

# Trexquant Interview Project (The Hangman Game)

## SUMMARY

---

### Hangman game solver:

1. Guessing Strategy: The guessing strategy is adaptive and changes based on the word length and the number of unknown letters:
  - a. For longer words ( $\text{length} > 11$ ):
    - If there are many unknown letters ( $> \text{len}(\text{clean\_word}) - 4$ ), it uses `find_most_common_letter()`.
    - Otherwise, it uses `final_strategy()`.
  - b. For shorter words ( $\text{length} \leq 7$ ):
    - If there are many unknown letters ( $> \text{len}(\text{clean\_word}) - 2$ ), it uses `find_most_common_letter()`.
    - For 3 or more unknown letters, it uses `final_strategy()`.
    - For 2 or fewer unknown letters, it uses the BERT model for prediction.
  - c. For medium-length words:
    - If there are many unknown letters ( $> \text{len}(\text{clean\_word}) - 3$ ), it uses `find_most_common_letter()`.
    - Otherwise, it uses `final_strategy()`.
2. Detailed Strategies:
  - a. `find_most_common_letter()`:
    - Counts the occurrence of each letter in the current dictionary.
    - Returns the most common unguessed letter.
  - b. `final_strategy()`:
    - Uses a combination of n-gram models (1-gram to 5-gram) to calculate probabilities for each letter.
    - The probabilities are weighted and combined across different n-gram levels.
    - Returns the letter with the highest probability.
  - c. BERT model strategy:
    - Used for short words with few unknown letters.

- Masks one unknown position and uses the BERT model to predict the most likely letter for that position.

d. `guess_last_letter_ngram()`:

- A specialized strategy for guessing the last few letters, using a combination of n-gram models.

This strategy combines statistical methods (n-grams), machine learning (BERT), and adaptive decision-making based on the game state. It starts with broader, frequency-based guesses and becomes more focused as the word becomes clearer, ultimately relying on advanced prediction methods for the final letters.

## Potential Improvements:

1. Advanced Transformer Models:
  - Upgrade to more sophisticated models (e.g., RoBERTa, GPT-3) for better letter prediction.
2. Word-level Prediction:
  - Generate likely full words for final guesses instead of individual letters.
3. Adaptive Learning:
  - Implement a system that learns and adjusts strategies based on game outcomes.

These enhancements could potentially boost accuracy to around 90%, especially effective when 80% of letters are already correctly guessed, by improving performance in the crucial final stages of word completion.