** would be best because it has **l, r, t**

if the **l** becomes yellow, the answer is **lakes** — (eliminates 2 words)

if the **r** becomes yellow, the answer is **rakes** — (eliminates 2 words)

if the **t** becomes yellow, the answer is **takes** — (eliminates 2 words)

Average reduction = 2

This is the process the algorithm follows to choose guesses:

1. To calculate the above reduction values, the algorithm first calculates the reduction value of each letter in each position. Here is how:

It checks every word in the solution list and counts every letter in every position.

Say this is the result for the letter **a** (assume no word has 2 a's):

2 words with **a** in position 1
1 words with **a** in position 2
1 words with **a** in position 3
1 words with **a** in position 4
1 words with **a** in position 5
5 words with no **a**

11 words in total

Then, it calculates the reduction from each letter. Say the guess has **a** in position 1:

case 1: the **a** becomes green

words eliminated: **9** (words without a in position 1)

Probability: **2/11** (# words with a in position 1 / total)

Reduction of case 1: **9 * 2/11**

case 2: the **a** becomes yellow

words eliminated: **7** (words with a in position 1 or with no a)

Probability: **4/11** (# of words with a NOT in position 1 / total)

Reduction of case 2: **7 * 4/11**

case 3: the **a** becomes gray

words eliminated: **6** (words with any a)

Probability: **5/11** (# of words with no a / total)

Reduction of case 3: **6 * 5/11**

*So the average reduction of an **a** in position 1 within a guess is*

(9 * 2/11) + (7 * 4/11) + (6 * 5/11)

The algorithm calculates this score for **a** in position 1, 2, 3, 4, 5, then for **b** in position 1, 2, 3, 4, 5, and then for every letter in every position.

- 2. For every possible guess (of the 13000), it sums the scores of its letters in each position.**

Say the algorithm is checking the possible guess 'apple'

It sums the reduction values of

a in position 1,
p in position 2,
p in position 3,
l in position 4,
and
e in position 5

That score is assigned to the word 'apple'.

It does this for every word.

- 3. All words now have a score for average reduction of the solution list. The bot chooses the word that has the best score.**

It sorts the list of words in the Wordle library by score then chooses the first one.

This solver assumes that guesses with all unique letters are best. If the best guess according to the algorithm has repeated letters, the algorithm chooses the best guess that has all unique letters.