A Deep Learning Approach for Deeply Inaccurate Wordle Solving

Ahana Deb † *& Sayan Goswami* †

Jadavpur University

ahanadeb01@gmail.com, email@sayan.page

Abstract

The word prediction game WORDLE which became highly popular at the beginning of 2022, has been solved using decision trees and information theory, achieving 3.42 average guesses per win benchmark score. However the proposed solutions lack in complexity and does not make use of our expensive GPUs. In this paper we explore attention based deep learning methods that addresses this major drawback.

1. Introduction

WORDLE[1] is a word game played by mostly students and academics, or broadly people with no social life, who need one little accomplishment to get through the day. It involves repeated guessing of a five-letter word, and usually ends with the player in shock that this many five-letter words existed in the first place. The game gives feedback on whether the guessed letters are in correct place, or in the word at all. Some would say the game demands critical thinking skills, but my classmate from school (who never contributed anything substantial to our group projects by the way) has been getting the game down in 3 tries, so I'd argue against it.

2. Previous Work

The game, which was already being coopted by loners, invited people further removed from society to attempt to solve it with information theory. The current state of the art using decision trees[2], achieves a score of 3.42 average guess per win, with other works[3] not far behind, scoring 3.43 on the same metric. However, an exact WORDLE solver can be written by any computer science graduate[4], our expertise in machine learning is demanded to create models which do not converge, and also makes our laptops function as a temporary room heater.

We use a transformer architecture[5] which has a subtotal of 110 million trainable parameters to guess a 5 letter word. Ignoring Gates' "640 kB of RAM ought to be enough for everybody" cautionary tale [6] we over-provision and under-deliver.

† denotes unequal contribution

3. Implementation

The implementation is left as an exercise to the reader. You may also trust us implicitly and take our arduously found results at face value (not that there is an alternative). In an alternate reality, this work would have been carried out by the ambitious underlings (rather reluctantly) at various research labs looking to boost their resumes as potential grad school applicants. However, as we and our readers are wiser, we have decided to exploit these otherwise wasted efforts by communicating telepathically across space and time. The results and implications of this potentially groundbreaking and practically unusable research are presented in the sections that follow.

4. Evaluation

Our initial approach involved utilizing the information we got from the wrong guesses, about the letters guessed correctly and their respective positions, to train our model. But at this point we realized, this would firstly make the task much easier for our model to learn, compromising on our complexity objective, and secondly would involve us doing some actual work. Solely based on the first reason, we decided to leave that approach untouched and evaluated the model as it is. A question that can be asked at this point is "why bother at all?" but we were already too deep into this to go down a second rabbit hole.

Even though Hoeffding's inequality theorizes the upper bound on the difference between the empirical risk and the generalisation error on the domain set as a function of the number of data points observed, we find that, in theory, our "learning" algorithm is a special exception to it, and learns practically nothing. We compare our model to pre-established and newly conjured baselines, and plot the trend for average number of moves to solve the puzzle as compared to the model complexity in figure 1.

5. Conclusions

As we can see our proposed solution outperforms (barely, if at all) a random word generator, and our one friend who does not speak English, and was quite reluctant to play this game in the first place.

Building on our main objective to create a model

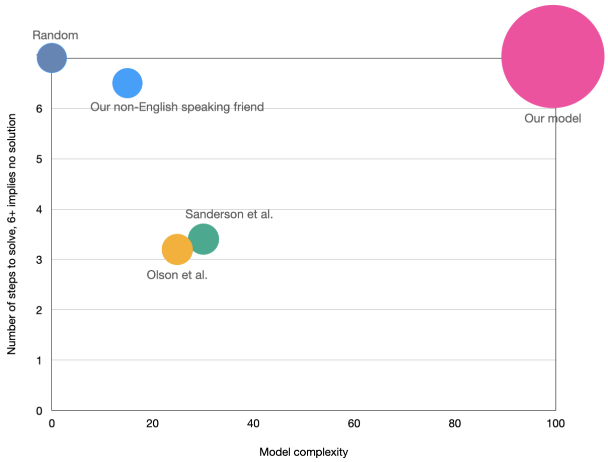

Figure 1: *Performances of our model as compared to baselines.*

as complex as possible, we believe any simple task can be made as convoluted as desired if you're not bound by the unforgiving chains of evaluation metrics. Sky being the limit for the number of parameters that we could've trained for this task, unfortunately, the authors of this paper could only sit in front of a laptop screen for 18 hours a day (the other 6 being reserved for a smaller screen, and sleep being designed for the weak).

6. Acknowledgements

We would like to thank our non-English-speaking friend who was dragged into this through no fault of their own, and express our gratitude for not cutting us off, and also to the creators of affordable GPUs (such as Nvdia's 3090 Ti & Titan V) without which this abomination would have never seen the light of day, and probably rightly so.

7. References

[1] *The New York Times Wordle Game*, 2022. [Online]. Available: https://www.nytimes.com/games/wordle/index.html

[2] J. Olson, *Optimal Wordle Solutions.* https://jonathanolson.net/experiments/optimal-wordle-solutions, 2022.

[3] G. Sanderson, *"Solving Wordle with Information Theory"*, 2022. [Online]. Available: https://www.youtube.com/watch?v=v68zYyaEmEA

[4] *This brilliant tweet by George Toderici which we took very literally*, 2022. [Online]. Available: https://bit.ly/3v6zud5

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. [Online]. Available: https://aclanthology.org/N19-1423

[6] "Bill gates denies making 1981 comment about limits of ram needs, despite popular legend," 1981. [Online]. Available: https://bit.ly/3wXxOVM