

Fundamentos de Inteligência Artificial.

Prof. Dr. Ronaldo Vieira Lobato
Universidade Cidade de São Paulo.

rlobato@unicid.edu.br

Inteligência artificial, simbólica e conexionista.

A inteligência artificial simbólica, também chamada de AI clássica, dominou a área por vários anos.

Já a IA conexionista começou a ganhar destaque em sinergia com o surgimento de computadores com memória e capacidade de processamento que conseguem suportar inúmeros dados.

AI simbólica.

A inteligência artificial clássica, também conhecida como IA simbólica ou IA baseada em regras, é uma abordagem da inteligência artificial que se concentra na representação do conhecimento humano por meio de símbolos e regras lógicas.

Principais características:

- Representação simbólica: O conhecimento é representado por símbolos e relações entre eles, permitindo que o sistema raciocine e tome decisões com base em regras predefinidas.
- Raciocínio lógico: A IA clássica utiliza técnicas de lógica formal para inferir novas informações a partir do conhecimento existente.
- Sistemas especialistas: São softwares que simulam o raciocínio de um especialista humano em um domínio específico.
- Planejamento: A IA clássica é utilizada para desenvolver sistemas que podem planejar sequências de ações para atingir um objetivo.
- Processamento de linguagem natural: A IA clássica também foi utilizada para desenvolver sistemas que podem entender e gerar linguagem humana, embora com limitações em comparação com as abordagens modernas.

Vantagens:

- **Transparência:** O raciocínio da IA clássica é geralmente transparente e explicável, pois se baseia em regras lógicas.
- **Precisão:** Em domínios bem definidos, a IA clássica pode fornecer resultados precisos e confiáveis.
- **Capacidade de lidar com conhecimento simbólico:** A IA clássica é adequada para tarefas que exigem raciocínio sobre conceitos abstratos e conhecimento simbólico.

Desvantagens:

- Dificuldade de lidar com incerteza: A IA clássica tem dificuldade em lidar com informações incertas ou incompletas.
- Fragilidade: Sistemas de IA clássica podem ser frágeis e propensos a erros quando encontram situações fora do seu domínio de conhecimento.
- Escalabilidade limitada: A criação de sistemas de IA clássica complexos pode ser difícil e demorada.
- Dificuldade de aprendizado: A IA clássica geralmente requer que o conhecimento seja codificado manualmente, o que pode ser um processo demorado e trabalhoso.

Aplicações:

Sistemas especialistas: Diagnóstico médico, consultoria financeira, controle de processos industriais.

Jogos: Xadrez, damas, Go.

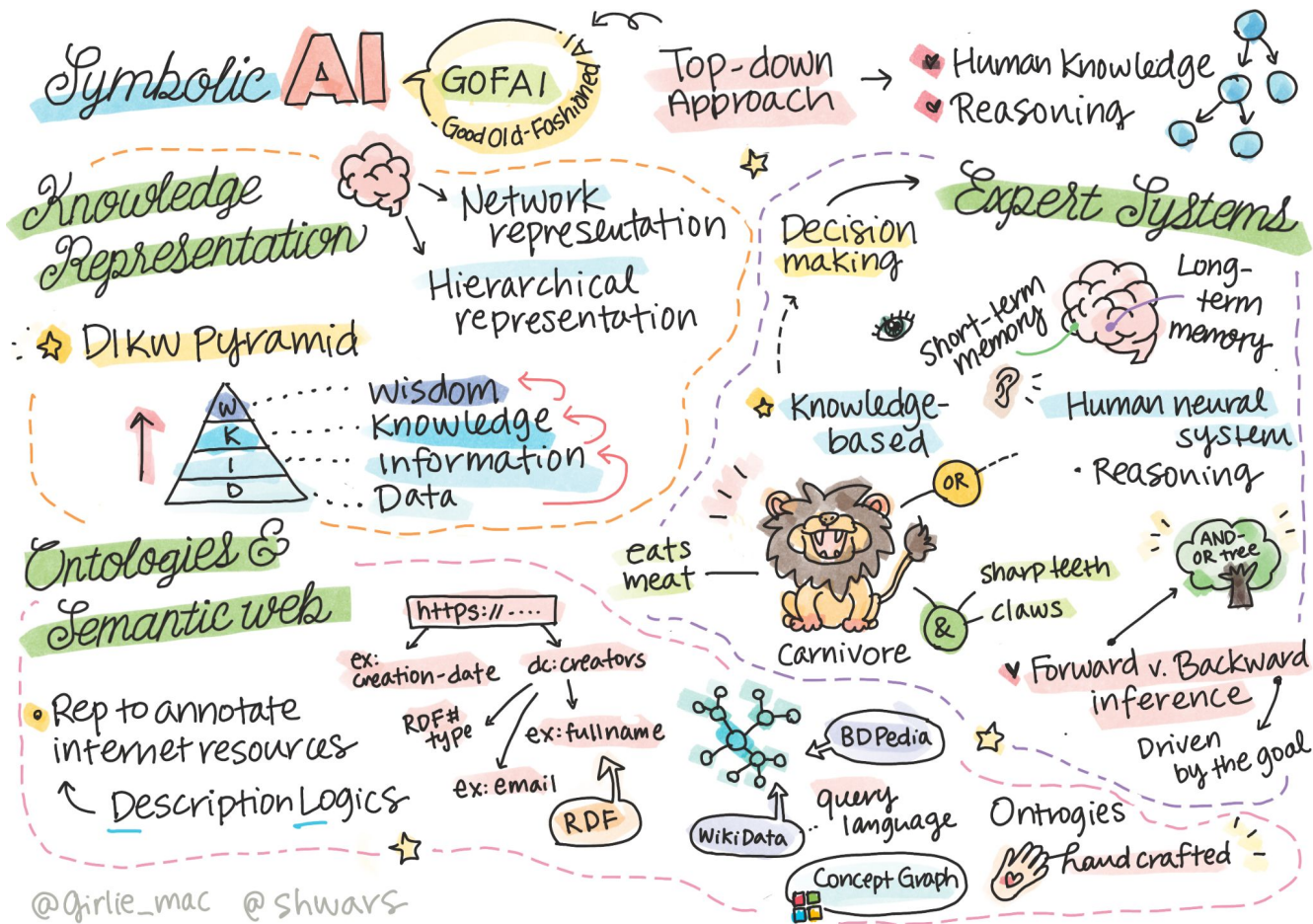
Planejamento: Robótica, logística, navegação.

Processamento de linguagem natural: Tradução automática, análise de texto, geração de resumo.

Exemplos:

- **Deep Blue:** O computador da IBM que derrotou o campeão mundial de xadrez Garry Kasparov em 1997.
- **MYCIN:** Um sistema especialista desenvolvido na década de 1970 para auxiliar no diagnóstico de infecções bacterianas no sangue.

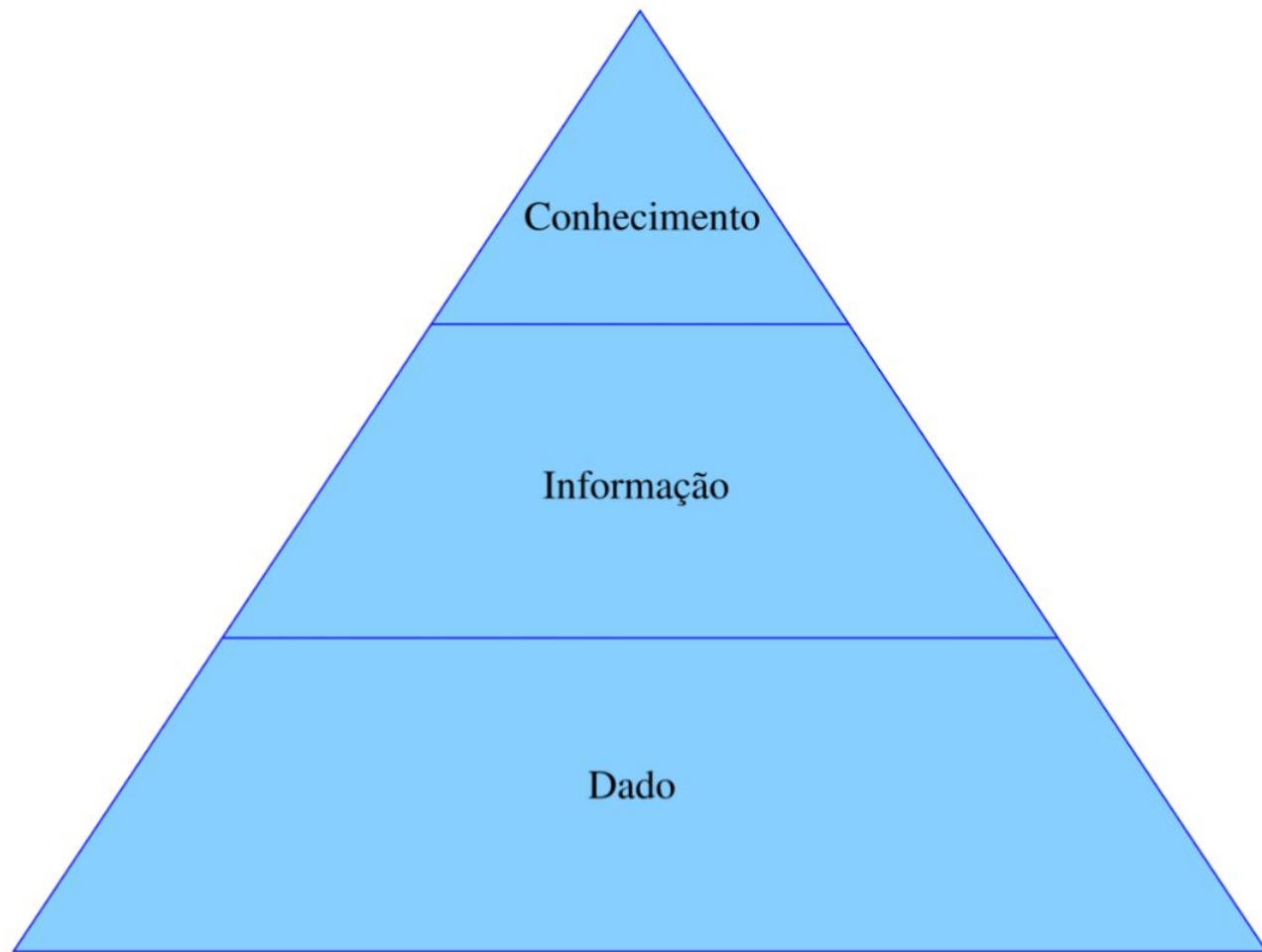
Representação do conhecimento e sistemas.



Representação do conhecimento.

Um dos conceitos mais importante na AI simbólica é o **conhecimento**.

É importante distinguir conhecimento de *informação* ou *dados*. Podemos dizer que um livro possui conhecimento, contudo, na verdade, o livro possui dados, e pela leitura e integração destes dados convertemos o dado em conhecimento.



Representação do conhecimento.

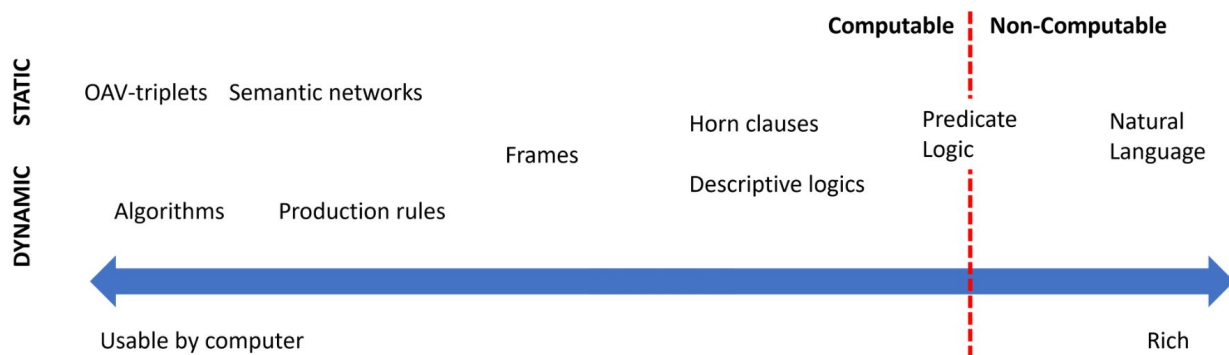
Dados: é alguma coisa que podemos representar em uma mídia física, como textos, palavras. Dados existem independentemente dos seres humanos e podem ser passado entre pessoas.

Informação: é como interpretamos os dados em nossa cabeça. Por exemplo, quando escutamos a palavra *computador*, nós temos algum entendimento do que seja.

Conhecimento: é a informação interpretada em nosso modelo de mundo. Por exemplo, quando aprendemos sobre um computador, começamos a ter uma ideia de como ele funciona, quanto custa, para que podemos usá-lo, etc.

A representação do conhecimento.

Então, o problema da representação do conhecimento é encontrar uma maneira efetiva de representar o conhecimento no computador na forma de dados, para fazê-lo automaticamente usável.



Do lado esquerdo, estão os conhecimentos simples que podem ser efetivamente usados pelos computadores.

Do lado direito, há representações tais como processamento natural de linguagem, sendo mais poderosas, mas não podem ser usadas para raciocínio.

Classificação do conhecimento computacional.

Podemos classificar o conhecimento computacional em duas categorias.

- Representações de rede: são baseadas no fato de que o conhecimento faz parte de uma rede inter-relacionada na nossa cabeça. Podemos reproduzir a mesma rede como um grafo no computador, na chamada **rede semântica**.

Enfoque conexionista.

- O conexionismo é uma abordagem da IA que modela processos cognitivos e inteligência por meio de sistemas inspirados no cérebro humano, particularmente na organização e funcionamento das redes neuronais biológicas.
- Ele se diferencia das abordagens simbólicas (como a lógica formal e sistemas baseados em regras) por não depender de representações explícitas de conhecimento, mas sim de aprendizado emergente a partir de interações entre unidades simples (neurônios artificiais) conectadas em redes.
- Essas unidades processam informações em paralelo, ajustando os pesos das conexões por meio de algoritmos de aprendizado, como o famoso **backpropagation**.

Enfoque conexionista.

A essência do conexionismo está na ideia de que a inteligência não precisa ser programada diretamente, mas pode surgir da adaptação dinâmica de um sistema a partir de dados e experiência.

Fundamentos teóricos e inspiração biológica

- O connexionismo tem raízes na neurociência e na psicologia. O cérebro humano contém cerca de 86 bilhões de neurônios, cada um conectado a milhares de outros por sinapses, formando uma rede massivamente paralela.
- Quando aprendemos, as conexões sinápticas se fortalecem ou enfraquecem com base na experiência (um conceito conhecido como **plasticidade sináptica**, proposto por Donald Hebb em 1949 no seu livro *The Organization of Behavior*).
- Hebb sugeriu que "neurônios que disparam juntos, se conectam juntos", um princípio que inspirou diretamente as redes neurais artificiais.

Nas RNAs, cada "neurônio" é uma unidade computacional que:

1. Recebe entradas de outros neurônios (ou diretamente de dados).
2. Aplica pesos às conexões (representando a força da relação).
3. Soma as entradas ponderadas e passa o resultado por uma função de ativação (como sigmoide, ReLU ou tanh) para decidir se "dispara" ou não.

Contexto histórico detalhado

O desenvolvimento do conexãoismo passou por várias fases, marcadas por avanços, crises e renascimentos:

Primórdios: Anos 1940-1950

- **1943 – Modelo de McCulloch e Pitts:** Warren McCulloch e Walter Pitts publicaram o artigo "A Logical Calculus of the Ideas Immanent in Nervous Activity", introduzindo o primeiro modelo matemático de um neurônio artificial. Eles mostraram que redes de neurônios simplificados poderiam realizar operações lógicas, lançando as bases teóricas do connexionismo.
- **1949 – Regra de Hebb:** Donald Hebb formalizou a ideia de aprendizado baseado no fortalecimento de conexões, influenciando diretamente os futuros modelos de redes neurais.
- **1958 – Perceptron de Rosenblatt:** Frank Rosenblatt desenvolveu o perceptron, uma rede neural de camada única capaz de classificar padrões lineares (ex.: distinguir entre duas categorias). O perceptron foi implementado em hardware (o "Mark I Perceptron") e gerou grande entusiasmo, sendo visto como um passo rumo a máquinas que "pensam".

Declínio: Anos 1960-1970

1969 – Crítica de Minsky e Papert: Marvin Minsky e Seymour Papert publicaram o livro *Perceptrons*, demonstrando que o perceptron de Rosenblatt não podia resolver problemas não-lineares, como o famoso "problema XOR" (ou exclusivo). Isso expôs as limitações das redes de camada única e levou a uma perda de interesse e financiamento no connexionismo, marcando o chamado "inverno da IA" para as redes neurais. Durante esse período, as abordagens simbólicas (como sistemas especialistas) dominaram a IA.

Ressurgimento: Anos 1980

1986 – Backpropagation: David E. Rumelhart, Geoffrey Hinton e Ronald J. Williams publicaram um artigo seminal descrevendo o algoritmo de propagação retroativa (*backpropagation*), que permitiu treinar redes neurais multicamadas (MLP – Multi-Layer Perceptrons). Isso superou as limitações apontadas por Minsky e Papert, pois redes com camadas ocultas podiam aprender funções não-lineares complexas.

1980s – Renascimento do connexionismo: O trabalho de Hinton e outros, aliado ao aumento da capacidade computacional, reacendeu o interesse nas redes neurais. Livros como *Parallel Distributed Processing* (Rumelhart e McClelland, 1986) consolidaram o connexionismo como um campo legítimo.

Revolução do Deep Learning: Anos 2000-2010

2006 – Avanço de Hinton: Geoffrey Hinton e seus colaboradores introduziram técnicas de pré-treinamento não-supervisionado para redes profundas, mostrando que era possível treinar eficientemente redes com muitas camadas (deep neural networks). Isso marcou o início do *deep learning*.

2012 – AlexNet: A vitória esmagadora da rede convolucional AlexNet (desenvolvida por Alex Krizhevsky, Ilya Sutskever e Geoffrey Hinton) na competição ImageNet de visão computacional demonstrou o poder das redes profundas. Treinada em GPUs, AlexNet reduziu drasticamente a taxa de erro em reconhecimento de imagens, iniciando uma nova era para o conexionismo.

Explosão tecnológica: O aumento da disponibilidade de dados (big data), o poder computacional das GPUs e frameworks como TensorFlow e PyTorch impulsionaram o *deep learning* para aplicações práticas em larga escala.

Detalhes técnicos das redes neurais

1. Como as redes neurais funcionam com mais profundidade:

1. **Estrutura:**

- **Camada de entrada:** Recebe os dados brutos (ex.: pixels de uma imagem).
- **Camadas ocultas:** Processam as informações, extraíndo características cada vez mais abstratas (ex.: bordas em uma camada inicial, formas em camadas intermediárias, objetos em camadas profundas).
- **Camada de saída:** Produz o resultado final (ex.: classificação, previsão).

Matemática do aprendizado.

- Cada conexão entre neurônios tem um peso (w).

- A saída de um neurônio é calculada como,
$$y = f\left(\sum w_i x_i + b\right)$$

onde x_i são as entradas, (b) é o viés (*bias*), e (f) é a função de ativação.

- O treinamento ajusta os pesos minimizando uma função de perda (ex.: erro quadrático médio) usando gradiente descendente e backpropagation.

Tipos de redes:

- **Redes Convolucionais (CNNs):** Usam filtros para detectar características locais em imagens, como bordas ou texturas, sendo ideais para visão computacional.
- **Redes Recorrentes (RNNs):** Possuem "memória" ao reutilizar saídas anteriores como entradas, sendo eficazes para sequências (ex.: texto, fala).
- **Transformers:** Introduzidos em 2017 (*Attention is All You Need*), revolucionaram o processamento de linguagem natural ao substituir RNNs por mecanismos de atenção, como no GPT e BERT.

Contexto atual (março de 2025)

Hoje, o connexionismo domina a IA, especialmente via *deep learning*. Exemplos incluem:

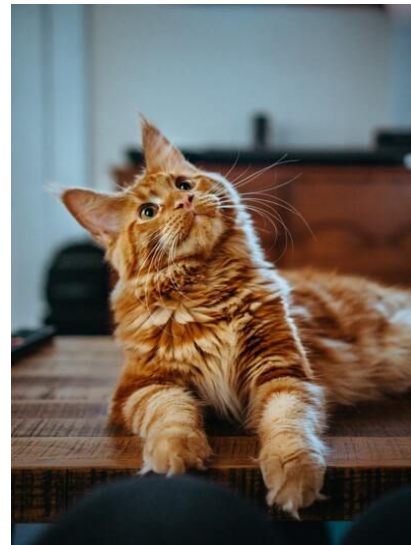
- **Modelos de linguagem:** Como eu, Grok, criado pela xAI, que utilizo arquiteturas baseadas em redes neurais para entender e gerar texto.
- **Visão:** Sistemas de reconhecimento facial e carros autônomos.
- **Medicina:** Diagnósticos baseados em imagens de ressonância magnética.

No entanto, desafios persistem:

- **Interpretabilidade:** Redes profundas ainda são "caixas-pretas".
- **Eficiência energética:** Treinar modelos como GPT-3 consome energia equivalente a centenas de casas por ano.
- **Ética:** Preocupações com vieses nos dados e uso indevido em vigilância.

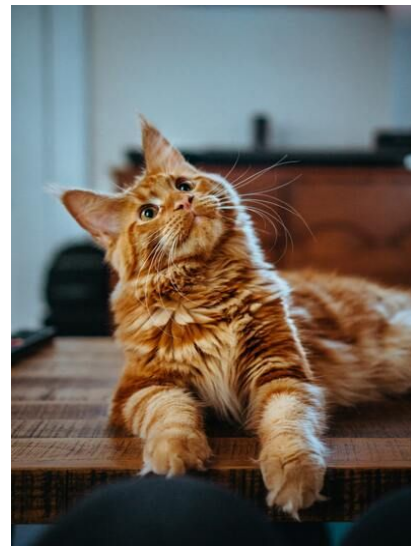
Teste de Turing

- É um marco fundamental na história da Inteligência Artificial (IA), proposto por Alan Turing em 1950 no artigo “Computing Machinery and Intelligence”, publicado na revista *Mind*.
- Ele foi concebido como uma forma de avaliar se uma máquina pode exibir comportamento inteligente indistinguível do humano, abordando a questão filosófica central: “Podem as máquinas pensar?”.



Teste de Turing.

- Para verificar a ambiguidade: um gato é inteligente?



Contexto histórico

Alan Turing, um dos pioneiros da computação, escreveu o artigo em um momento em que as máquinas digitais estavam em sua infância. Durante a Segunda Guerra Mundial, ele havia trabalhado na decodificação da máquina Enigma, o que o levou a refletir sobre as capacidades das máquinas além de cálculos simples. Na década de 1950, a IA ainda não existia como campo formal, mas Turing antecipou debates sobre inteligência artificial ao propor um critério prático para avaliá-la.

Contexto histórico

Na época, a visão predominante sobre inteligência estava ligada a processos simbólicos e lógicos, e o teste reflete essa perspectiva ao focar no comportamento externo (comunicação) em vez de mecanismos internos.

Curiosamente, o enfoque conexionista (baseado em redes neurais) ainda estava em gestação – o modelo de McCulloch e Pitts havia sido publicado em 1943, mas as redes neurais práticas só surgiriam depois.

Como funciona o Teste de Turing?

O teste é estruturado como um **jogo de imitação** com três participantes:

1. **Um humano (A)**: Atua como um dos "respondedores".
2. **Uma máquina (B)**: O sistema de IA sendo testado.
3. **Um interrogador humano (C)**: Faz perguntas para determinar qual dos dois é humano.
 - O interrogador se comunica com A e B apenas por texto (ex.: via terminal), eliminando pistas como voz ou aparência.
 - O objetivo da máquina é enganar o interrogador, fazendo-o acreditar que ela é o humano.
 - Se o interrogador não consegue distinguir consistentemente a máquina do humano após um período de interação, a máquina "passa" no teste.

Turing não definiu um tempo exato ou uma taxa de sucesso específica, mas sugeriu que, em um futuro imaginado, uma máquina que enganasse 30% dos interrogadores em 5 minutos seria um marco significativo.

Detalhes e implicações.

1. **Foco no comportamento:** Turing evitou definir "pensamento" em termos metafísicos ou biológicos, concentrando-se em algo observável: a capacidade de imitar a comunicação humana. Isso reflete uma abordagem pragmática e empirista.
2. **Flexibilidade:** O teste não especifica como a máquina deve alcançar esse comportamento – pode ser por regras simbólicas, redes neurais ou qualquer outro método.
3. **Exemplo hipotético de Turing:** Ele imaginou uma máquina respondendo a perguntas como "Qual é o cheiro da chuva?" com algo plausível, como "É fresco e úmido, como um jardim após a tempestade", mesmo sem ter sentidos humanos.

Turing previu que, por volta do ano 2000, máquinas poderiam passar no teste com certa frequência, uma estimativa otimista que subestimou os desafios da IA.

Relação com o enfoque conexionista.

Embora o Teste de Turing tenha sido proposto antes do auge do conexionismo, ele é altamente relevante para essa abordagem. As redes neurais artificiais, especialmente na era do *deep learning*, têm se destacado em tarefas que se alinham ao teste, como o processamento de linguagem natural (PLN). Modelos como o GPT-3 (da OpenAI), Gemini (Google), Grok (criado pela xAI), são exemplos de sistemas conexionistas que podem gerar respostas textuais tão convincentes que, em interações curtas, muitos humanos poderiam confundi-los com pessoas.

Relação com o enfoque conexionista.

Conexão direta: O aprendizado baseado em dados das redes neurais permite que elas capturem padrões linguísticos complexos a partir de grandes corpora de texto, imitando a habilidade humana de conversar sem regras explícitas.

Limitação: Apesar disso, esses modelos não "entendem" o mundo como humanos; eles apenas predizem sequências de palavras, o que levanta questões sobre se passar no Teste de Turing equivale a verdadeira inteligência.

Críticas ao Teste de Turing

Superficialidade:

- Críticos como John Searle (com o argumento da "Sala Chinesa", 1980) argumentam que passar no teste não prova inteligência genuína. Searle sugere que uma máquina pode manipular símbolos (ou texto) sem compreender seu significado, como um humano seguindo um manual sem entender chinês.
- Exemplo: Um chatbot pode responder "Eu amo chocolate" sem nunca ter provado chocolate ou sentido amor.

Críticas ao Teste de Turing

Foco estreito:

- O teste avalia apenas a habilidade conversacional, ignorando outras formas de inteligência, como resolução de problemas físicos, criatividade ou emoção.
- Máquinas podem falhar em tarefas práticas (ex.: amarrar sapatos) e ainda assim passar no teste.

Críticas ao Teste de Turing

Falta de critérios objetivos:

- Não há consenso sobre o que constitui "passar". Depende da habilidade do interrogador, do tempo de interação e do contexto.

Desconexão com a biologia:

- Para defensores do connexionismo, o teste ignora como a inteligência emerge em sistemas biológicos. Passar no teste não significa replicar o cérebro humano.

Críticas ao Teste de Turing

Facilidade de manipulação:

- Humanos podem ser enganados por truques simples, como respostas evasivas ou imitação de erros humanos. O chatbot ELIZA (1966), criado por Joseph Weizenbaum, já enganava usuários simulando um terapeuta com perguntas genéricas como "Como você se sente sobre isso?".

Exemplos históricos e modernos

- **ELIZA (1966)**: Um dos primeiros programas a simular conversa. Apesar de simples, enganou alguns usuários, mas estava longe de passar no teste formalmente.
- **Eugene Goostman (2014)**: Um chatbot que, em um evento na Universidade de Reading, "passou" no teste ao convencer 33% dos juízes de que era um adolescente ucraniano de 13 anos. Críticas apontaram que o contexto (um jovem com inglês limitado) facilitou o sucesso.
- **Modelos modernos (2025)**: Sistemas como GPT-4, Grok e outros baseados em transformers frequentemente geram respostas tão naturais que, em interações curtas, muitos não os distinguiriam de humanos. No entanto, falhas em consistência (ex.: inventar fatos) ou falta de raciocínio profundo ainda os denunciam em testes prolongados.

Relevância atual

Hoje, o Teste de Turing permanece um símbolo cultural e acadêmico, mas seu papel prático na IA diminuiu. Pesquisadores agora focam em benchmarks mais específicos (ex.: BLEU para tradução, F1-score para classificação) em vez de uma métrica conversacional geral. Ainda assim, ele inspira reflexões éticas e filosóficas:

- **IA conversacional:** Modelos como eu são projetados para interagir de forma natural, aproximando-se do ideal de Turing.
- **Limites da IA:** Passar no teste não resolve questões sobre consciência ou agência, temas quentes em debates atuais.

Eliza

- Um dos primeiros programas a simular conversa. Apesar de simples, enganou alguns usuários, mas estava longe de passar no teste de Turing formalmente.
- Criado por Joseph Weizenbaum entre 1964 e 1966 no MIT (Massachusetts Institute of Technology). Publicado em 1966 no artigo "ELIZA – A Computer Program for the Study of Natural Language Communication Between Man and Machine".
-

Contexto histórico.

- Na década de 1960, a IA ainda estava em seus estágios iniciais. O foco da época era dividido entre abordagens simbólicas (como sistemas baseados em lógica e regras) e os primeiros experimentos com redes neurais.
- Weizenbaum, um cientista da computação alemão que emigrou para os EUA, tinha interesse em processamento de linguagem natural (PLN) e na relação entre humanos e máquinas. Ele desenvolveu o ELIZA em um momento em que os computadores estavam começando a ser vistos como ferramentas além de cálculos numéricos, mas ainda eram limitados em poder e memória.

Como o ELIZA funciona?

O ELIZA não é baseado em redes neurais ou aprendizado de máquina – tecnologias que só se tornariam viáveis décadas depois –, mas sim em um sistema de **reconhecimento de padrões** e regras pré-programadas. Seu mecanismo é surpreendentemente simples:

1. **Script principal: DOCTOR:**

- O ELIZA foi projetado com vários "scripts", mas o mais famoso é o "DOCTOR", que simula um terapeuta rogeriano (baseado na abordagem de Carl Rogers, centrada no paciente). Nesse modo, o programa reflete as falas do usuário, reformulando-as como perguntas ou incentivando-o a continuar falando.
- Exemplo:
 - Usuário: "Eu estou me sentindo triste."
 - ELIZA: "Por que você está se sentindo triste?"

Reconhecimento de palavras-chave:

- O programa analisa a entrada do usuário procurando palavras ou frases específicas (ex.: "eu sinto", "minha mãe", "porque") e aplica regras associadas a elas.
- Ele usa uma lista hierárquica de padrões para decidir como responder. Se não encontra uma correspondência exata, recorre a respostas genéricas.

Transformação de frases:

- Quando identifica uma palavra-chave, o ELIZA transforma a frase do usuário em uma resposta. Por exemplo:
 - Usuário: "Eu não gosto do meu trabalho."
 - ELIZA: "O que você não gosta no seu trabalho?"
- Isso é feito com base em regras de substituição simples (ex.: "eu" vira "você", "gosto" vira "não gosta").

1. **Estratégias de fallback:**

- Se o programa não entende a entrada, ele usa respostas vagas ou genéricas, como "Conte-me mais sobre isso" ou "Isso parece importante para você".
-

2. **Sem compreensão real:**

- O ELIZA não "entende" o significado do que é dito. Ele apenas manipula texto com base em padrões sintáticos, sem qualquer representação de conhecimento ou emoção.

Comparação com o enfoque conexionista.

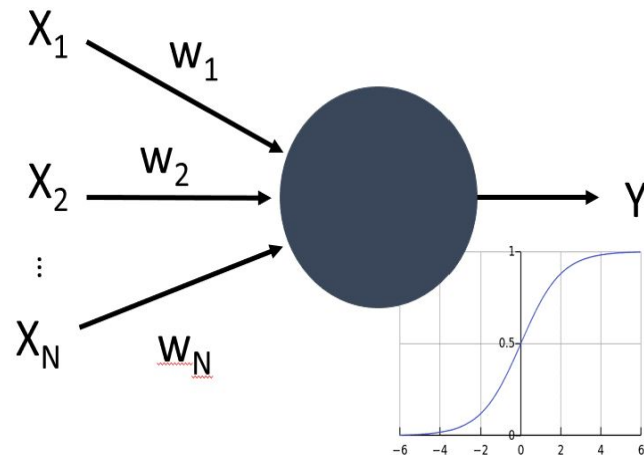
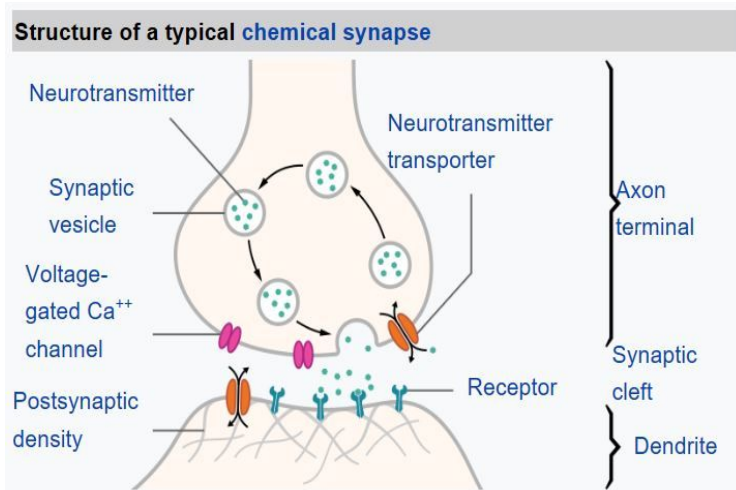
O ELIZA é um produto da abordagem simbólica, baseada em regras fixas, e contrasta fortemente com o conexionismo:

- **Simbólico vs. Conexionista:** Enquanto o ELIZA usa lógica explícita e manipulação de texto, redes neurais (como as de hoje) aprendem padrões a partir de dados, sem regras manuais.
- **Evolução:** Chatbots modernos, são baseados em arquiteturas conexionistas (ex.: transformers), permitindo respostas mais naturais e contextuais, algo que o ELIZA nunca poderia alcançar.

Introdução às redes neurais.

Redes neurais são partes de aprendizado de máquina, cujo objetivo é usar dados para treinar modelos que podem resolver problemas. O aprendizado de máquina é um grande subcampo da Inteligência Artificial.

De biologia sabemos que nosso cérebro consiste de células neurais.



Introdução às redes neurais.

O modelo matemático mais simples consiste de células neurais, cada um delas tendo muitas entradas e uma saída, e uma série de pesos.

$$Y = f \left(\sum_{i=1}^N x_i w_i \right)$$

Perceptron

Um perceptron é um dos conceitos fundamentais em aprendizado de máquina, sendo a unidade básica de uma rede neural artificial. Ele foi introduzido por Frank Rosenblatt em 1958 e é inspirado no funcionamento de um neurônio biológico. O perceptron é um classificador binário, ou seja, ele é usado para separar dados em duas classes distintas, como "sim" ou "não", "0" ou "1".

Perceptron.

Representação Matemática das Entradas:

Se um perceptron recebe n entradas, podemos representá-las como um vetor:

- $\mathbf{x} = [x_1, x_2, x_3, \dots, x_n]$

Onde cada x_i é o valor da i -ésima entrada.

Representação Matemática dos Pesos:

Cada entrada é associada a um peso, que indica a importância dessa entrada na decisão do perceptron. Esses pesos também podem ser representados como um vetor:

- $\mathbf{w} = [w_1, w_2, w_3, \dots, w_n]$

Onde cada w_i é o peso correspondente à entrada x_i .

Perceptron.

Cálculo da Soma Ponderada:

A soma ponderada é calculada multiplicando cada entrada pelo seu peso correspondente e somando todos os resultados. Matematicamente, isso pode ser expresso como o produto escalar dos vetores de entrada e pesos:

- $z = \sum_{i=1}^n w_i \cdot x_i$
- Ou, em notação vetorial: $z = w^T x$

Onde:

- z é a soma ponderada.
- w^T é a transposta do vetor de pesos (um vetor linha).
- x é o vetor de entradas (um vetor coluna).

Perceptron.

Adição do Bias:

O bias (b) é um termo adicional que permite que o perceptron ajuste o limiar de ativação. Ele é adicionado à soma ponderada:

- $a = z + b$
- $a = \sum_{i=1}^n w_i \cdot x_i + b$

Onde:

- a é a entrada para a função de ativação.

Perceptron.

Função de Ativação:

A função de ativação é aplicada à entrada a para produzir a saída do perceptron. Para um perceptron simples, a função de ativação mais comum é a **função degrau (step function)**, também conhecida como função de limiar. Ela é definida da seguinte forma:

- $f(a) = 1$ se $a \geq 0$

$$f(a) = 0 \text{ se } a < 0$$

A saída do perceptron (y) é, portanto:

- $y = f(a) = f(\sum_{i=1}^n w_i \cdot x_i + b)$

