

Aula 1

Breve introdução à Inteligência Artificial

O que é Inteligência Artificial?

A Inteligência Artificial (IA) é o campo da ciência da computação que desenvolve sistemas capazes de realizar tarefas que normalmente exigiriam inteligência humana: tomar decisões, reconhecer padrões, adaptar-se ao ambiente e aprender com a experiência, que será o foco desse curso.

Algumas motivações da IA

- Automatizar tarefas perigosas, repetitivas ou em ambientes hostis (como minas, zonas de desastre ou o fundo do mar);
- Expandir a capacidade humana em tarefas cognitivas complexas, como diagnóstico médico assistido por computador;
- Melhorar a eficiência e a tomada de decisão em setores como logística, energia, transporte, agricultura e finanças;
- Criar sistemas adaptativos capazes de aprender com dados e se ajustar a mudanças dinâmicas no ambiente;
- Facilitar interações mais naturais entre humanos e máquinas por meio de linguagem natural, visão computacional ou reconhecimento de emoções;
- Reduzir custos operacionais em empresas por meio de otimizações inteligentes baseadas em dados;
- Ampliar o acesso a serviços essenciais (educação, saúde, finanças) via agentes inteligentes e personalizados;
- **Encontrar maneiras de se adaptar para resolver problemas, e como otimizar essa adaptação de maneira eficiente (esse será um dos focos desse curso).**

Alguns problemas que podem ser resolvidos com IA

- Diagnóstico médico automatizado com apoio à decisão para radiologistas, dermatologistas ou cardiologistas;
- Reconhecimento e tradução de fala em tempo real (ex: assistentes virtuais multilíngues, legendas automáticas);
- Análise e classificação automática de imagens (ex: veículos autônomos, drones de vigilância, agricultura de precisão);
- Previsão de séries temporais, como demanda de energia elétrica, fluxo de tráfego, ou variações de mercado financeiro;
- Otimização logística (ex: planejamento de rotas em tempo real para entregas, controle de estoques, cadeia de suprimentos);
- Desenvolvimento de agentes autônomos: carros autônomos, robôs domésticos, drones de inspeção;
- Sistemas de recomendação personalizados para filmes, músicas, compras, notícias ou cursos online;
- Detecção de fraudes em transações bancárias e sistemas de cibersegurança adaptativa;
- Criação de conteúdo com IA (textos, imagens, vídeos, música);
- Apoio a decisões judiciais ou administrativas com base em análise de grandes volumes de documentos legais.

Paradigmas de Aprendizado em IA

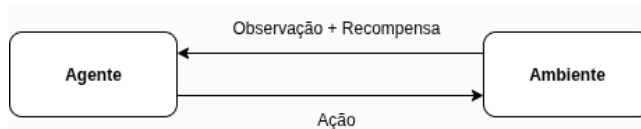
A aprendizagem de máquina (*machine learning*) é uma das áreas centrais da IA, podendo ser dividida em três grandes paradigmas:

Paradigma	Características Principais	Exemplos
Aprendizado Supervisionado	Aprende com dados rotulados (entrada → saída)	Classificação de e-mails, diagnóstico médico
Aprendizado Não Supervisionado	Descobre padrões em dados não rotulados	Agrupamento de clientes, redução de dimensionalidade
Aprendizado por Reforço (RL)	Aprende por tentativa e erro, com recompensas/punições	Jogar xadrez, controlar um robô, conduzir um carro autônomo

Elementos Básicos do Aprendizado por Reforço

Loop Ambiente-Agente

Em problemas de aprendizado por reforço, teremos dois principais elementos que irão interagir entre si: o agente e o ambiente. O agente é aquele que realiza ações (como um robô) e que busca melhorar as tomadas de decisão. O ambiente é aquele que sofre a ação e retorna uma observação (informações sobre a configuração atual do ambiente) ao agente. A figura abaixo mostra esse loop de interação.



Note que o agente também recebe um sinal de recompensa, o qual será responsável por informar ao agente quão boa (ou ruim) foi a ação tomada anteriormente. Vale observar que, em algumas situações, a recompensa pode ser representada como uma função interna do agente, facilitando sua manipulação e entendimento para alguns casos. Com isso, estabelecemos que o objetivo do agente é maximizar o cumulativo de recompensas.

Tipos de Ambientes

Em aprendizado por reforço, os termos observação e estado possuem, apesar de semelhantes, algumas diferenças. A distinção ocorre na origem das informações acerca do ambiente: se essas são advindas apenas das percepções do agente, então temos uma observação; caso contrário, se as informações são completas e advindas do próprio ambiente, então temos um estado. Com base nisso, Nos problemas de aprendizado por reforço, podemos encontrar dois tipos de ambientes:

- Ambiente Totalmente Observável: neste caso, o agente possui conhecimento de todo o estado do ambiente (observação e estado são iguais). Um exemplo é de um robô que se move por uma sala e que conhece sua posição atual, posição final desejada e obstáculos. Apesar de ser um cenário ideal, ele é menos comum e mais difícil de processar pela quantidade de dados do ambiente.
- Ambiente Parcialmente Observável: neste caso, o agente observa indiretamente o estado (observação e estado são diferentes). Um exemplo é de um robô que apenas conhece o ambiente a partir de uma câmera acoplada a sua estrutura. Esta é a situação mais comum.

O que é Aprendizado por Reforço (Reinforcement Learning)?

O *Reinforcement Learning* (RL) é uma abordagem de aprendizado em que o sistema aprende por meio da **interação com o ambiente**, ajustando seu comportamento com base nas **recompensas ou punições** que recebe ao longo do tempo.

Reinforcement Learning é aprender a tomar decisões com base na experiência, explorando ações para maximizar recompensas no longo prazo.

Exemplos Intuitivo

Considere um bebê aprendendo a andar. Sem receber instruções explícitas, ele tenta se levantar, cai (punição), tenta novamente, equilibra-se por mais tempo (recompensa). Esse ciclo de tentativa e erro é a essência do RL.

Outro exemplo seria o caso da adestração de um cachorro. Para aprender a dar a pata, o dono primeiro sugere que ele deve mexer a pata, dando um petisco (recompensa) após ele realizar a ação de dar a pata. Com o tempo, o cachorro percebe que sempre ganha o petisco após dar a pata, então ele aprende qual deve ser o movimento realizado para ganhar mais petiscos.

Por que estudar Aprendizado por Reforço?

Enquanto outros paradigmas de aprendizado focam em reconhecimento de padrões ou agrupamento de dados, o Aprendizado por Reforço (RL) se destaca por lidar com problemas onde a tomada de decisão precisa considerar consequências de longo prazo e interação contínua com o ambiente.

- Muitos problemas reais envolvem decisões sequenciais, onde o impacto de uma ação não é imediato, mas afeta os resultados futuros (ex: dirigir um carro, administrar um estoque, jogar um jogo).
- Nem sempre há um "gabarito" com exemplos corretos — o agente precisa aprender a partir das próprias experiências, avaliando se suas ações resultam em boas ou más consequências.
- O RL permite desenvolver sistemas com comportamento adaptativo: capazes de melhorar seu desempenho com o tempo, com base em recompensas e punições do ambiente.
- RL é o paradigma mais próximo do aprendizado que observamos em humanos e animais, sendo fundamental para entender inteligência em sistemas naturais e artificiais.
- Avanços recentes em RL levaram a resultados impressionantes: de agentes super-humanos em jogos (ex: AlphaGo, AlphaStar) até controle de robôs complexos e sistemas de negociação autônoma.

Diferenças em Relação a Outros Paradigmas

Aspecto	Supervisionado	Não Supervisionado	Por Reforço (RL)
Precisa de rótulos?	Sim	Não	Não
Aprende com interação?	Não	Não	Sim
Utiliza recompensas?	Não	Não	Sim (inclusive atrasadas)
Objetivo principal	Prever saída correta	Descobrir estrutura	Maximizar recompensa total
Lida com decisões sequenciais?	Parcialmente	Não	Sim

Exemplos de Aplicações do RL na computação

- Um sistema que aprende a jogar xadrez somente ao disputar partidas (sem conhecimento prévio das regras);
- Um robô que decide quando explorar um novo ambiente ou recarregar sua bateria;
- Algoritmos de recomendação que se adaptam ao comportamento do usuário com base em cliques e rejeições;
- Sistemas industriais que ajustam parâmetros operacionais para maximizar rendimento e reduzir custo;
- O sistema de controle de um helicóptero, que precisa parar em postos para reabastecer e chegar ao destino final.

Aplicações do RL - LLMs

A técnica de RL pode ser utilizada no treinamento de **modelos de linguagem** para obter respostas mais naturais e úteis. Nesse processo, parte-se de um LLM pré-treinado (modelo base) e de um conjunto de *prompts*. Para cada prompt, o modelo gera várias respostas, que são comparadas e ordenadas por avaliadores humanos. Essas preferências servem para treinar um **modelo de recompensa** que estima a qualidade percebida de uma resposta no seu contexto, funcionando como um substituto das preferências dos usuários. Em seguida, o LLM é ajustado para produzir saídas que maximizem essa estimativa, fechando o ciclo de aprendizado por reforço. Esse processo, usado em sistemas como o ChatGPT, incorpora feedback humano ao comportamento do modelo e o alinha a critérios de utilidade, clareza e segurança.

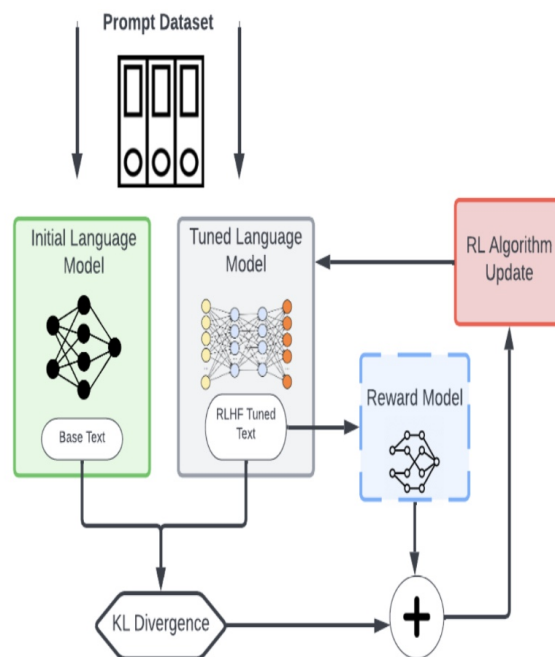


Figura 1: RL com Feedback humano para treinar modelos de Linguagem

Aplicações do RL – AWS DeepRacer

O **AWS DeepRacer** é uma competição da AWS em que carrinhos autônomos devem percorrer uma pista no menor tempo possível sem sair dos limites. Cada veículo usa visão computacional (por meio de uma câmera frontal) para obter o estado do ambiente e, a partir disso, um modelo de aprendizado por reforço decide uma ação a tomar. O comportamento do carrinho é guiado por uma *função de recompensa* definida pelo participante, que toma a forma de um script em python que retorna um valor escalar, representando a recompensa que um modelo tem em um dado estado.



Loop de execução (inferência). Em tempo de corrida, o sistema opera em malha fechada, como ilustrado no diagrama do “Loop Execução”: do estado atual observado pela câmera, a política escolhe uma ação; o carrinho executa essa ação; o ambiente muda e um novo estado é observado. Esse ciclo rápido *estado* → *ação* → *nova observação* se repete ao longo de toda a volta. Embora a recompensa seja usada principalmente para treinar o agente, o mesmo fluxo também gera os pares de *estado*, *ação* e a transição para o próximo estado que servem de base para análise e melhoria posterior.

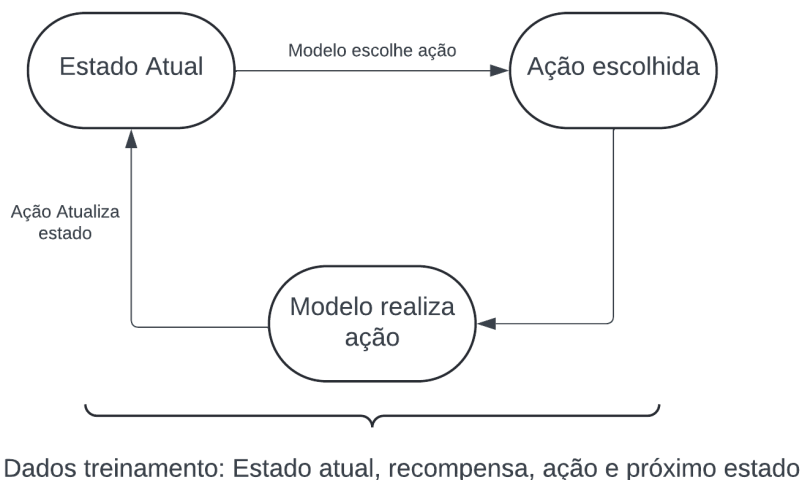


Figura 2: Processo de inferência do modelo de RL usado no DeepRacer

Treinamento no simulador. No treinamento, representado no diagrama “Treinamento”, cada episódio é composto por transições do tipo (estado s_t , ação a_t , próximo estado s_{t+1}), às quais se associa uma recompensa calculada por uma função que lê parâmetros do simulador (posição na pista, velocidade, proximidade de bordas etc.). O agente é então ajustado para aumentar a soma dessas recompensas ao longo do episódio, refinando a política para escolher melhores ações nas próximas iterações. Após treinar e validar no simulador, o modelo é implantado no carrinho físico para competir, repetindo o ciclo de melhoria até atingir desempenho competitivo na pista.

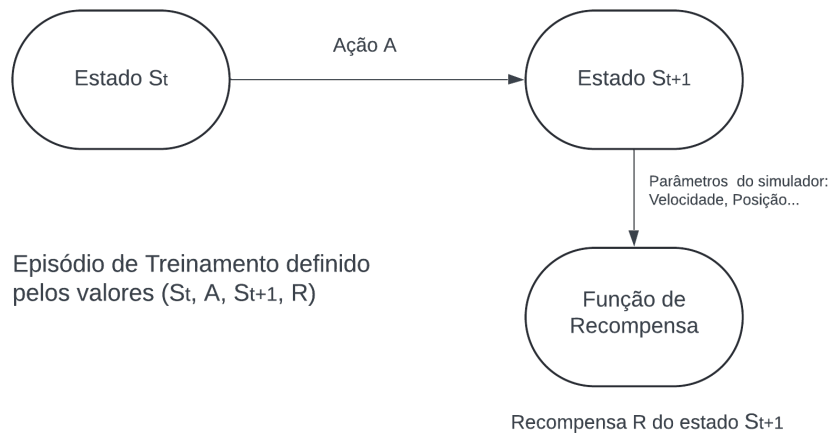


Figura 3: Processo de treinamento do modelo de RL usado no DeepRacer