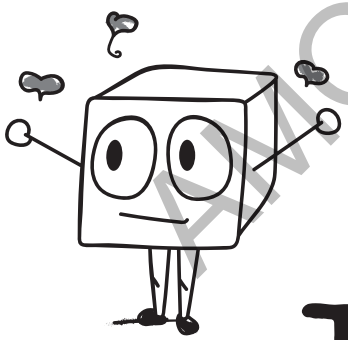
Como a **Inteligência Artificial**
funciona e por que está fazendo
do mundo um lugar mais estranho

VOCÊ PARECE UMA COISA E EU TE AMO



Janelle Shane

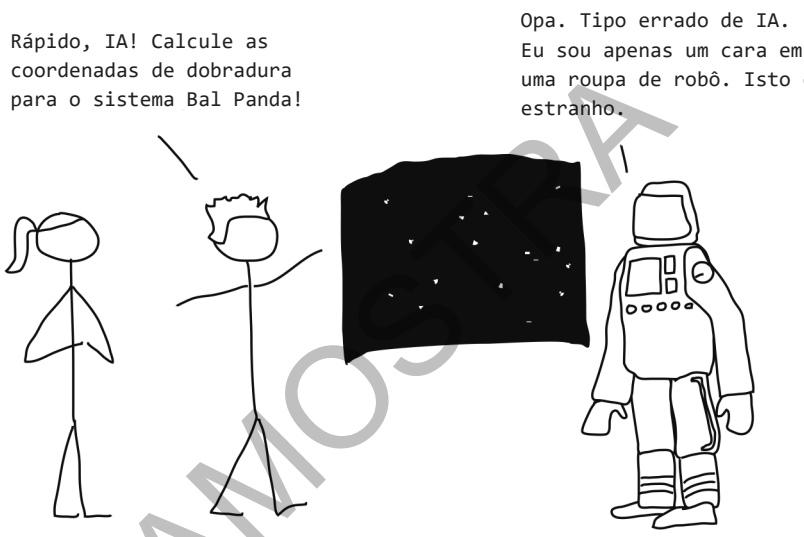
Nomeada uma das 100 pessoas mais
criativas nos negócios pela *Fast Company*

ALTA BOOKS
EDITORA
Rio de Janeiro, 2022

Sumário

CAPÍTULO 1

O que é IA?



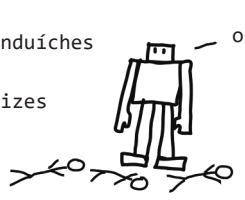
Se parece que a IA está em todo lugar, é em parte porque "inteligência artificial" significa muitas coisas, depende se você está lendo ficção científica, ou vendendo um novo aplicativo, ou fazendo pesquisas acadêmicas. Quando alguém diz que tem um chatbot com inteligência artificial, devemos esperar que ele tenha opiniões e sentimentos como o fictício C-3PO? Ou é apenas um algoritmo que aprendeu a adivinhar como os humanos provavelmente responderão a uma determinada frase? Ou uma planilha que combina as palavras da sua pergunta com uma biblioteca de respostas pré-formuladas?

Ou um humano mal pago que digita todas as respostas de algum local remoto? Ou — até — uma conversa completamente escrita, na qual humanos e IA estão lendo frases escritas por humanos como se fossem personagens de uma peça? De maneira confusa, em vários momentos, tudo isso foi referido como IA.

Para os fins deste livro, usarei o termo IA da maneira como é usado hoje, principalmente pelos programadores: para me referir a um estilo específico de programa de computador chamado algoritmo de aprendizado de máquina. Esta lista mostra muitos dos termos que abordarei neste livro e onde eles se enquadram de acordo com essa definição.

Coisas chamadas de IA

Chamada de IA neste livro	Neste livro, mas não é IA
Algoritmos de aprendizado de máquina	IAs de ficção científica
Aprendizagem profunda	Programas baseados em regras
Redes neurais	Humanos fantasiados de robô
Redes neurais recorrentes	Robôs lendo roteiros
Cadeias de Markov	Humanos contratados para fingir serem IA
Florestas Aleatórias	Baratas autoconscientes
Algoritmos genéticos	Girafas fantasmas
Rede Contraditória Generativa	
Aprendizado por reforço	
Texto preditivo	
Classificadores de sanduíches mágicos	
Bots assassinos infelizes	



Tudo o que eu estou chamando de "IA" neste livro também é um algoritmo de aprendizado de máquina — vamos falar sobre o que é isso.

TOC, TOC, QUEM É?

Para identificar uma IA selvagem, é importante saber a diferença entre os **algoritmos de aprendizado de máquina** (o que chamamos de IA neste livro) e os programas tradicionais (o que os programadores chamam de **baseado em regras**). Se você já fez uma programação básica ou até se já usou HTML para criar um site, você usou um programa baseado em regras. Você cria uma lista de comandos ou regras em um idioma que o computador possa entender, e o computador faz exatamente o que você diz. Para resolver um problema com um programa baseado em regras, é necessário conhecer todas as etapas necessárias para concluir a tarefa do programa bem como saber descrever cada uma dessas etapas.

Mas um algoritmo de aprendizado de máquina descobre as regras por si próprio via tentativa e erro, avaliando seu sucesso de acordo com os objetivos que o programador especificou. O objetivo pode ser uma lista de exemplos a serem imitados, uma pontuação de jogo a ser aumentada ou qualquer outra coisa. À medida que a IA tenta alcançar esse objetivo, ela pode descobrir regras e correlações que o programador nem sabia que existiam. Programar uma IA é quase como ensinar uma criança, mais do que programar um computador.

Programação baseada em regras

Digamos que eu queira usar a programação baseada em regras mais habitual para ensinar um computador a contar piadas de toc-toc. A primeira coisa que eu faria seria descobrir todas as regras. Analisaria a estrutura das piadas e descobria que existe uma fórmula básica, como segue:

```
Toc, Toc.
Quem é?
[Nome]
[Nome] quem?
[Nome] [bordão]
```

Depois de definir essa fórmula, há apenas duas inserções livres que o programa pode controlar: [Nome] e [bordão]. Agora, o problema é reduzido a apenas gerar esses dois itens. Mas ainda preciso de regras para gerá-los.

Eu poderia configurar uma lista de nomes válidos e uma lista de bordões válidos, da seguinte maneira:

```
Nomes          Bordões
Alface         -me o favor, está frio aqui fora!
Andy           logo, está frio aqui fora!
Seis           vão me deixar entrar?
Vozes          não vão me deixar entrar?
```

Agora, o computador pode produzir piadas de toc-toc ao escolher um par de nome–bordão da lista e inseri-los no modelo. Isso não cria *novas* piadas, mas só me dá piadas que eu já conheço. Eu posso tentar tornar as coisas interessantes ao permitir que [está frio aqui!] seja substituído por algumas frases diferentes: [estou sendo atacado por enguias!] e [para que eu não me transforme em um horror indescritível]. Então o programa pode gerar uma nova piada:

```
Toc, Toc
Quem é?
Andy.
Andy quem?
Andy logo, estou sendo atacado por enguias!
```

Eu poderia substituir [enguias] por [uma abelha brava] ou [uma raia manta] ou qualquer tipo de coisa. Então eu conseguiria fazer o computador gerar ainda mais piadas novas. Com regras suficientes, eu poderia gerar centenas de piadas.

Dependendo do nível de sofisticação que eu estou buscando, posso gastar muito tempo criando regras mais avançadas. Eu poderia encontrar uma lista de trocadilhos existentes e descobrir uma maneira de transformá-los em formato de bordão. Eu poderia até tentar programar regras de pronúncia, rimas, semi-homófonos,

referências culturais e assim por diante na tentativa de fazer com que o computador as recombinasse em trocadilhos interessantes. Se eu for esperta, posso fazer com que o programa gere novos trocadilhos nos quais ninguém jamais pensou. (Embora uma pessoa que já tentou isso descobriu que a lista de provérbios do algoritmo continha palavras e frases tão antigas ou obscuras que quase ninguém conseguiu entender suas piadas.) Não importa quão sofisticadas sejam minhas regras de criação de piadas, eu ainda estou dizendo ao computador exatamente como resolver o problema.

Treinando a IA

Mas quando treino a IA para contar piadas de toc-toc, eu não faço as regras. A IA precisa descobrir essas regras por conta própria.

A única coisa que dou a ela é um conjunto de piadas de toc-toc existentes e instruções que são essencialmente: "Aqui estão algumas piadas; tente criar mais piadas." E os materiais que dou para ela trabalhar? Um balde de letras aleatórias e pontuação.

Então eu saio para tomar café.

A IA começa a trabalhar.

A primeira coisa que ela faz é tentar adivinhar algumas letras de algumas piadas de toc-toc. Ela está fazendo suposições 100% aleatórias neste momento, então o primeiro palpite pode ser qualquer coisa. Digamos que adivinhe algo como "qasdnw, m sne? Mso d". Até onde ela sabe, é assim que você conta uma piada de toc-toc.

Em seguida, a IA analisa como essas piadas de toc-toc *realmente* deveriam ser. Provavelmente, está muito errada. "Tudo bem", diz a IA, e ela ajusta sutilmente sua própria estrutura para que adivinhe com um pouco mais de precisão da próxima vez. Há um limite para o quão drasticamente ela pode mudar a si mesma, porque não queremos que ela tente memorizar cada novo pedaço de texto que vir. Mas, com um mínimo de ajustes, a IA pode descobrir que, se adivinhar nada além de *t* e espaços, pelo menos estará certa algumas vezes. Depois de analisar

um lote de piadas de toc-toc e fazer uma rodada de correções, sua ideia de piada de toc-toc é mais ou menos assim:

```
  t t t  t   t t
tt     t tttot
 t tttt
t
tt
 tt   t   tt
teott   t
     t
 t
```

Agora, não é a melhor piada de toc-toc do mundo. Mas tendo isso como ponto de partida, a IA pode passar para um segundo lote de piadas de toc-toc, depois para outro. Cada vez, ela ajusta sua fórmula de piada para melhorar seus palpites.

Depois de mais algumas rodadas de adivinhação e autoajuste, ela aprendeu mais regras. Aprendeu a empregar um ponto de interrogação ocasional no final de uma frase. Está começando a aprender a usar vogais (*o* em particular). Está até tentando usar apóstrofos.

```
noo,
Lnoc noo
Kor?
hnos h nc
pt'b oa to'

asutWtnl
toy nooc
doc kKe
w ce
e
```

O quanto suas regras sobre geração de piadas de toc-toc combinam com a realidade? Ainda parece estar faltando alguma coisa.

Se ela quiser se aproximar da geração de uma piada de toc-toc aceitável, ela precisará descobrir algumas regras sobre a *ordem* das letras. Novamente, ela começa adivinhando. O palpite de que *o* é sempre seguido por *q*? Não é tão bom, ao que parece. Mas então ela palpita que *o* é frequentemente seguido por *ck*. Ouro. Ela fez algum progresso. Eis a sua ideia da piada perfeita:

```
Whock
Whock
Whock
Whock
Whock Whock Whock
Whock Whock
Whock
Whock
```

Não é exatamente uma piada de toc-toc — soa mais como um tipo de galinha. A IA precisará descobrir mais algumas regras.

Ela analisa o conjunto de dados novamente. Ela tenta novas maneiras de usar o "ock", procurando por novas combinações que correspondam melhor ao seu exemplo de piadas de toc-toc.

```
nooc hock hork
aToo tWhoc
Thock
hors Whnes Whee
noo oooc
Too?
loo
noc no'c Tno h orea
ao no oo k TeockWnork Toos
Thoe
nock T ock
Koc
hoo
KTock Toc Thock Whonk
```

Todo esse progresso acontece em apenas alguns minutos. Quando eu volto com meu café, a IA *já* descobriu que começar com "Toc, Toc/ Quem é?" se encaixa muito bem nas piadas de toc-toc existentes. Ela decide aplicar essa regra *sempre*.

Mas o resto da fórmula leva um bom tempo para ser descoberto, e a IA frequentemente se dedica a jogar um jogo de telefone consigo mesma — ou encenar uma versão robótica do esquete de humor "Quem vem primeiro?":

```
Toc, Toc
Quem é?
Iane
Aatar quem?
Aaane quem?
Aan quem?
Anac quem?
Iobe quem?
Irata quem?
```

Logo, ela descobre o restante da fórmula, mas não há, você sabe, *palavras* reais:

```
Toc, Toc
Quem é?
Bool
Hane quem?
Scheres são então lá
```

Às vezes, também fica um pouco empolgada com o comprimento do bordão.

```
Toc, Toc
Quem é?
Cheed
I? len quem?
Ita gee o lasd emotiva oud ane weonges carvalho
   anet ane
```

```
tee doolorido theis onithit onde sua thasg tate
Dente se thele ter o theung!
```

Em algum momento de seu treinamento, ela aprende a piada sobre a Vaca Sem Lábios, que se torna sua Coisa Favorita de Todos os Tempos. Se você não conhece, a piada da Vaca Sem Lábios é assim:

```
Toc, Toc.
Quem é?
Uma vaca sem lábios.
Uma vaca sem lábios quem?
Uma vaca sem lábios diz ooo ooo ooooooo.
```

Segundo a IA, isso é o ouro do humor. Assim, a IA gasta um tempo pensando que o bordão para Todas As Piadas é "ooo ooo ooooooo"…

```
Toc, Toc
Quem é?
Alec
Anic quem?
Wow co poo tho tho porta para a fouligem oo ooo
   oooo ooo oo oo oo oo ooo oo oo oo ooo ooo oooo
   ooo ooo oo oo ooo ooo ooo ooo ooo oo oo ooo ooo
   ooo ooo oo oo oo ooo ooo ooo ooo…
```

Ela descobriu uma regra, tudo bem. Eventualmente (e assumo que com relutância), ela diminui a frequência com que usa a regra da Vaca Sem Lábios para criar bordões. Mas ela ainda tem problemas que são notavelmente semelhantes ao de duas pessoas contando piadas de toc-toc por uma conexão telefônica muito ruim, talvez esquecendo quem estava realmente contando a piada:

```
Toc, Toc
Quem é?
Alther
Água quem?
Água quem quem?
Quem quem?
Arton e fora no pântano?
```

Por fim, ela descobre a estrutura de uma piada de toc-toc e começa a fazer piadas que, embora tecnicamente corretas, não fazem muito sentido. Muitas delas são parcialmente plagiadas das piadas no conjunto de dados de treinamento.

```
Toc, Toc
Quem é?
Robin
Robin quem?
Robando você! Tolo eu e estou saindo

Toc, Toc
Quem é?
Bet
Carne quem?
Carne ano sem lábios aslly.

Irlanda
Irlanda quem?
Irlanda seu dinheiro, bunda.
```

E então. Ela produziu. Uma piada de verdade. Que ela criou inteiramente por conta própria, sem plagiar o conjunto de dados, e que não era apenas inteligível, mas também realmente… engraçada?

```
Toc, Toc
Quem é?
Augusto
Augusto quem?
Augusto de piadas de toc-toc.
```

De repente, a IA começou a entender piadas de toc-toc e trocadilhos em inglês? Muito provavelmente não, dado o tamanho muito pequeno do conjunto de dados. Mas a liberdade que a IA tinha — o domínio livre de todo o conjunto de caracteres possíveis — permitiu que ela experimentasse novas combinações de sons, umas das quais acabou realmente funcionando. É mais uma vitória para a teoria do

macaco infinito* do que uma prova de conceito para o próximo clube de comédia somente para IA.

A beleza de deixar a IA fazer suas próprias regras é que uma única abordagem — aqui estão os dados; tente descobrir como copiá-los — funciona em muitos problemas diferentes. Se eu tivesse dado ao algoritmo de contar piada outro conjunto de dados em vez de piadas de toc-toc, ele aprenderia a copiar esse conjunto de dados.

Ele poderia criar novas espécies de aves:

```
Pato selvagem Yucatan
Pássaro-sol de bico-de-barco
Pica-pau de bico-ocidental
Rabo-de-cavalo de tampa-preta
João-da-palha islandês
Garça-real Robin de luto nevado
```

Ou novos perfumes:

```
Chique Dez
Eau de Boffe
Flor Frogrante
Moleja
Papai-noel para mulheres
```


* O velho ditado de que um macaco que escreve aleatoriamente em uma máquina de escrever por um tempo infinito acabará produzindo toda a obra de Shakespeare, na verdade descreve com bastante precisão o método da "força bruta" de procurar por uma solução para um problema, tentando sistematicamente tudo. Idealmente, o uso da IA para resolver o problema é uma melhoria. Idealmente.

Ou até novas receitas.

```
MOLUSCO BÁSICO CONGELADO
prato principal, sopas
½ quilo de frango
½ quilo de carne suína, cortado em cubos
½ dente de alho, esmagado
1 xícara de aipo, fatiado
1 cabeça (cerca de ½ de xícara)
6 colheres de sopa de triturador elétrico
1 colher de chá de pimenta preta
1 cebola picada
3 xícaras de caldo de carne de coruja para
   uma fruta
1 meio e meio recém triturado; equivalente
   de água
Com purê de suco de limão e fatias de limão
   em uma panela de 3 litros.
Adicione os legumes, adicione o frango ao
   molho, misturando bem na cebola. Adicione
   folha de louro, pimenta vermelha e cubra
   lentamente e cozinhe por 3 horas. Adicione
   as batatas e as cenouras ao fogo brando.
   Aqueça até o molho ferver. Sirva com
   tortas.
Se as peças licenciadas cozeram sobremesas e
   cozinhar na panela wok.
Leve à geladeira até 1⁄2 hora decorada.
Rendimento: 6 porções
```


APENAS DEIXE QUE A IA RESOLVA

Dado um conjunto de piadas de toc-toc e nenhuma instrução adicional, a IA conseguiu descobrir muitas das regras que eu teria de programar manualmente. Algumas de suas regras eu nunca teria pensado em programar ou nem saberia que existiam — como A Vaca Sem Lábios ser a melhor piada.

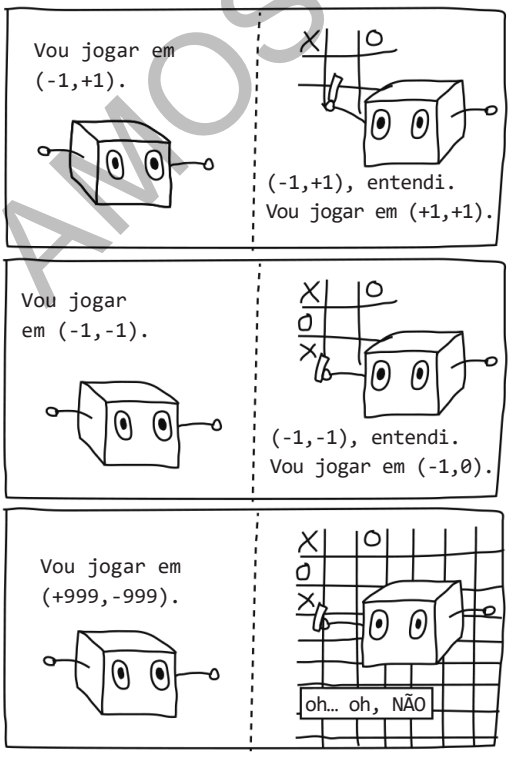
Isso faz parte do que torna as IAs atraentes para solução de problemas e é particularmente útil se as regras forem realmente complicadas ou simplesmente misteriosas. Por exemplo, a IA é frequentemente usada para reconhecimento de imagens, uma tarefa surpreendentemente complicada e difícil de executar com um programa comum de computador. Embora a maioria de nós consiga identificar facilmente um gato em uma foto, é realmente difícil definir as regras que definem um gato. Dizemos ao programa que um gato tem dois olhos, um nariz, duas orelhas e um rabo? Isso também descreve um rato e uma girafa. E se o gato estiver enrolado ou virado para o outro lado? Até escrever as regras para detectar um único olho é complicado. Mas uma IA pode olhar para dezenas de milhares de imagens de gatos e criar regras que identificam corretamente um gato na maioria das vezes.

Às vezes, a IA é apenas uma pequena parte de um programa, enquanto o restante são scripts baseados em regras. Considere um programa que permita aos clientes ligar para seus bancos para obter informações da conta. A IA de reconhecimento de voz combina sons falados com opções do menu da linha de suporte, mas as regras emitidas pelo programador controlam a lista de opções que o chamador pode acessar e o código que identifica a conta como pertencente ao cliente.

Outros programas começam com a IA, mas mudam o controle para os humanos se as coisas ficarem difíceis, uma abordagem chamada de pseudo-IA. Algumas janelas de bate-papo de atendimento ao cliente funcionam assim. Quando você inicia uma conversa com um bot, se você for muito confuso ou se a IA detectar que você está ficando irritado, de repente você poderá se ver conversando com um humano. (Um humano que, infelizmente, agora tem que lidar com um cliente confuso e/ou irritado — talvez uma opção "conversar com um humano" fosse melhor para o cliente e para o funcionário.) Os carros autônomos de hoje também funcionam dessa maneira — o motorista precisa estar sempre pronto para assumir o controle se a IA ficar perturbada.

A IA também é excelente em jogos de estratégia como o xadrez, para os quais sabemos como descrever todos os movimentos possíveis, mas não como escrever uma fórmula que nos diga qual é a melhor jogada. No xadrez, o grande número de jogadas possíveis e a complexidade do jogo significa que mesmo um grande mestre seria incapaz de criar regras sólidas e rápidas que administrem a melhor jogada em qualquer situação. Mas um algoritmo pode jogar várias partidas de treino contra si mesmo — milhões delas, mais do que o mestre mais dedicado — para criar regras que a ajudem a vencer. E como a IA aprendeu sem instruções explícitas, às vezes suas estratégias são pouco convencionais. Às vezes, um pouco não convencional *demais*.

Se você não informar à IA quais movimentos são válidos, ela poderá encontrar e explorar brechas estranhas que quebram completamente o seu jogo. Por exemplo, em 1997, alguns programadores construíram algoritmos que podiam jogar jogo da velha remotamente um contra



o outro em um tabuleiro infinitamente grande. Um programador, em vez de projetar uma estratégia baseada em regras, construiu uma IA que poderia evoluir sua própria abordagem. Surpreendentemente, a IA de repente começou a ganhar todos os seus jogos. Acontece que a estratégia da IA era colocar sua jogada muito, muito longe, de modo que quando o computador do oponente tentasse simular o novo tabuleiro, bastante expandido, o esforço faria com que ele ficasse sem memória e travasse, perdendo o jogo[1]. A maioria dos programadores de IA tem histórias como essa — momentos em que seus algoritmos os surpreenderam ao encontrar soluções que não esperavam. Às vezes, essas novas soluções são engenhosas e, às vezes, são um problema.

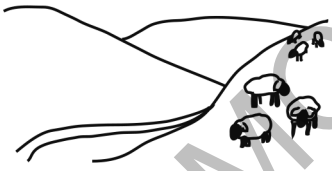
Na sua forma mais básica, tudo que a IA necessita é uma meta e um conjunto de dados a serem aprendidos e dão a largada, seja o objetivo copiar exemplos de decisões de empréstimo feitas por um ser humano, seja prever se um cliente comprará uma meia, seja maximizar a pontuação em um videogame ou maximizar a distância que um robô pode percorrer. Em todos os cenários, a IA usa tentativa e erro para inventar regras que a ajudarão a atingir seu objetivo.

ALGUMAS VEZES SUAS REGRAS SÃO RUINS

Às vezes, as brilhantes regras de solução de problemas de uma IA dependem de suposições equivocadas. Por exemplo, alguns dos meus experimentos de IA mais estranhos envolveram o produto de reconhecimento de imagem da Microsoft, que te permite enviar qualquer imagem para a IA marcar e legendar. Geralmente, esse algoritmo acerta as coisas — identificando nuvens, trens do metrô e até uma criança fazendo alguns truques de skate. Mas um dia notei algo estranho em seus resultados: estava marcando ovelhas em fotos que definitivamente não continham nenhuma ovelha. Quando investiguei mais, descobri que ela tendia a ver ovelhas em paisagens com campos verdejantes — independentemente de as ovelhas estarem lá. Por que o erro persistente e específico? Talvez, durante o treinamento,

tenham mostrado a essa IA principalmente ovelhas que estavam em campos desse tipo, e ela falhou em perceber que a legenda "ovelha" se referia aos animais, não à paisagem gramada. Em outras palavras, a IA estava olhando para a coisa errada. E, com certeza, quando mostrei exemplos de ovelhas que *não* estavam em campos verdejantes, ela tendia a ficar confusa. Se eu mostrasse fotos de ovelhas em carros, ela tenderia a rotulá-las como cães ou gatos. Ovelhas nas salas de estar também foram rotuladas como cães e gatos, assim como ovelhas seguradas nos braços das pessoas. E ovelhas em coleiras foram identificadas como cães. A IA também teve problemas semelhantes com as cabras — quando elas subiam nas árvores, como costumam fazer, o algoritmo pensava que eram girafas (e outro algoritmo semelhante as chamava de pássaros).

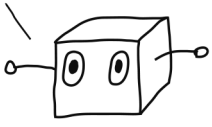
Um rebanho de ovelhas pastando em uma paisagem verdejante



Um rebanho de ovelhas pastando em uma paisagem verdejante

Embora eu não tivesse certeza, eu poderia supor que a IA tinha criado regras como grama verde = ovelha e pelos em carros ou cozinhas = gatos. Essas regras a serviram bem no treinamento, mas falharam quando encontrou o mundo real e sua variedade estonteante de situações relacionadas a ovelhas.

… pássaro peludo?



Erros de treinamento como esses são comuns nas IAs de reconhecimento de imagem. Mas as consequências desses erros podem ser graves. Uma equipe da Universidade de Stanford treinou uma IA para diferenciar imagens de peles saudáveis e imagens de câncer de pele. Depois que os pesquisadores treinaram sua IA, no entanto, descobriram que haviam inadvertidamente treinado um detector de réguas — muitos dos tumores em seus dados de treinamento foram fotografados ao lado de réguas para dar escala.[2]

COMO DETECTAR UMA REGRA RUIM

Muitas vezes, não é fácil saber quando as IAs cometem erros. Como não escrevemos suas regras, elas criam suas próprias e não as anotam ou explicam da maneira que um ser humano faria. Em vez disso, as IAs fazem ajustes interdependentes complexos em suas próprias estruturas internas, transformando uma estrutura genérica em algo aprimorado para uma tarefa individual. É como começar com uma cozinha cheia de ingredientes genéricos e terminar com biscoitos. As regras podem ser armazenadas nas conexões entre células cerebrais virtuais ou nos genes de um organismo virtual. As regras podem ser complexas, espalhadas e estranhamente entrelaçadas entre si. Estudar a estrutura interna de uma IA pode ser muito parecido com estudar o cérebro ou um ecossistema — e você não precisa ser um neurocientista ou um ecologista para saber o quão complexas essas coisas podem ser.

Pesquisadores estão trabalhando para descobrir como as IAs tomam decisões, mas, em geral, é difícil descobrir quais são as regras internas de uma IA. Frequentemente, é apenas porque as regras são difíceis de entender, mas em outros momentos, principalmente quando se trata de algoritmos comerciais e/ou governamentais, é porque o próprio algoritmo é patenteado. Infelizmente, os problemas geralmente aparecem nos resultados do algoritmo

quando ele já está em uso, às vezes tomando decisões que podem afetar vidas e potencialmente causar danos reais.

Por exemplo, uma IA que estava sendo usada para recomendar quais prisioneiros teriam liberdade condicional foi pega tomando decisões preconceituosas, copiando sem saber os comportamentos racistas que encontrou em seu treinamento.[3] Mesmo sem entender o que é preconceito, a IA ainda pode ser tendenciosa. Afinal, muitas IAs aprendem copiando seres humanos. A pergunta que elas estão respondendo não é "Qual é a melhor solução?", mas "O que os humanos teriam feito?".

Testar sistematicamente em busca de preconceitos pode ajudar a detectar alguns desses problemas comuns antes que eles causem danos. Mas outra peça do quebra-cabeça é aprender a antecipar problemas antes que eles ocorram e projetar IAs para evitá-los.

QUATRO SINAIS DE CONDENAÇÃO DA IA

Quando as pessoas pensam no desastre da IA, elas pensam em IAs recusando ordens, decidindo que seus maiores interesses estão em matar todos os seres humanos ou criar bots exterminadores. Mas todos esses cenários de desastre assumem um nível de pensamento crítico e uma compreensão humana do mundo das quais as IAs não serão capazes no futuro próximo. Como afirmou o pesquisador líder em aprendizado de máquina, Andrew Ng, preocupar-se com uma tomada de poder da IA é como se preocupar com a superlotação em Marte.[4]

Isso não quer dizer que as IAs de hoje não possam causar problemas. De irritar levemente seus programadores a perpetuar preconceitos ou bater um carro autônomo, as IAs de hoje não são exatamente inofensivas. Mas, sabendo um pouco sobre a IA, podemos prever alguns desses problemas.