

Este é o CS50




Introdução à Ciência da Computação (CS50)


OpenCourseWare

Doar  (<https://cs50.harvard.edu/donate>)


David J. Malan (<https://cs.harvard.edu/malan/>)

malan@harvard.edu

 (<https://www.facebook.com/dmalan>)  (<https://github.com/dmalan>) 

(<https://www.instagram.com/davidjmalan/>)  (<https://www.linkedin.com/in/malan/>)

 (<https://www.reddit.com/user/davidjmalan>) 

(<https://www.threads.net/@davidjmalan>)  (<https://twitter.com/davidjmalan>)

Inteligência artificial

- [Bem-vindo!](#)
- [IA generativa](#)
- [IA e CS50](#)
- [Engenharia Rápida](#)
- [IA](#)
- [Árvores de decisão](#)
- [Minimax](#)
- [Aprendizado de máquina](#)
- [Aprendizado profundo](#)
- [Modelos de linguagem de grande porte](#)
- [Resumindo](#)

Bem-vindo!

- Em círculos de ciência da computação e programação, "*rubber ducking*" ou "*rubber ducking debugging*" é o ato de conversar com um objeto inanimado para conseguir *resolver* um problema complexo.
- Mais recentemente, a CS50 criou seu próprio depurador de "pato de borracha" no [CS50.ai](https://cs50.ai) (<https://cs50.ai>), que usa inteligência artificial como forma de interagir com os alunos e ajudá-los a resolver seus próprios problemas mais complexos.

- Os alunos que interagem com essa ferramenta podem começar a compreender o potencial que a IA pode oferecer ao mundo.

IA generativa

- Diversas ferramentas de IA criaram o potencial para que imagens geradas artificialmente entrem no mundo.
- Até anos atrás, a maioria dessas ferramentas apresentava diversos indícios que podiam levar um observador a crer que uma imagem havia sido gerada por IA.
- No entanto, as ferramentas estão se tornando cada vez mais eficientes na geração dessas imagens.
- De fato, com o avanço da tecnologia, em breve será quase, senão totalmente, impossível detectar tais imagens a olho nu.
- O software também adquiriu a capacidade de modificar imagens individuais dentro de um vídeo.
- Uma ferramenta de ponta muito conhecida é a ferramenta de geração de texto *chatGPT*.

IA e CS50

- Em CS50, não permitimos o uso do ChatGPT.
- A disciplina CS50 oferece orientações claras aos alunos, em nossa política de honestidade acadêmica, sobre o que é considerado uso *razoável* e *não razoável* de IA.
- Permitimos o uso do nosso próprio depurador de "pato de borracha" no CS50.ai (<https://cs50.ai>)
- No CS50, aproveitamos as ferramentas do Azure e do OpenAI, juntamente com nosso próprio banco de dados vetorial que contém informações muito recentes de nossas palestras e ofertas mais recentes, para fornecer nossa ferramenta de depuração "pato de borracha".
- O CS50.ai (<https://cs50.ai>) é amplamente utilizado por nossos alunos em todo o mundo. 201 mil alunos e professores geraram aproximadamente 35 mil sugestões por dia desde o lançamento da ferramenta. Ou seja, um total de 9,4 milhões de sugestões foram fornecidas ao simulador.
- Nosso depurador de patinho de borracha (<https://cs50.ai>) pode fornecer ajuda conceitual com conceitos de ciência da computação.
- Além disso, o depurador de patinho de borracha (<https://cs50.ai>) pode ajudar os alunos a escreverem códigos mais eficientes.
- Além disso, o depurador em forma de patinho de borracha (<https://cs50.ai>) pode ajudar quando um aluno está com dificuldades em uma de suas tarefas. Por exemplo, os alunos podem encontrar erros que os impedem de progredir em suas tarefas. Quando os alunos

se deparam com um obstáculo, eles não precisam esperar que a equipe de suporte esteja disponível.

- Pedagogicamente, nosso objetivo é fornecer um tutor individual para os alunos 24 horas por dia.

Engenharia Rápida

- A *engenharia de perguntas* é a forma pela qual um indivíduo pode fazer boas perguntas a uma IA.
- Usamos um *comando do sistema* para ensinar a IA a interagir com os usuários. Ensinamos a IA a trabalhar com os alunos usando esse comando.
- Os *comandos do usuário* são aqueles fornecidos pelos usuários para interagir com a IA. Com esses comandos, os alunos interagem com a IA.
- Podemos implementar isso em código da seguinte forma:

```
import os
from dotenv import load_dotenv
load_dotenv()

from openai import OpenAI

client = OpenAI(api_key=os.environ["API_KEY"])

system_prompt = "You are a friendly and supportive teaching assistant for CS50"

user_prompt = input("What's your question? ")

chat_completion = client.chat.completions.create(
    messages=[
        {"role": "system", "content": system_prompt},
        {"role": "user", "content": user_prompt}
    ],
    model="gpt-4o"
)

response_text = chat_completion.choices[0].message.content

print(response_text)
```

Você pode executar este código com `python chat.py`.

IA

- A inteligência artificial está presente em nossas vidas há muito tempo! Os softwares se adaptaram aos usuários há bastante tempo. Os algoritmos buscam padrões em spam, no reconhecimento de escrita à mão, na criação de recomendações de filmes e em jogos.

- Em jogos, por exemplo, instruções passo a passo podem permitir que um adversário computadorizado jogue uma partida de Breakout.

Árvores de decisão

- *Árvores de decisão* são usadas por um algoritmo para decidir qual decisão tomar.
- Por exemplo, no jogo Breakout, um algoritmo pode considerar qual escolha fazer com base nas instruções do código:

```
While game is ongoing:  
  If ball is left of paddle:  
    Move paddle left  
  Else if ball is right of padding:  
    Move paddle right  
  Else:  
    Don't move paddle
```

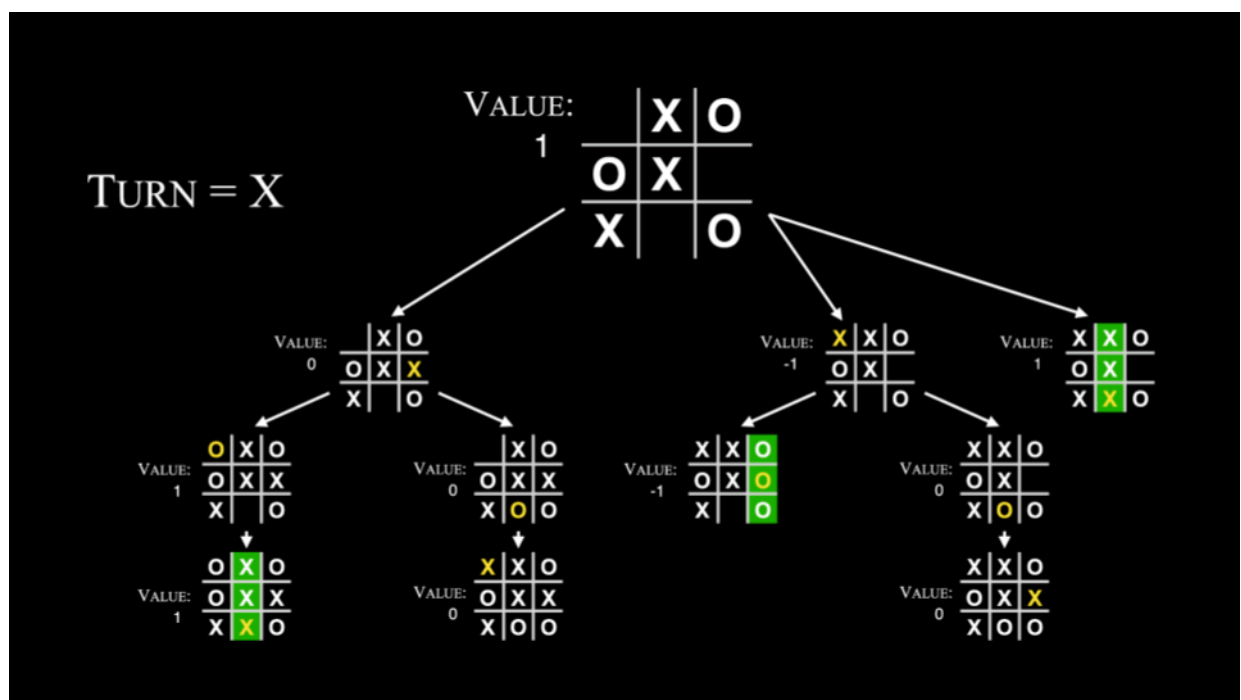
- Na maioria dos jogos, eles tentam minimizar o número de cálculos necessários para competir com o jogador.

Minimax

- A IA costuma ser boa em jogos porque reduz movimentos e resultados a valores matemáticos.
- Você pode imaginar um algoritmo que classifique os resultados como positivos, negativos e neutros.
- No jogo da velha, a IA pode considerar um tabuleiro onde o computador ganha como `1` e um onde o computador perde como `-1`.
- Você pode imaginar como um computador pode analisar uma árvore de decisão com possíveis resultados e atribuir pontuações a cada movimento potencial.
- O computador tentará vencer maximizando sua própria pontuação.
- No contexto do jogo da velha, o algoritmo pode conceituar isso da seguinte forma:

```
If player is X:  
  For each possible move:  
    Calculate score for board  
  Choose move with highest score  
  
Else if player is O:  
  For each possible move:  
    Calculate score for board  
  Choose move with lowest score
```

- Isso pode ser visualizado da seguinte forma:



- Graças à sua enorme capacidade de processamento, os computadores conseguem lidar com uma quantidade imensa de resultados possíveis. No entanto, os computadores que carregamos no bolso ou que estão sobre nossas mesas podem não ser capazes de calcular trilhões de opções. É aí que o aprendizado de máquina pode ajudar.

Aprendizado de máquina

- Aprendizado de máquina é uma forma pela qual um computador pode aprender por meio de reforço.
- Um computador pode aprender a virar uma panqueca.
- Um computador pode aprender a jogar "O Chão é Lava".
- O computador repete o teste diversas vezes para descobrir quais comportamentos devem ser repetidos e quais não devem.
- Em muitos algoritmos baseados em IA, existem os conceitos de *exploração versus aproveitamento*, onde a IA pode tentar aleatoriamente algo que não seja considerado ideal. A aleatoriedade pode gerar melhores resultados. Isso pode ser representado em código da seguinte forma:

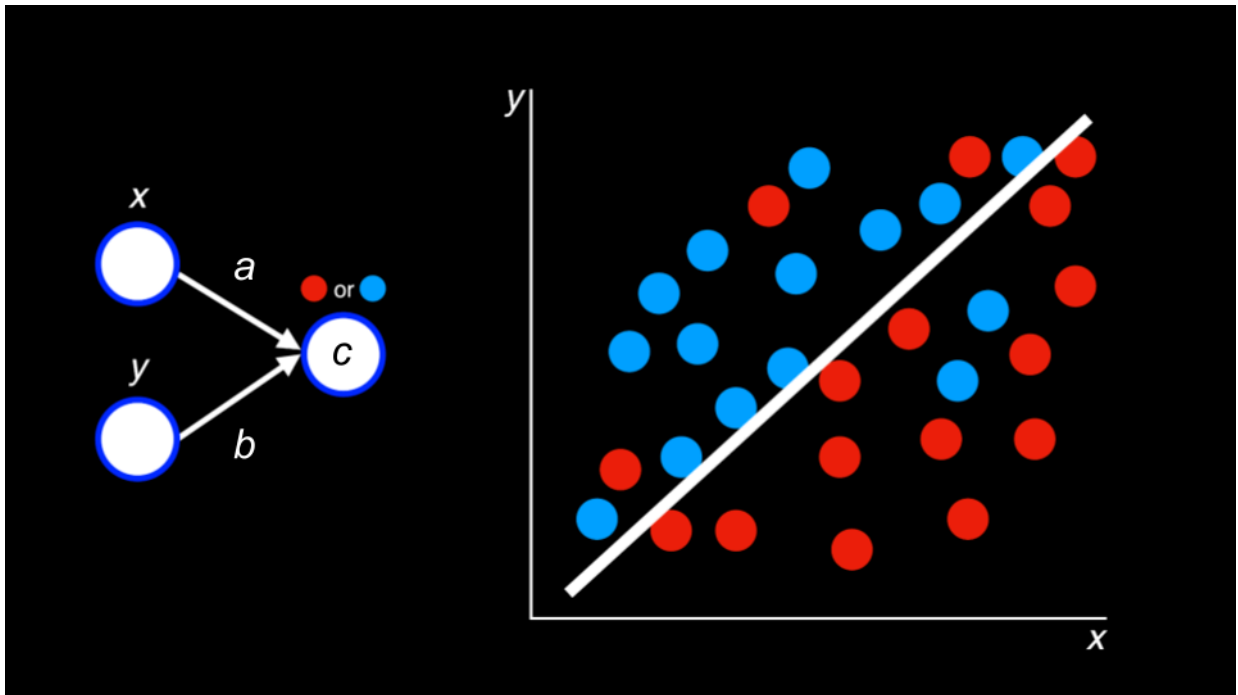
```
epsilon = 0.10

If random() < epsilon:
    Make a random move
Else:
    Make the move with the highest value
```

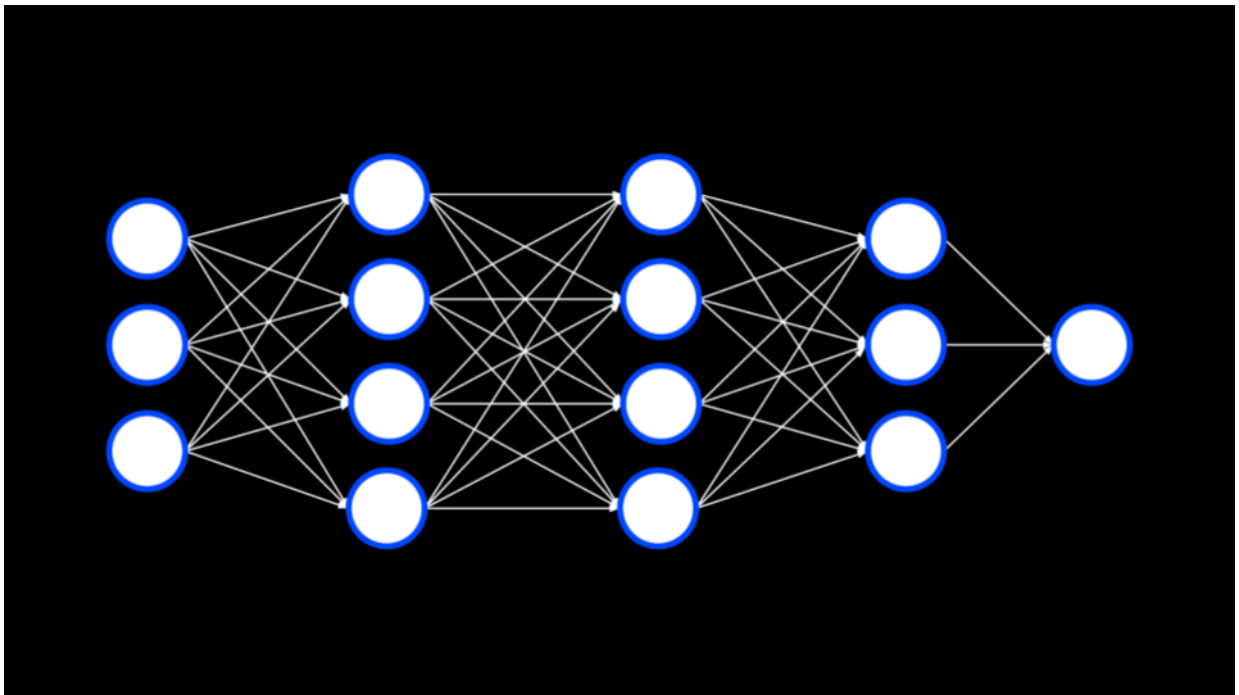
Aprendizado profundo

- A *aprendizagem profunda* utiliza redes neurais para explorar problemas e soluções.

- Por exemplo, o aprendizado profundo pode tentar prever se um ponto azul ou vermelho aparecerá em algum lugar em um gráfico. Considere a seguinte imagem:



- Os dados de treinamento existentes são usados para prever um resultado. Além disso, a IA pode criar mais dados de treinamento para descobrir novos padrões.
- O aprendizado profundo cria nós (ilustrados abaixo) que associam entradas e saídas.



Modelos de linguagem de grande porte

- *Os grandes modelos de linguagem* são modelos massivos que fazem previsões com base em enormes quantidades de treinamento.
- Há poucos anos, a IA não era muito boa em completar e gerar frases. O Google publicou um artigo em 2017 sobre como essas IAs podem ter sua *atenção* direcionada para as

relações entre várias palavras.

- Os Transformers pré-treinados generativos (GPT, na sigla em inglês) são treinados nas relações entre palavras.
- A IA codifica palavras em *representações vetoriais* para encontrar relações entre elas. Assim, por meio de um enorme treinamento, uma rede neural massiva pode prever a associação entre palavras, resultando na capacidade da IA generativa de gerar conteúdo e até mesmo manter conversas com os usuários.

Resumindo

Nesta lição, você aprendeu sobre algumas das tecnologias por trás [do CS50.ai. \(https://CS50.ai\)](https://CS50.ai)
Especificamente, discutimos...

- Geração de Imagens
- IA generativa
- IA e CS50
- Engenharia Rápida
- IA
- Árvores de decisão
- Minimax
- Aprendizado de máquina
- Aprendizado profundo
- Modelos de linguagem de grande porte

Este foi o CS50!