Segundo Trabalho de Inteligência Artificial

Filipe Gomes Arante de Souza

Julho 2023

Resumo

Este artigo consiste numa investigação sobre a aplicação de aprendizado por reforço no contexto do jogo do dinossauro do Google, um passatempo amplamente conhecido. Para o agente aprender a sobreviver o máximo possível numa corrida, foi utilizado um classificador de Árvore de Decisão associado a metaheurística de Otimzação por Enxame de Partículas (PSO). O classificador recebe informações do ambiente do jogo para decidir se o personagem pula ou abaixa perante algum obstáculo, enquanto o PSO tenta descobrir os parâmetros ótimos que a Árvore de Decisão deve receber.

Confira uma breve demonstração em vídeo do resultado **clicando aqui.**

Palavras-chave: Aprendizado por reforço; Classificação; Árvore de decisão; Metaheurística; PSO; Dino;

1 Introdução

O aprendizado por reforço é uma técnica de aprendizado de máquina que busca capacitar agentes a tomar decisões autônomas em ambientes complexos e dinâmicos. Uma ótima plataforma para estudar e analisar o comportamento de algoritmos de aprendizado é o jogo do dinossauro do Google. A fim de aplicar os conceitos lecionados em sala de aula, será feita a construção de um classificador associado a uma metaheurística para treinar o dinossauro, objetivando o alcance de um desempenho competitivo. Será apresentada a evolução da pontuação do agente ao longo das iterações do algoritmo, além da comparação estatística com um método baseline para verificar se foi obtido um resultado superior.

2 Descrição do Classificador

O classificador utilizado para determinar as ações do dinossauro foi definido conforme o número de matrícula do aluno. Para final de matrícula 5, foi utilizada a árvore de decisão.

2.1 Árvore de Decisão

Uma árvore de decisão é um algoritmo de aprendizado de máquina supervisionado utilizado para classificação e regressão. Esta árvore estabelece nós de decisão que se relacionam entre si hierarquicamente, que, de modo simplificado, se organizam da seguinte maneira:

- Nós intermediários: Representam os testes a serem feitos com uma entrada para classificá-la.
- Nós folha: Representam os resultados finais de predição para um dado ramo.

No contexto do trabalho, a árvore de decisão foi construída de modo a orientar as ações do dinossauro ao longo de uma corrida. O classificador recebe diversas informações do ambiente do jogo, como por exemplo a distância, altura e tipo dos dois obstáculos mais próximos do personagem.

2.2 Implementação

A estratégia para tornar o dinossauro esperto consiste em determinar o melhor momento para pular um obstáculo e o melhor momento para abaixar quando se está no ar. Portanto, foram definidos dois parâmetros para a árvore de decisão:

- **Alpha** (α): É uma constante de proporcionalidade entre a distância mínima que se deve pular e a velocidade do jogo. Desse modo, ela determina quando o dinossauro pula algum obstáculo. Além disso, quanto mais rápido o jogo, mais longe do obstáculo o dinossauro irá pular, portanto a tendência após o salto é cair no chão a uma distância razoável do próximo obstáculo.
- **Beta** (β): É uma constante de proporcionalidade entre a distância limite que se deve abaixar e a velocidade do jogo. Desse modo, ela determina quando o dinossauro abaixa após um salto. A ideia deste parâmetro é chegar ao solo o mais rápido possível para ter uma reação melhor ao próximo obstáculo.

A partir destas constantes, são determinadas os valores limite **limDistUp** e **limDistDown**, que são calculados da seguinte maneira:

$$limDistUp = \alpha \cdot gameSpeed$$

$$limDistDown = \beta \cdot gameSpeed$$

Com base nessa intuição, a árvore ficou da seguinte forma:

Figure 1: Árvore de decisão que determina as ações do dinossauro.

3 Descrição da Meta Heurística

Assim como o classificador, a metaheurística também foi definida conforme número de matrícula. Para o final de matrícula 5 foi utilizada a técnica de Otimização por Enxame de Partículas (PSO).

3.1 Otimização por Enxame de Partículas (PSO)

O PSO é um algoritmo heurístico inspirado no comportamento social de um bando de pássaros. Seu objetivo é buscar a solução ótima num determinado espaço de busca através da troca de informações entre indivíduos de uma população determinando qual trajetória cada membro dessa população deve tomar no espaço de busca.

Resumidamente, cada partícula k do enxame possui uma posição x_k no espaço e uma velocidade v_k. Elas variam, a cada geração, da seguinte maneira:

$$v_{k+1} = w \cdot v_k + c_1 \cdot r_1 \cdot (p_{best_k} - x_k) + c_2 \cdot r_2 \cdot (g_{best} - x_k)$$

$$x_{k+1} = x_k + v_k$$

Onde w é o coeficiente inercial, c_1 é o coeficiente individual, c_2 é o coeficiente social, r_1 e r_2 são valores aleatórios entre 0 e 1, p_{best_k} é a melhor posição já encontrada da partícula k e g_{best} é a melhor posição já encontrada em todo o enxame.

Os "pássaros" navegam no espaço em busca de bons valores para α e β para instanciar os classificadores de árvore de decisão, e assim tentar maximizar a função objetivo, que neste caso é a pontuação do jogo.

3.2 Implementação

O PSO foi implementado com algumas especificidades para se adotar bem as particularidades do jogo. Os valores de w, c_1 e c_2 foram os seguintes:

$$w = 0.08, \quad c_1 = 0.2, \quad c_2 = 0.05$$

Estes parâmetros foram escolhidos assim pois foi percebido ao longo do desenvolvimento do trabalho que os valores das componentes de posição e velocidade estavam crescendo muito rapidamente com poucas dezenas de gerações, chegando a ordem de grandeza de 10^{30}, o que claramente não tinha significado no contexto do jogo. Portanto, a ideia dessas constantes estarem com o valor razoavelmente baixo é frear esse crescimento brusco que estava ocorrendo.

Além disso, caso alguma componente de x_k ou v_k ultrapassar 50, elas são sorteadas aleatoriamente nos ranges $[0, 30]$ e $[0, 10]$, respectivamente, pois estes intervalos são os que melhor se comportaram para o jogo.

Todas as partículas do enxame foram inicializadas com uma posição de valor aleatório entre 0 e 1 nas componentes α e β. Já suas velocidades foram inicializadas em 0.

A quantidade de iterações determinadas para treinar o algoritmo foi 10.000, com 50 dinossauros em sua população. Em cada geração serão feitas 5 corridas, a fim de se obter um resultado consistente para cada partícula numa determinada iteração.

4 Resultados

4.1 Iteração vs Melhor Pontuação do Enxame / Média do Enxame

As melhores pontuações ficaram no range $[2500, 3500]$. Já a pontuação média do enxame no range $[500, 1000]$.

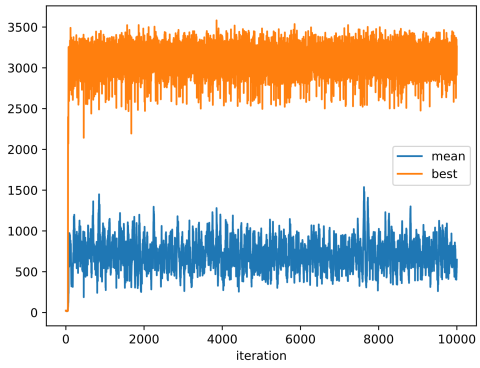

Figure 2: Melhor resultado e média da pontuação em função da iteração do PSO.

4.2 Comparação com baseline fornecido pelo professor

4.2.1 Resultados das 30 corridas

Os valores ótimos encontrados ($\alpha = 23.891$ e $\beta = 6.298$) pelo PSO tiveram desempenho bem superior ao baseline. Note que a média da pontuação foi, em média, cerca de 3 vezes maior.

Run	PSO	Baseline	Run	PSO	Baseline	Run	PSO	Baseline
01	3741.0	1214.0	**11**	3315.75	728.5	**21**	3515.5	751.0
02	3669.0	759.5	**12**	3086.25	419.25	**22**	3560.75	1418.75
03	3479.0	1164.25	**13**	3439.0	1389.5	**23**	3468.75	1276.5
04	3143.0	977.25	**14**	3166.5	730.0	**24**	3220.0	1645.75
05	3442.0	1201.0	**15**	2783.25	1306.25	**25**	3588.75	860.0
06	3383.5	930.0	**16**	3042.5	675.5	**26**	2760.75	745.5
07	3606.25	1427.75	**17**	3116.0	1359.5	**27**	3311.0	1426.25
08	3123.75	799.5	**18**	2952.25	1000.25	**28**	2781.75	783.5
09	3529.25	1006.25	**19**	3307.0	1284.5	**29**	3210.5	1149.75
10	3369.5	783.5	**20**	3241.75	1350.0	**30**	3362.0	1482.25

Table 1: Resultados obtidos pelo PSO e pelo baseline disponbilizado.

	PSO	Baseline
Mean	3290.54	1068.18
STD	256.43	304.04

Table 2: Média e desvio padrão do PSO e do baseline fornecido.

4.2.2 Boxplot

Confira visualmente os resultados da tabela mostrada acima. Veja como a estratégia aplicada na árvore de decisão proporcionou ótimas performances do dinossauro.

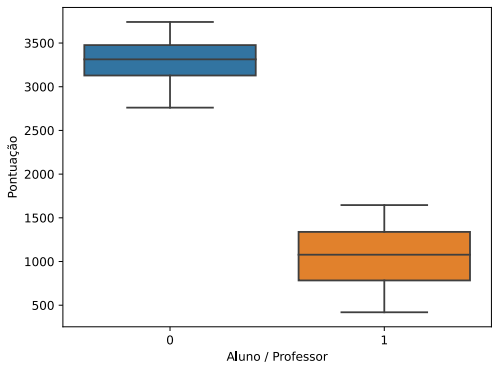

Figure 3: Boxplot da distribuição da pontuação do dinossauro ajustado pelo PSO e o do professor.

4.2.3 Testes de Hipótese

Foram obtidos os seguintes p-values para os testes de hipótese:

- Teste T Pareado: $4.353 \cdot 10^{-25}$
- Teste de Wilcoxon: $1.863 \cdot 10^{-9}$

Como os valores obtidos foram muito baixos, para os dois testes houveram diferenças estatísticas significativas.

5 Conclusões

5.1 Análise geral dos resultados

O princípio de encontrar o melhor momento para o dinossauro pular e abaixar funcionou muito bem junto aos parâmetros α e β, que tornaram o dinossauro mais adaptável a velocidade do jogo. Devido a este método ter sido modelado com apenas 2 argumentos, o espaço de busca foi bastante reduzido e permitiu o PSO encontrar soluções boas rapidamente. Note que no gráfico de linhas o melhor resultado de uma geração teve um salto próximo da melhor solução global com poucas dezenas de iterações. Após isso, os resultados ficaram oscilando entre 2500 e 3500 pontos até o final das 10000 gerações. Muito provavelmente esta oscilação ocorreu devido a restrição que foi imposta na atualização da posição e velocidade das partículas.

Esse fato pode ser indício de uma estagnação num máximo local, entretanto, durante o desenvolvimento do projeto, foram explorados diversos outros espaços de busca e todos tiveram resultados muito inferiores. Portanto, há uma boa chance deste máximo local também ser global.

Além disso, o PSO se mostrou muito sensível aos valores inicializados em sua primeira iteração. Foi necessário fazer vários experimentos para descobrir qual o melhor intervalo para definir os valores de α e β, que no final foi definido entre 0 e 1, para ambas as variáveis.

No que tange a comparação com o baseline, todas as métricas apresentadas apontaram para um resultado superior, o que é comprovado pelos testes de hipótese. Eles apresentaram diferença estatística significativa num nível acima de 99%.

5.2 Contribuições do Trabalho

O uso da Otimização por Enxame de Partículas permitiu explorar uma abordagem diferente para treinar a IA, já que o mais comum para este trabalho é a utilização do Algoritmo Genético com Redes Neurais. Portanto, é possível realizar com este artigo uma análise comparativa com outras técnicas de Aprendizado por Reforço, destacando vantagens e desvantagens da abordagem proposta em relação a outros métodos.

5.3 Melhorias e trabalhos futuros

O próximo passo deste trabalho, a fim de tornar o dinossauro mais inteligente, é pensar numa estratégia para a árvore de decisão que faça o personagem reconhecer quando pular dois obstáculos com um único salto. Isso o tornaria bem mais eficaz em velocidades mais altas, e, com certeza, quebraria a fronteira dos 3500 pontos.

Outra alternativa pode ser aplicar outra metaheurística no lugar do PSO. Técnicas como o Algoritmo Genético e o Simulated Annealing são boas opções. Além disso, outros classificadores podem ser testados, tais como Redes Neurais e KNN.

Referências Bibliográficas

Os materiais utilizados para o desenvolvimento do artigo foram os slides e notebooks do professor Flávio Varejão mostrados em sala de aula.