



IME - Instituto de
Matemática e Estatística

Criação de um jogo com dificuldade dinâmica e aprendizagem por reforço

Aluno: João Gabriel Basi
Orientador: Prof. Dr. Roberto Hirata Jr.

Departamento de Ciência da Computação, Instituto de Matemática e Estatística,
Universidade de São Paulo



Introdução

Em jogos *player versus environment* (PvE) é necessário que os inimigos considerem todas as situações de jogo, caso contrário, em métodos tradicionais como árvores de comportamento, a experiência do jogador pode ser prejudicada.

Outro problema que assombra todos os tipos de jogos é o balanceamento. Se o jogo não for balanceado de forma que a curva de aprendizado do jogador seja suave, o jogador se sentirá frustrado ou entediado durante o jogo.

Andrade et al., 2006, e Demasi e Cruz, 2003, mostram através de experimentos com jogadores que algoritmos de aprendizagem de máquina podem resolver os dois problemas, contudo é muito difícil encontrar jogos no mercado que de fato os utilizam.

O jogo

Este trabalho consistiu no desenvolvimento de um jogo PvE, utilizando o motor Godot em conjunto com PyTorch, com o intuito de explorar ao máximo os algoritmos de adaptação.

O jogo se passa em um mundo de fantasia medieval, no qual algumas pessoas chamadas de aventureiras exploram lugares desconhecidos e são famosas por derrotarem monstros terríveis e conseguirem riquezas inestimáveis. O objetivo do jogador é ajudar o protagonista, que não tem dinheiro nem habilidades, a se tornar um aventureiro famoso.



Figura 1: Protagonista em sua cidade natal.

Agentes inimigos

Na criação dos agentes inimigos foi utilizado o algoritmo *Q-Learning* (Watkins e Dayan, 1992), em conjunto com redes neurais para aproximar a função Q . O algoritmo permite que a AI aprenda uma política útil em dois minutos de treino e não tenha os problemas da árvore de comportamentos.

$$R(s, a, s') = \frac{v_s^{(E)} - v_{s'}^{(E)}}{v_s^{(E)}} - \frac{v_s^{(A)} - v_{s'}^{(A)}}{v_s^{(A)}}$$

A recompensa recebida pelos agentes leva em conta a vida $v^{(E)}$ do inimigo e a vida $v^{(A)}$ do agente em dois estados consecutivos do jogo.



Figura 2: Protagonista enfrentando um goblin em uma aventura.

Adaptação de dificuldade

O jogo utiliza o algoritmo *Computerized adaptive practice* (CAP) (Van der Maas e Nyamsuren, 2017) para estimar a habilidade do jogador e a dificuldade dos conjuntos de inimigos através de métricas das batalhas.

$$\delta = \theta + \ln \left(\frac{1-p}{p} \right)$$

O algoritmo então usa a habilidade do jogador θ e uma probabilidade de vitória p para escolher os grupos de inimigos com dificuldade próxima de δ de forma que o jogador vença em média 75% das vezes (Eggen e Verschoor, 2006).



Figura 3: Interface da loja de poções.

Conclusão

Por ser do estilo PvE o jogo conseguiu explorar bem a criação de agentes por técnicas de aprendizagem de máquina, e a adaptação de dificuldade ajudou a construir a progressão do jogo.

Os agentes utilizados na versão final do jogo foram pré-treinados por dois minutos em fase de desenvolvimento, o que foi suficiente para que eles aprendessem a combater o jogador. Os algoritmos de aprendizagem diminuíram muito o tempo de desenvolvimento se comparado às árvores de comportamento.

O algoritmo CAP fez com que a progressão do jogo se tornasse mais linear, o que contribui para que a curva de aprendizado do jogador seja suave. Ele também ajudou a ditar a progressão do jogo, já que a habilidade do jogador pode ser utilizada como requisito de progressão.