

## Jogador Automático de Forca - Explicação do Projeto

### Introdução ao Projeto

Este projeto implementa um jogador automático para o jogo de forca. O objetivo é desenvolver uma estratégia baseada em teoria da informação, especificamente no conceito de entropia, para maximizar o número de vitórias com um limite de 5 vidas. A ideia é usar a entropia para selecionar letras que forneçam a maior quantidade de informação sobre a palavra oculta, enquanto um processo de filtragem contínua reduz o conjunto de palavras possíveis.

### Estrutura e Configuração do Jogo

O jogo foi implementado na classe `JogoDeForca`, que controla as regras e mantém as vidas e tentativas do jogador. O vocabulário utilizado foi obtido de uma lista de palavras em português, garantindo um conjunto consistente e diversificado. As funções principais `tentar_letra`` e `tentar_palavra`` são responsáveis por validar as tentativas do jogador e determinar vitórias ou derrotas.

### Vocabulário e Pré-processamento

O vocabulário foi carregado a partir de uma fonte online e filtrado para remover palavras inválidas, mantendo apenas palavras em minúsculo e sem acentos. Isso assegura que o jogador opere com um conjunto de palavras limpo e fundamental para a precisão das tentativas.

### Cálculo de Entropia para Seleção de Letras

O conceito de entropia é essencial neste projeto. A entropia mede a incerteza associada a cada letra, ajudando a selecionar letras que maximizem a informação revelada. Na função `calcular_entropia``, a probabilidade de uma letra aparecer é calculada com base na frequência nas palavras restantes, e as letras com alta entropia são preferidas. Essa estratégia reduz a incerteza sobre a palavra e otimiza as tentativas.

### Filtragem Dinâmica de Palavras

A função `filtrar_palavras` refina o vocabulário a cada tentativa, mantendo apenas palavras que correspondam às letras e posições já descobertas. Isso permite que o jogador foque em um conjunto menor de palavras em cada jogada, tornando a próxima tentativa ainda mais precisa. Esse processo é essencial para o cálculo de entropia, pois reduz a incerteza e maximiza a eficiência.

## Execução e Avaliação dos Jogos

No notebook `demo.ipynb`, o jogador automático foi testado em um conjunto de 100 jogos. Os resultados mostram uma taxa de vitória baseada na eficácia da estratégia de entropia e filtragem dinâmica. As falhas ocorrem principalmente com palavras de padrão ambíguo ou letras raras, onde a incerteza permanece elevada mesmo com o refinamento.

## Conclusão e Próximos Passos

A entropia e o processo de filtragem mostraram-se eficientes para maximizar o número de vitórias. Como próximos passos, seria interessante explorar técnicas adicionais que possam identificar padrões específicos ou lidar com palavras com padrões difíceis de prever. Isso poderia melhorar ainda mais a taxa de acerto do jogador.

Esse projeto foi uma exploração prática da teoria da informação, demonstrando como a entropia pode ser aplicada em problemas de tomada de decisão com informações limitadas.