

Arthur de Senna Rocha

Desenvolvimento de uma Inteligência Artificial para aprender a jogar jogos em Allegro

Brasil

15 de outubro de 2019

Arthur de Senna Rocha

Desenvolvimento de uma Inteligência Artificial para aprender a jogar jogos em Allegro

Trabalho de Conclusão de Curso I apresentado como requisito parcial à obtenção de título de bacharel em Engenharia de Sistemas pela Escola de Engenharia da Universidade Federal de Minas Gerais.

Universidade Federal de Minas Gerais – UFMG

Escola de Engenharia

Engenharia de Sistemas

Orientador: Pedro Olmo Stancioli Vaz De Melo

Brasil

15 de outubro de 2019

Arthur de Senna Rocha

Desenvolvimento de uma Inteligência Artificial para aprender a jogar jogos em Allegro

Trabalho de Conclusão de Curso I apresentado como requisito parcial à obtenção de título de bacharel em Engenharia de Sistemas pela Escola de Engenharia da Universidade Federal de Minas Gerais.

Trabalho aprovado. Brasil, xx de xxxx de xxxx:

Pedro Olmo Stancioli Vaz De Melo
Orientador

Professor
Convidado 1

Professor
Convidado 2

Brasil
15 de outubro de 2019

*“When it is not in our power to determine what is true,
we ought to act according to what is most probable.”
(René Descartes)*

Resumo

O uso de inteligência artificial (IA) e de algoritmos de *machine learning* possibilita que máquinas aprendam com experiências, se ajustem às novas entradas de dados e performem tarefas como seres humanos. Com essas tecnologias, os computadores podem ser treinados para cumprir tarefas específicas ao processar grandes quantidades de dados e reconhecer padrões nesses dados. O presente trabalho se propõe a desenvolver uma IA capaz de aprender a jogar diferentes jogos, desde que se tenha acesso ao código fonte e feito em Allegro. Para isso, será implementado um algoritmo de *Deep Reinforcement Learning*, abordagem que consiste em fornecer ao sistema parâmetros relacionados ao seu estado e uma recompensa positiva ou negativa com base em suas ações. Nenhuma regra sobre o jogo é dada e, inicialmente, a IA não tem informações sobre o que precisa fazer. A única informação passada para a IA são os comandos básicos do jogo. O objetivo do sistema é descobrir e elaborar uma estratégia para maximizar a pontuação - ou a recompensa. Diferente de muitas IAs que focam na solução de um único problema, a proposta deste projeto é elaborar uma IA que seja genérica e capaz solucionar e elaborar estratégias para uma variedade de situações diferentes.

Palavras-chave: deep learning, allegro, inteligência artificial, jogos digitais, machine learning.

Abstract

The use of Artificial intelligence (AI) and machine learning algorithms enables computers to learn from experience, adjust to new data inputs, and perform tasks as human beings. With these technologies, computers can be trained to perform specific tasks by processing large amounts of data and recognizing patterns in that data. The present work aims to develop an AI capable of learning how to play different games, provided that it has access to the source code and the game is made in Allegro. For this, a Deep Reinforcement Learning algorithm will be implemented, which provides the system with parameters related to its state and a positive or negative reward based on its actions. No rules about the game are given and initially AI has no information on what it needs to do. The only information passed to AI is the basic commands of the game. The goal of the system is to discover and devise a strategy to maximize its score - or the reward. Unlike many AIs that focus on solving a single problem, the purpose of this project is to design a generic AI that can solve and develop a strategy for a variety of different situations.

Keywords: deep learning, allegro, artificial intelligence, video games, machine learning.

Lista de ilustrações

Figura 1 – Ilustração de um modelo de aprendizado profundo	24
Figura 2 – Diagrama de aprendizagem por reforço	26

Lista de abreviaturas e siglas

ANNs	<i>Artificial Neural Networks</i>
DL	<i>Deep Learning</i>
IA	Inteligência Artificial
ML	<i>Machine Learning</i>
NPC	<i>Non-player Character</i>
PCG	<i>Procedural Content Generation</i>
PNL	Processamento de Linguagem Natural
RL	<i>Reinforcement Learning</i>

Sumário

	Sumário	15
1	INTRODUÇÃO	17
1.1	Motivação	17
1.2	Objetivos	18
1.3	Descrição do problema	19
1.4	Revisão da literatura	21
1.5	Organização do trabalho	21
2	ABORDAGEM PROPOSTA	23
2.1	<i>Deep Learning</i>	23
2.2	<i>Reinforcement Learning</i>	25
2.3	Aplicação de DRL no treinamento de uma IA para aprender a jogar jogos em <i>Allegro</i>	27
	REFERÊNCIAS	29

1 Introdução

A inteligência artificial (IA) vem ganhando manchetes no mundo todo, sendo anunciada tanto como uma salvação econômica quanto como precursora de desintegração social. Quando computadores programáveis foram concebidos pela primeira vez, as pessoas se perguntavam se essas máquinas poderiam se tornar inteligentes, mais de cem anos antes de uma ser construída (MENABREA et al., 1843). Hoje, a inteligência artificial é um campo com inúmeras aplicações práticas e tópicos de pesquisa ativos. Buscamos softwares inteligentes para automatizar o trabalho de rotina, entender a fala ou as imagens, fazer diagnósticos em medicina e apoiar a pesquisa científica (GOODFELLOW; BENGIO; COURVILLE, 2016).

A IA adiciona inteligência a produtos existentes. Na maioria dos casos, a inteligência artificial não é vendida como uma aplicação individual. Pelo contrário, produtos já existentes são aprimorados com funcionalidades de IA, de maneira parecida como a Siri foi adicionada aos produtos da *Apple*. Automação, plataformas de conversa, robôs e aparelhos inteligentes podem ser combinados com grandes quantidades de dados para aprimorar diversas tecnologias para casa e escritório, de inteligência em segurança à análise de investimentos.

A maioria dos exemplos de IA sobre os quais se ouve falar hoje – de computadores mestres em xadrez a carros autônomos – dependem de *deep learning* e processamento de linguagem natural (PNL) (RODRIGUES, 2017). Treinar um agente para superar os jogadores humanos e otimizar sua performance pode nos ensinar como otimizar diferentes processos em uma grande variedade de situações. Foi o que o *DeepMind* do Google fez com seu popular *AlphaGo* e seu sucessor *AlphaZero*, vencendo os campeões mundiais em Go, xadrez e shogi, e obtendo resultados de performance nunca antes vistos.

1.1 Motivação

Técnicas de aprendizado de máquina e algoritmos de *deep learning* têm consistentemente melhorado a capacidade de um computador de fornecer reconhecimento de padrões e previsões cada vez mais precisas. Além disso, sistemas de DL são consistentemente aplicados com sucesso a conjuntos de aplicações cada vez mais amplos.

Ao mesmo tempo em que a escala e a precisão das redes profundas aumentaram, a complexidade das tarefas que podem ser resolvidas também cresceu significativamente. Uma conquista importante de sistemas de DL é a sua extensão ao domínio da aprendizagem por reforço (*reinforcement learning*) (SUTTON; BARTO, 2018). No contexto do aprendizado

por reforço, um agente autônomo deve aprender a executar uma tarefa por tentativa e erro, sem nenhuma orientação do operador humano.

Além do valor para pesquisa em múltiplas áreas da ciência, muitas dessas aplicações de aprendizado de máquina e *deep learning* são altamente lucrativas. O aprendizado de máquina hoje é usado por muitas empresas de tecnologia, incluindo *Google*, *Microsoft*, *Facebook*, *IBM*, *Baidu*, *Apple*, *Adobe*, *Netflix*.

Diante à crescente presença de sistemas que utilizam técnicas de *deep learning* no dia-a-dia, nota-se o grande potencial do investimento em pesquisa, modelagem de novos problemas e estudo de técnicas de aprendizado de máquina. Uma interessante aplicação desses sistemas está na área de jogos digitais. A indústria de videogames tem testemunhado um enorme crescimento, graças, em boa parte, ao incrível aumento no poder da computação em termos de representações visuais. Seja no controle de personagens não-jogadores (NPC), ou para a geração de conteúdo processual (PCG), são inúmeras as potenciais aplicações dessas técnicas em jogos digitais. O potencial dessas ferramentas de obter uma vantagem competitiva no mercado, ou simplesmente fornecer uma melhor experiência para o usuário é, no mínimo, instigante. Nesse contexto, a modelagem de novos problemas, implementação de soluções utilizando técnicas de *deep learning* e investimento na área, torna-se uma relevante contribuição para o estado da arte.

1.2 Objetivos

O presente trabalho tem como objetivo geral propor o desenvolvimento de uma IA capaz de aprender a jogar diferentes jogos, desde que se tenha acesso ao código fonte e feito em Allegro. Para isso, será implementado um algoritmo utilizando *Deep Reinforcement Learning*, abordagem que consiste em fornecer ao sistema parâmetros relacionados ao seu estado e uma recompensa positiva ou negativa com base em suas ações. Nenhuma regra sobre o jogo é dada e, inicialmente, a IA não tem informações sobre o que precisa fazer. A única informação passada para a IA são os comandos básicos do jogo. O objetivo do sistema é descobrir e elaborar uma estratégia para maximizar a pontuação - ou a recompensa.

Os objetivos mais específicos deste trabalho são:

1. Revisão da literatura do problema;
2. Descrição e modelagem do problema;
3. Proposta de critérios adicionais que possibilitem estimar outras características das possíveis soluções do projeto, tais como performance, confiabilidade, entre outras;
4. Proposta de um algoritmo de *deep learning* para a solução do problema;

5. Análise dos resultados obtidos em comparação com diferentes soluções implementadas por outras entidades e utilizadas na prática por empresas atuando no mercado.

Perante o exposto, a implementação de algoritmos que utilizam o aprendizado de máquina de forma a serem aplicados em diferentes cenários, apresenta um potencial de propor novas estratégias e otimizar sistemas já existentes, melhorar a qualidade do produto final e a experiência do usuário, além de proporcionar uma vantagem competitiva no mercado.

1.3 Descrição do problema

O campo da inteligência artificial é capaz de solucionar, com certa facilidade, problemas que são intelectualmente muito difíceis para os seres humanos, mas relativamente diretos para os computadores - problemas que podem ser descritos por uma lista de regras formais e matemáticas. Tarefas abstratas e formais que estão entre os empreendimentos mentais mais difíceis para um ser humano estão entre os mais fáceis para um computador.

Ironicamente, o grande desafio à inteligência artificial provou estar em resolver tarefas fáceis de executar para um ser humano. Problemas que parecem automáticos, que resolvemos intuitivamente, como reconhecer palavras faladas ou rostos em imagens. Os computadores há muito conseguem derrotar até o melhor jogador de xadrez humano (HSU, 2002), mas apenas recentemente começaram a alcançar algumas das habilidades dos seres humanos comuns, como reconhecer objetos ou fala.

A vida cotidiana de uma pessoa requer uma imensa quantidade de conhecimento sobre o mundo. A grande quantidade de informação desses cenários torna inviável a codificação de todas as regras do sistema e, por isso, o computador tem uma grande dificuldade para solucionar esses problemas. Além disso, grande parte desse conhecimento é subjetivo e intuitivo e, portanto, difícil de articular de maneira formal. Os computadores precisam capturar esse mesmo conhecimento para se comportarem de maneira inteligente. Um dos principais desafios da inteligência artificial é como obter esse conhecimento informal em um computador.

As dificuldades enfrentadas por sistemas que dependem de conhecimento codificado sugerem que os sistemas de IA necessitam da capacidade de adquirir seu próprio conhecimento, extraindo padrões de dados brutos. Esse recurso é conhecido como aprendizado de máquina ou *machine learning* (ML). A introdução do aprendizado de máquina permitiu que os computadores resolvessem problemas que envolvem o conhecimento sobre o mundo real e tomassem decisões mais subjetivas.

No contexto de jogos digitais, treinar um agente para superar os jogadores humanos e otimizar sua pontuação pode nos ensinar como otimizar processos diferentes em uma

variedade de subcampos diferentes e intrigantes (COMI, 2018). Uma abordagem proposta na literatura, obtendo ótimos resultados, e que tem como objetivo treinar um computador pra aprender e desenvolver estratégias para jogar diferentes jogos, é o *deep reinforcement learning*.

O problema proposto nesse trabalho é o de implementar uma IA que, utilizando algoritmos de *deep reinforcement learning*, seja capaz de aprender e desenvolver estratégias para jogar diferentes jogos digitais. Os requisitos do sistema podem ser resumidos pelos seguintes critérios:

1. O sistema receberá, inicialmente, somente os comandos básicos do jogo. Nenhuma regra sobre o jogo é dada e, inicialmente, o agente não tem nenhuma informação sobre o que precisa fazer;
2. O agente deve ser capaz de elaborar uma estratégia para maximizar sua pontuação e que alcance resultados consideravelmente superiores aos de uma abordagem aleatória e próximos aos de um agente humano;
3. O sistema deverá ser capaz de lidar com cenários aleatórios, onde os obstáculos mudam a cada partida, e não aleatórios, onde os obstáculos são “fixos” e a dificuldade varia de acordo com o progresso no jogo;
4. O sistema deve ser generalizado para que possa ser aplicado à diferentes cenários e treinado para jogar diferentes jogos digitais.

De modo a garantir a factibilidade da implementação do sistema, algumas restrições devem ser acatadas. Por exemplo, além de haver a necessidade de se conhecer os comandos básicos do jogo, o sistema precisa ser capaz de obter informações atualizadas sobre o estado do jogo em que se encontra. No caso deste trabalho, foram definidas as seguintes restrições:

1. O sistema deve ter acesso ao código fonte do jogo no qual será aplicado;
2. O jogo deverá ter sido implementado em *Allegro*¹;
3. O jogo deve ser 2D para garantir a viabilidade do sistema;

¹ O acesso ao código fonte nos permite ter conhecimento dos comandos básicos do jogo, enquanto a biblioteca *Allegro* fornece rotinas de baixo nível comumente necessárias na programação de jogos (HARGREAVES, 1990). Essas rotinas, por serem fáceis de manipular, auxiliarão na implementação de um sistema de aprendizado.

1.4 Revisão da literatura

Apesar de se falar sobre *deep learning* como uma emocionante nova tecnologia, este tem uma história longa e rica, mas apresentando diversos nomes, os quais refletem diferentes pontos de vista filosóficos. Em termos gerais, ocorreram três ondas de desenvolvimento com níveis de popularidade variados: DL conhecido como *cybernetics* nas décadas de 1940 a 1960, DL conhecido como *connectionism* entre as décadas de 1980 e 1990 e o ressurgimento atual sob o nome de aprendizado profundo ou *deep learning* a partir de 2006 (GOODFELLOW; BENGIO; COURVILLE, 2016).

Alguns dos primeiros algoritmos de aprendizado que são reconhecidos hoje pretendiam ser modelos computacionais de aprendizado biológico, isto é, modelos de como o aprendizado acontece ou pode acontecer no cérebro. Como resultado, um dos nomes que o DL passou é o de *artificial neural networks* (ANNs). No entanto, o termo moderno "*deep learning*" vai além da perspectiva neurocientífica da atual geração de modelos de aprendizado de máquina. Ele apela a um princípio mais geral de aprendizado de vários níveis de composição, que podem ser aplicados em estruturas de aprendizado de máquina que não são necessariamente inspiradas em neurônios.

Uma das muitas contribuições do DL está no reconhecimento de fala. Depois de melhorar ao longo dos anos 90, as taxas de erro para reconhecimento de fala estagnaram a partir de 2000. A introdução do *deep learning* (Dahl et al., 2010; Denget al., 2010b; Seide et al., 2011; Hinton et al., 2012a) para o reconhecimento de fala resultou em uma queda repentina nas taxas de erro, com algumas taxas de erro cortadas pela metade. Exploramos essa história em mais detalhes na seção 12.3

O *deep learning* também contribuiu para outras ciências. As redes convolucionais modernas para reconhecimento de objetos fornecem um modelo de processamento visual que os neurocientistas podem estudar (AFRAZ DANIEL L.K. YAMINS, 2014). O *deep learning* também fornece ferramentas úteis para processar grandes quantidades de dados e fazer previsões úteis em campos científicos. Ele tem sido usado com sucesso para prever como as moléculas irão interagir, a fim de ajudar as empresas farmacêuticas a projetar novos medicamentos (DAHL; JAITLEY; SALAKHUTDINOV, 2014), a procurar partículas subatômicas (BALDI; SADOWSKI; WHITESON, 2014). Espera-se que o DL apareça em cada vez mais campos científicos no futuro.

1.5 Organização do trabalho

Este trabalho está estruturado em cinco capítulos. O **Capítulo 1** consiste em uma breve introdução ao tema do projeto e uma análise da literatura do problema. O **Capítulo 2** apresenta uma contextualização do problema nos âmbitos social, ambiental e

econômico. O **Capítulo 2** discorre a abordagem proposta para o problema. O **Capítulo 4** encerra o trabalho com as conclusões. E por fim, o **Capítulo 5** apresenta as propostas de continuidade para o Trabalho de Conclusão de Curso II.

2 Abordagem Proposta

No contexto de jogos digitais, treinar um agente para superar os jogadores humanos e otimizar sua pontuação pode nos ensinar como otimizar processos diferentes em uma variedade de subcampos diferentes e intrigantes (COMI, 2018). Uma solução proposta na literatura, obtendo ótimos resultados, e que tem como objetivo treinar um computador pra aprender e desenvolver estratégias para jogar diferentes jogos, é o *deep reinforcement learning* (DRL).

No presente trabalho é proposto a implementação de uma inteligência artificial que, utilizando um algoritmo de *deep reinforcement learning*, seja capaz de aprender a jogar diferentes jogos e desenvolver estratégias para maximizar sua pontuação. O DRL é uma abordagem do *deep learning* que, em contraste à abordagens mais comuns como o aprendizado supervisionado e não supervisionado, utiliza as técnicas de aprendizagem por reforço para treinar o agente.

Diante das peculiaridades e restrições do problema discutidos em 1.3, a biblioteca *Allegro* foi escolhida como a base para a implementação dos jogos que serão apresentados ao sistema. O *Allegro* é uma biblioteca multiplataforma destinada principalmente a jogos de vídeo e programação multimídia. A biblioteca fornece rotinas de baixo nível comumente necessárias na programação de jogos, como a criação de janelas, aceitação de entrada do usuário, carregamento de dados, desenho de imagens, reprodução de sons etc (HARGREAVES, 1990).

Por fim, para auxiliar a implementação do sistema, será utilizado uma ferramenta para desenvolvimento de inteligência artificial em jogos implementados em *Allegro* (SILVA, 2019), que oferece um ambiente facilitador ao estudo de soluções de IA aplicada em jogos. Essa ferramenta fornece funcionalidades como a exportação dos comandos básicos de um jogo, que precisam ser passados para o agente para que o mesmo tenha conhecimento dos limites físicos do ambiente no qual está inserido. Essa ferramenta permite que o pesquisador não fique limitado a um jogo existente, mas possa usar qualquer jogo que ele tenha acesso ao código fonte e feito em *Allegro*.

2.1 *Deep Learning*

O *deep learning* (DL) é uma área do aprendizado de máquina que propõe que os computadores aprendam com a experiência, se ajustem à novas entradas de dados e compreendam o mundo em termos de hierarquia de conceitos, sendo cada conceito definido por sua relação com conceitos mais simples. Ao reunir conhecimento a partir da

experiência, essa abordagem evita a necessidade dos operadores humanos de especificar formalmente todo o conhecimento que o computador precisa. Além disso, a hierarquia de conceitos permite que o computador aprenda conceitos complexos, construindo-os a partir de conceitos mais simples. O *deep learning* apresenta grande poder e flexibilidade a nos permitir o treinamento de computadores para cumprir tarefas específicas ao processar grandes quantidades de dados e reconhecer padrões nesses dados.

A **Figura 1** mostra como um sistema de *deep learning* representa o conceito de imagem de uma pessoa combinando conceitos mais simples, como cantos e contornos, que por sua vez são definidos em termos de arestas. O mapeamento de funções de um conjunto de pixels para uma identidade de objeto é muito complicado. O algoritmo de *deep learning* resolve essa dificuldade dividindo o mapeamento complicado desejado em séries de mapeamentos simples aninhados, cada um deles descrito por uma camada diferente do modelo. A entrada é apresentada na camada visível, assim denominada porque contém as variáveis que podemos observar. Em seguida, uma série de camadas ocultas extrai recursos cada vez mais abstratos da imagem. Essas camadas são chamadas de "ocultas" porque seus valores não são fornecidos nos dados; em vez disso, o algoritmo deve determinar quais conceitos são úteis para explicar os relacionamentos nos dados observados. A camada de saída obtém a identidade de objeto abstrata a partir dos conceitos obtidos pelas camadas ocultas.

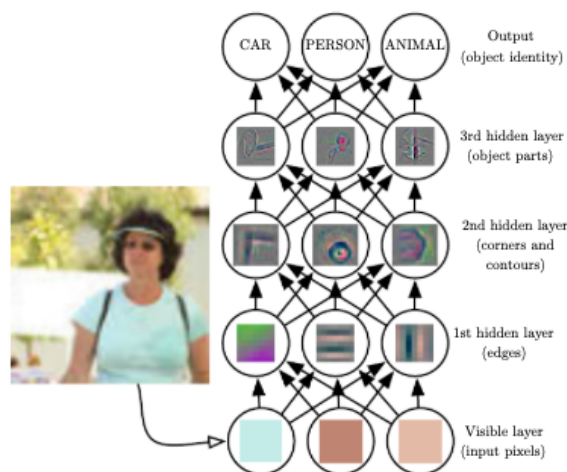


Figura 1: Ilustração de um modelo de aprendizado profundo. Dados os pixels, a primeira camada pode identificar facilmente as bordas, comparando o brilho dos pixels vizinhos. Dada a descrição das arestas pela primeira camada oculta, a segunda camada oculta pode procurar facilmente cantos e contornos estendidos, reconhecíveis como coleções de arestas. Dada a descrição da segunda camada oculta da imagem em termos de cantos e contornos, o terceiro setor oculto pode detectar partes inteiras de objetos específicos, encontrando coleções específicas de contornos e cantos. Finalmente, esta descrição da imagem em termos das partes do objeto que ela contém pode ser usada para reconhecer os objetos presentes na imagem. Imagem retirada de (GOODFELLOW; BENGIO; COURVILLE, 2016)

2.2 Reinforcement Learning

O aprendizado por reforço ou *reinforcement learning* (RL) é uma abordagem computacional para entender e automatizar o aprendizado direcionado a objetivos e a tomada de decisões. O aprendizado por reforço distingue-se de outras abordagens computacionais por sua ênfase na aprendizagem de um agente a partir da interação direta com seu ambiente, sem exigir supervisão exemplar ou modelos completos do ambiente (SUTTON; BARTO, 2018).

Em algoritmos de *reinforcement learning*, o agente não é informado sobre quais ações executar, mas, em vez disso, deve descobrir quais ações geram mais recompensa, através de tentativa e erro. Em alguns casos mais interessantes, as ações podem afetar não apenas a recompensa imediata, mas também a próxima situação e, com isso, todas as recompensas subsequentes. Essas duas características - pesquisa por tentativa e erro e recompensa atrasada - são as duas características distintivas mais importantes do aprendizado por reforço.

Além do agente e do ambiente, é interessante ressaltar outros elementos importantes de um sistema de aprendizado por reforço: a **política**, o **signal de recompensa** e a **função de valor**.

A **política** define a maneira que o agente deve se comportar em um determinado momento. Uma política é basicamente um mapeamento dos estados do ambiente para as ações a serem tomadas quando nesses estados. A política em casos mais simples tem a forma de uma função simples ou uma tabela de pesquisa, enquanto em casos mais complexos pode envolver cálculos mais extensivos. Em geral, as políticas podem ser estocásticas, especificando probabilidades para cada ação.

Um **signal de recompensa** define o objetivo de um problema de aprendizado por reforço. Em cada etapa, o ambiente envia ao agente de aprendizado por reforço um único número que funciona como uma recompensa para o agente. O único objetivo do agente é maximizar a recompensa total que recebe a longo prazo. O signal de recompensa define, portanto, quais são os eventos bons e ruins para o agente. O signal de recompensa é a base principal para alterar a política - se uma ação selecionada pela política for seguida por uma baixa recompensa, a política poderá ser alterada para selecionar outra ação nessa situação no futuro. Em geral, os sinais de recompensa podem ser funções estocásticas do estado do ambiente e das ações tomadas.

Enquanto o signal de recompensa indica o que é bom em um sentido imediato, uma **função de valor** especifica o que é bom a longo prazo. O valor de um estado representa a quantidade total de recompensa que um agente pode esperar acumular no futuro, a partir desse estado. Enquanto as recompensas determinam a conveniência imediata e intrínseca dos estados ambientais, os valores indicam a conveniência a longo prazo dos estados após

levar em conta os estados que provavelmente seguirão e as recompensas disponíveis nesses estados. Por exemplo, um estado sempre pode gerar uma recompensa imediata baixa, mas ainda tem um valor alto porque é seguido regularmente por outros estados que produzem recompensas altas. Ou o contrário poderia ser verdade.

A **Figura 2** mostra um diagrama de aprendizagem por reforço relacionando o agente de aprendizado com o ambiente no qual ele é inserido. O ambiente representa o mundo pelo qual o agente se move. O ambiente nada mais é do que um sistema que toma o estado atual e a ação do agente como entrada e retorna como saída a recompensa do agente e seu próximo estado.

Ambientes podem ser modelados como funções que transformam uma ação executada no estado atual, no próximo estado e uma recompensa. Já os agentes podem ser modelados como funções que transformam o novo estado e recompensam na próxima ação. Podemos conhecer a função do agente, mas não podemos conhecer a função do ambiente. É uma caixa preta onde só vemos as entradas e saídas. O aprendizado por reforço representa a tentativa de um agente de aproximar a função do ambiente, para que possamos enviar ações para o ambiente de caixa preta que maximize as recompensas que ele distribui (NICHOLSON, 2016).

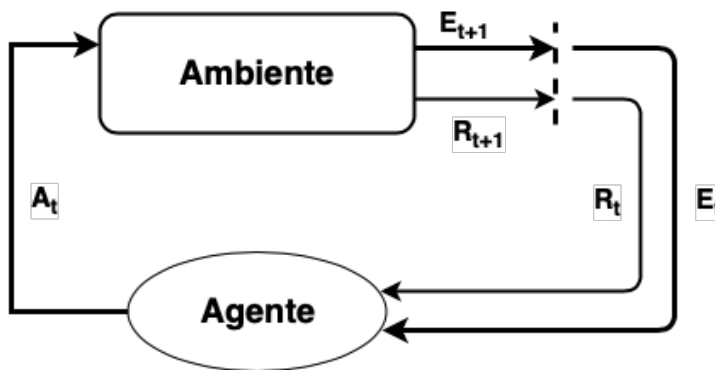


Figura 2: Diagrama de aprendizagem por reforço. No *loop de feedback* acima, os subscritos indicam as etapas de tempo t e $t + 1$, cada uma das quais se refere a estados diferentes: o estado no momento t e o estado no momento $t + 1$. A ação A_t de um agente é determinada por sua **política**, que por sua vez é uma função que depende do estado atual do sistema E_t . A política de um agente tem como objetivo maximizar a **função de valor** que é calculada utilizando o **signal de recompensa** R_t . O ambiente se comporta como um sistema caixa preta que transforma uma ação executada no estado atual A_t , no próximo estado E_{t+1} e uma recompensa R_{t+1} .

As escolhas de ação são feitas com base em julgamentos de valor. Buscamos ações que gerem estados de maior valor, e não de maior recompensa, porque essas ações obtêm a maior quantidade de recompensa para nós a longo prazo. Infelizmente, é muito mais difícil determinar valores do que determinar recompensas. As recompensas são basicamente

dadas diretamente pelo ambiente, mas os valores devem ser estimados e re-estimados a partir das sequências de observações que um agente faz ao longo de toda a sua vida útil.

2.3 Aplicação de DRL no treinamento de uma IA para aprender a jogar jogos em *Allegro*

O *deep reinforcement learning* (DRL) é uma abordagem do *deep learning* que, em contraste a abordagens mais tradicionais como o aprendizado supervisionado e não supervisionado, utiliza as técnicas de aprendizagem por reforço para treinar o agente. Essa abordagem consiste em fornecer ao sistema parâmetros relacionados ao seu estado e uma recompensa positiva ou negativa com base em suas ações. Nenhuma regra sobre o jogo é dada e, inicialmente, o agente não tem nenhuma informação sobre o que precisa fazer. O objetivo do sistema é descobrir e elaborar uma estratégia para maximizar sua pontuação - ou recompensa.

[TODO]

Referências

- AFRAZ DANIEL L.K. YAMINS, J. J. D. A. Neural mechanisms underlying visual object recognition. *Cold Spring Harb Symp Quant*, Cold Spring Harbor Laboratory Press; all rights reserved, 2014. Disponível em: <<https://doi.org/10.1101/sqb.2014.79.024729>>. Acesso em: 2 ago 2019. Citado na página 21.
- BALDI, P.; SADOWSKI, P.; WHITESON, D. Searching for exotic particles in high-energy physics with deep learning. *Nature Communications*, v. 5, n. 1, p. 4308, 2014. Disponível em: <<https://doi.org/10.1038/ncomms5308>>. Citado na página 21.
- COMI, M. *How to teach AI to play Games: Deep Reinforcement Learning*. 2018. Disponível em: <<https://towardsdatascience.com/how-to-teach-an-ai-to-play-games-deep-reinforcement-learning-28f9b920440a>>. Acesso em: 8 out 2019. Citado 2 vezes nas páginas 20 e 23.
- DAHL, G. E.; JAITLEY, N.; SALAKHUTDINOV, R. *Multi-task Neural Networks for QSAR Predictions*. 2014. Citado na página 21.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. [S.l.]: MIT Press, 2016. <<http://www.deeplearningbook.org>>. Citado 3 vezes nas páginas 17, 21 e 24.
- HARGREAVES, S. *Allegro*. 1990. Disponível em: <<https://liballeg.org/index.html>>. Acesso em: 8 out 2019. Citado 2 vezes nas páginas 20 e 23.
- HSU, F.-H. *Behind Deep Blue: Building the Computer That Defeated the World Chess Champion*. Princeton, NJ, USA: Princeton University Press, 2002. ISBN 0691090653. Citado na página 19.
- MENABREA, L. et al. *Sketch of the Analytical Engine invented by Charles Babbage ... with notes by the translator. Extracted from the 'Scientific Memoirs,' etc. [The translator's notes signed: A.L.L. ie. Augusta Ada King, Countess Lovelace.]*. R. & J. E. Taylor, 1843. Disponível em: <<https://books.google.com.br/books?id=hPRmnQEACAAJ>>. Citado na página 17.
- NICHOLSON, C. *A Beginner's Guide to Deep Reinforcement Learning*. 2016. Disponível em: <<https://skymind.ai/wiki/deep-reinforcement-learning>>. Acesso em: 8 out 2019. Citado na página 26.
- RODRIGUES, J. *O que é o Processamento de Linguagem Natural?* 2017. Disponível em: <<https://medium.com/botsbrasil/o-que-é-o-processamento-de-linguagem-natural-49ece9371cff>>. Acesso em: 2 set 2019. Citado na página 17.
- SILVA, A. P. Ambiente para desenvolvimento de inteligência artificial em jogos allegro. Departamento de Ciência da Computação (DCC) da Universidade Federal de Minas Gerais (UFMG), Belo horizonte, Brasil, 2019. Disponível em: <https://github.com/artphil/allegro_game_ai>. Acesso em: 8 out 2019. Citado na página 23.

SUTTON, R. S.; BARTO, A. G. *Reinforcement Learning: An Introduction*. 2. ed. [S.l.]: MIT Press, 2018. <<http://incompleteideas.net/book/the-book.html>>. Citado 2 vezes nas páginas 17 e 25.