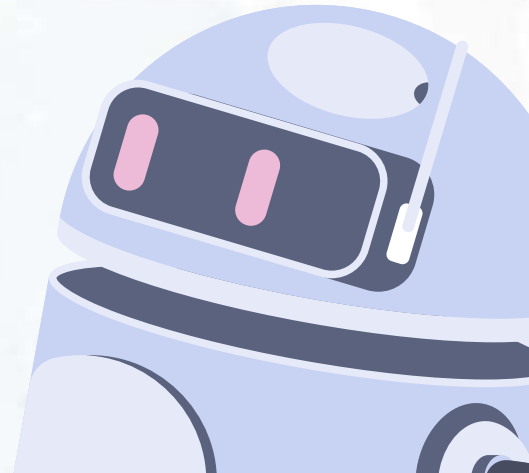


Explorando o Poder das Redes Neurais: Uma Jornada Pela Inteligência Artificial





- Formado em licenciatura em matemática na UEPA (2018)
- Curso de ciência de dados pela Blueedtech (2022-2023)
- Atualmente fazendo pós em ciência de dados e inteligência artificial pela Uninter
- Lex Luthor dos dados



The diagram consists of three nested ellipses. The outermost ellipse is dark blue and contains the text for 'ARTIFICIAL INTELLIGENCE'. Inside it is a medium blue ellipse containing the text for 'MACHINE LEARNING'. The innermost ellipse is light blue and contains the text for 'DEEP LEARNING'. This visualizes that Deep Learning is a subset of Machine Learning, which is a subset of Artificial Intelligence.

ARTIFICIAL INTELLIGENCE

Qualquer técnica que capacite uma máquina a imitar a inteligência humana

MACHINE LEARNING

Métodos estatísticos que possibilitam que as máquinas aprendam a partir dos dados sem programação

DEEP LEARNING

Redes Neurais com múltiplas camadas que assimilam tarefas e reconhecem símbolos a partir dos dados

Índice

01 → O que é?

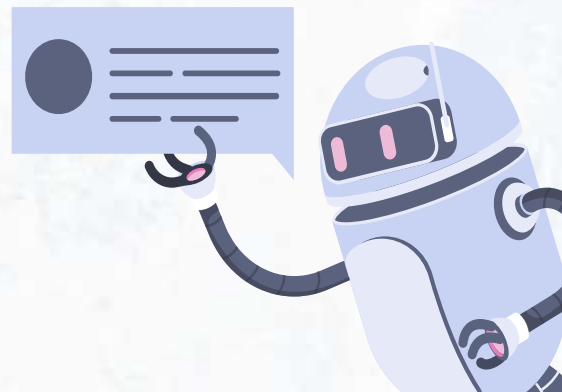
02 → Breve História

03 → Estrutura

04 → Arquitetura

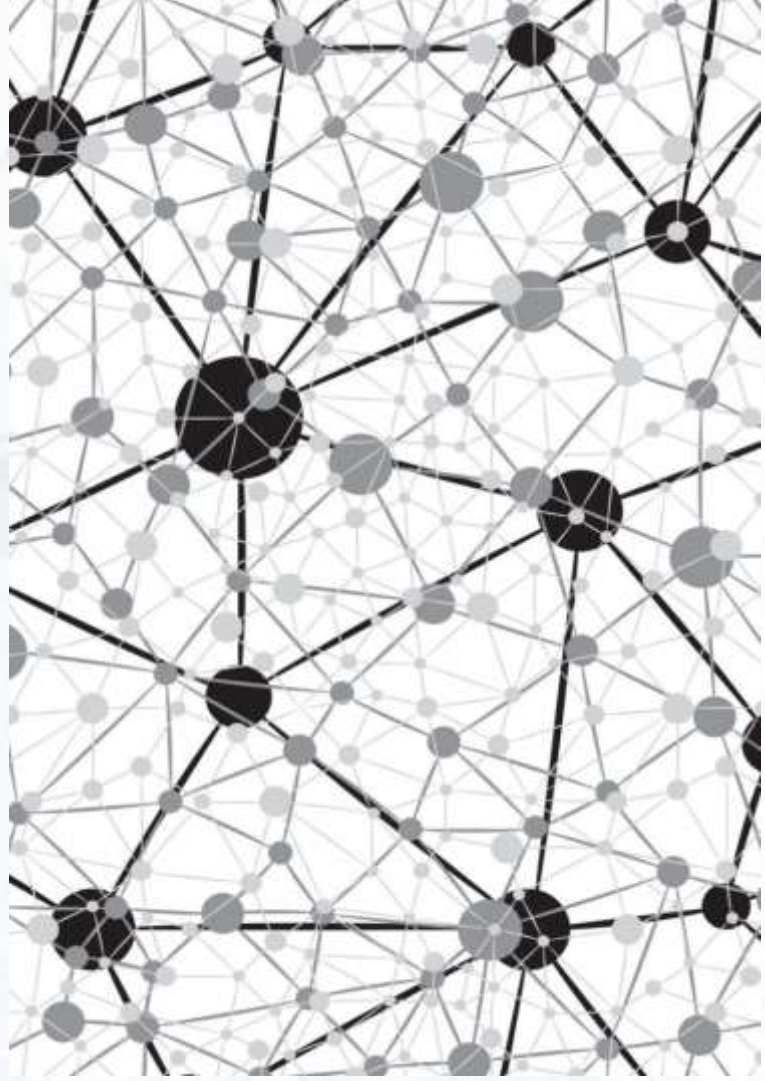
05 → Treinamento

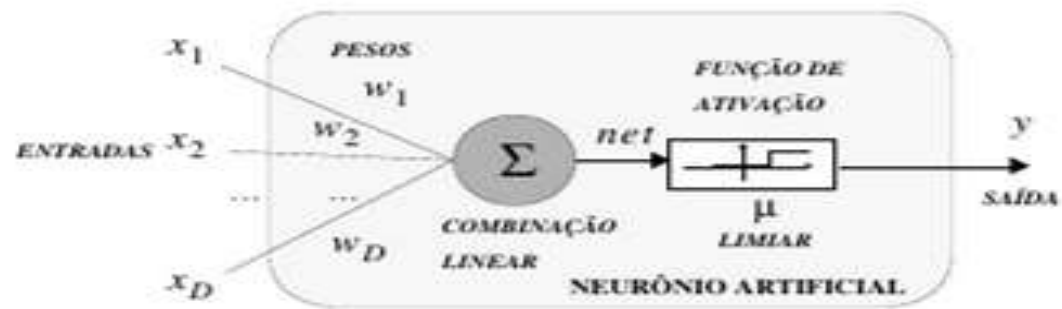
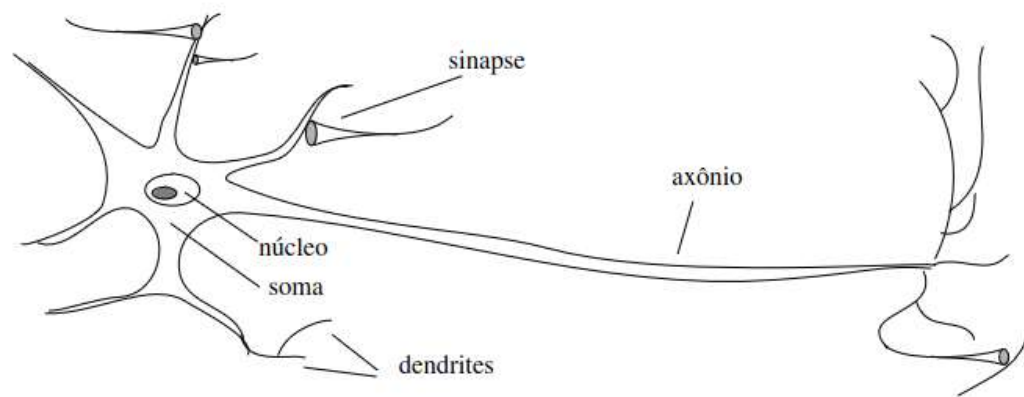
06 → Mãos ao código



01 → O que é?

- Sendo uma sub-área da *Inteligência Artificial*, são sistemas computacionais construídos através de técnicas que procuram imitar o cérebro humano com suas conexões de elementos de processamento.
- Uma rede neural artificial pretende simular o sistema nervoso humano, o qual contém uma coleção de neurônios (os elementos de processamento) que se comunicam através de axônios e dendritos







Reconhecimento de Imagens
(Visão Computacional)

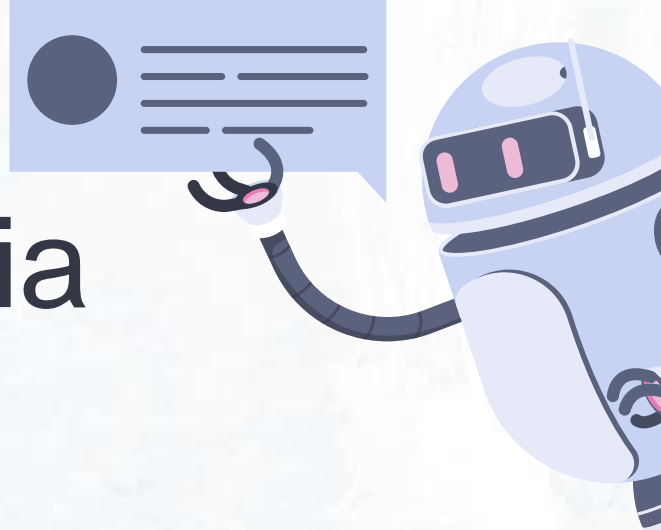


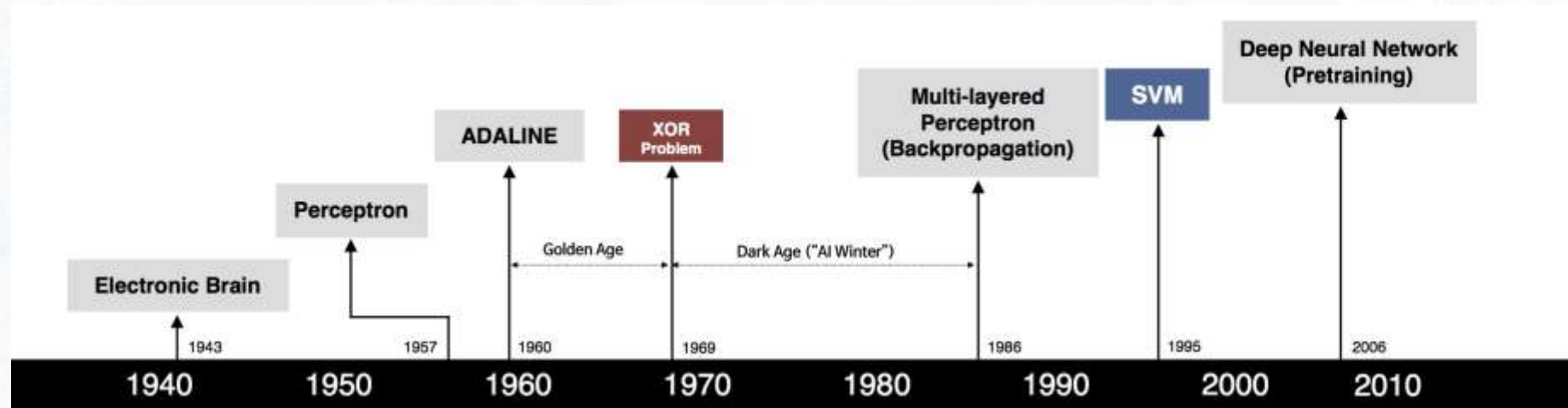
NLP (ChatGPT)



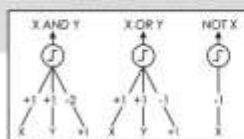
Reconhecimento
por Voz

02 → Breve História





S. McCulloch - W. Pitts



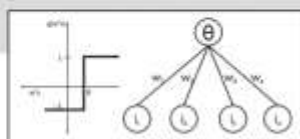
- Adjustable Weights
- Weights are not Learned



F. Rosenblatt



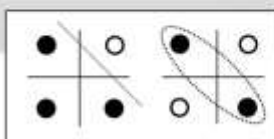
B. Widrow - M. Hoff



- Learnable Weights and Threshold



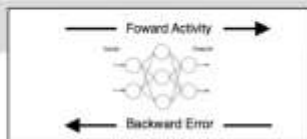
M. Minsky - S. Papert



- XOR Problem



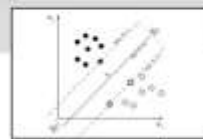
D. Rumelhart - G. Hinton - R. Williams



- Solution to nonlinearly separable problems
- Big computation, local optima and overfitting



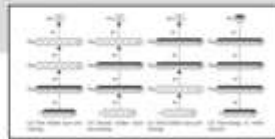
V. Vapnik - C. Cortes



- Limitations of learning prior knowledge
- Kernel function: Human Intervention

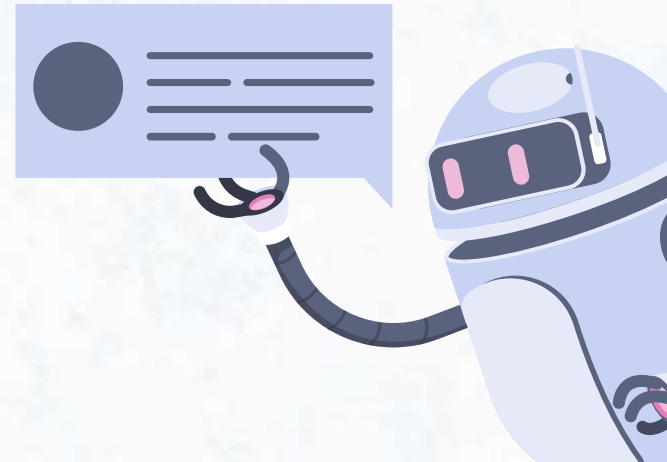


G. Hinton - S. Ruslan



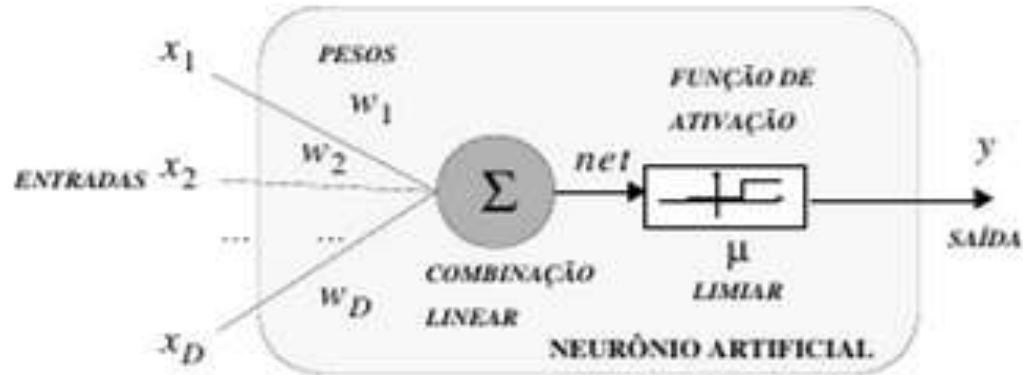
- Hierarchical feature Learning

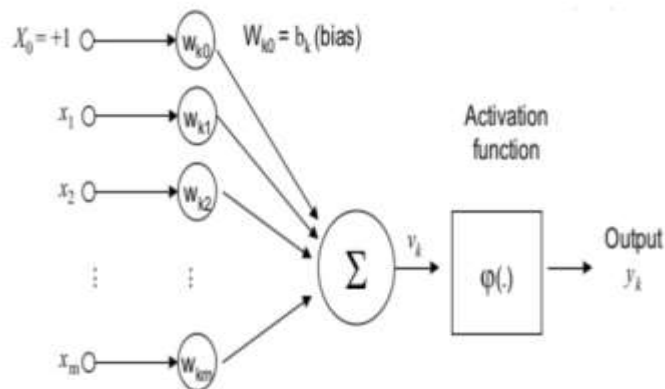
3 → Estrutura de uma Rede Neural



Perceptron

- É um modelo simples de rede neural artificial que recebe as entradas, processa-as e produz uma saída. É composto por um conjunto de entradas, pesos sinápticos, uma função de ativação de uma saída.





Elemento	Representação	Descrição
sinais de entrada	$\{x_1, x_2, \dots, x_n\}$	São as variáveis do problema estudado. Cada valor de entrada no vetor é um tipo de valor associado à amostra. Ex: x_1 = valor de declividade; x_2 = espessura; x_3 = leitura do instrumento.
pesos sinápticos	$\{w_1, w_2, \dots, w_n\}$	São valores associados a cada variável de entrada de forma a ponderar cada uma delas no problema em questão. Os pesos sinápticos iniciam com valores aleatórios entre 0 e 1, e são ajustados no processo de cálculo da rede (doravante melhor explicado).
combinador linear	Σ	Tem a função de agrupar os produtos dos sinais de entrada e pesos sinápticos.

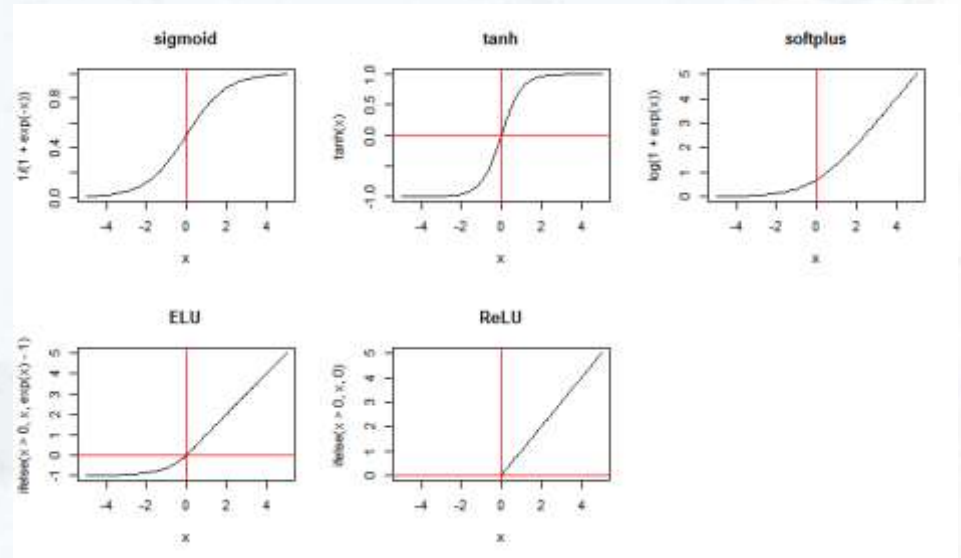
$$Y = \text{Activation}(\Sigma(\text{weight} * \text{input}) + \text{bias})$$

Elemento	Representação	Descrição
limiar de ativação	θ	Determina um valor que deve ser atingido pelo combinador linear de forma a produzir um resultado pelo neurônio.
potencial de ativação	u	É a diferença de valores entre o combinador linear e o limiar de ativação e determina o comportamento do neurônio (excitatório ou inibitório).
função de ativação	$g(\cdot)$	Normaliza a saída do neurônio para valores que possam ser interpretados.
sinal de saída	y	É o resultado final do processo do neurônio. Pode ser o produto final geral da análise, bem como ser utilizado por outros neurônios da rede.

$$Xw = y$$

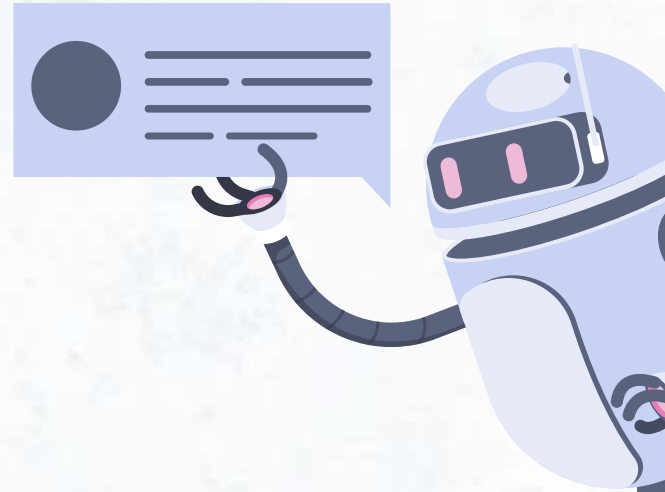
$$\begin{bmatrix} 1 & x_{11} & \dots & x_{1d} \\ 1 & x_{21} & \dots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & \dots & x_{nd} \end{bmatrix} \times \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_d \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix}$$

Multiplicação dos pesos sinápticos com os modelos de entrada (modo simplificado)



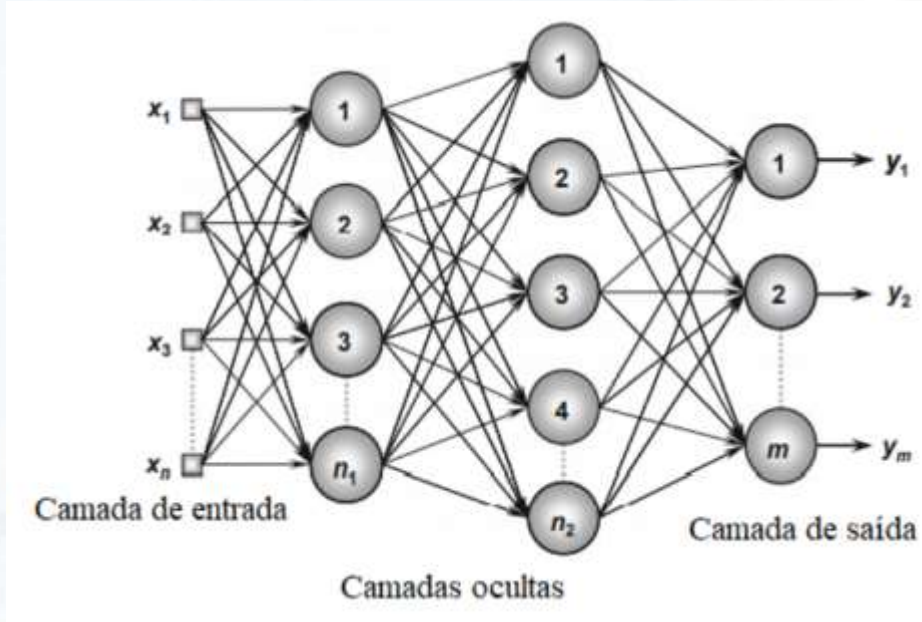
Funções de ativação

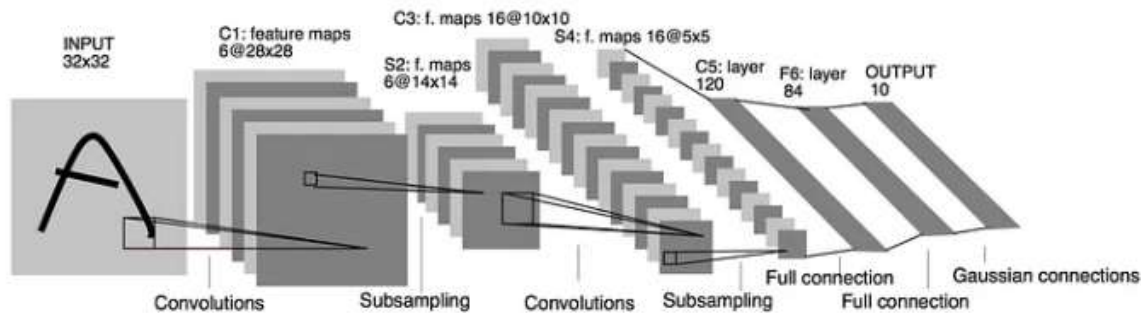
4 → Arquitetura



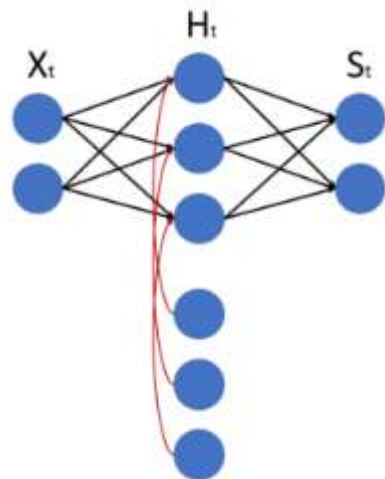
As redes neurais podem ser estruturadas em camadas conforme a figura ao lado

- **Camada de entrada:** responsável por receber as informações de entrada
- **Camadas ocultas:** Camadas onde os neurônios estão localizados
- **Camadas de saída:** Responsável de produzir a saída final da rede resultante do processamento executado pelas camadas anteriores

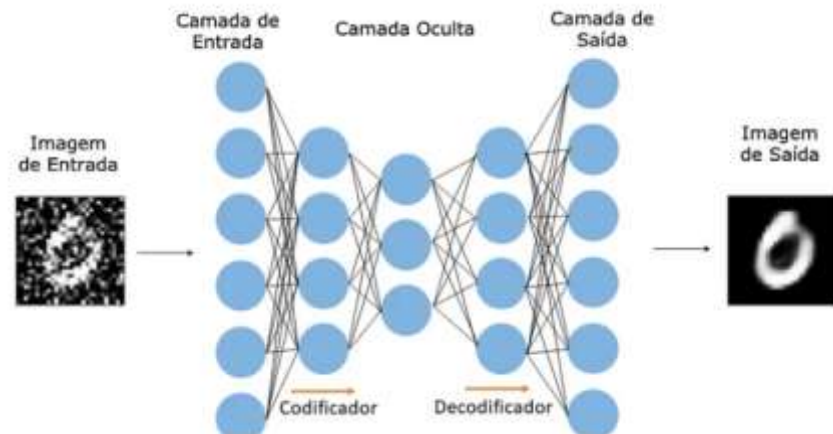




Cnn



Rnn



Autoencoder Convolutacional

LLM (Large Language Models)

Entrada

Camada onde entra os dados

Embedding

mapeia cada palavra do texto de entrada para um vetor denso de números reais, chamado de embedding. Essa camada captura as informações semânticas e distribucionais das palavras no texto.

Encoder

codificação chamadas de encoder. Cada camada do encoder é composta por uma subcamada de self-attention e uma subcamada de feed-forward.

self-attention

permite que o modelo leve em consideração as interações entre todas as palavras do texto. Ele atribui pesos diferentes a cada palavra com base na importância relativa para a compreensão do contexto

feed-forward

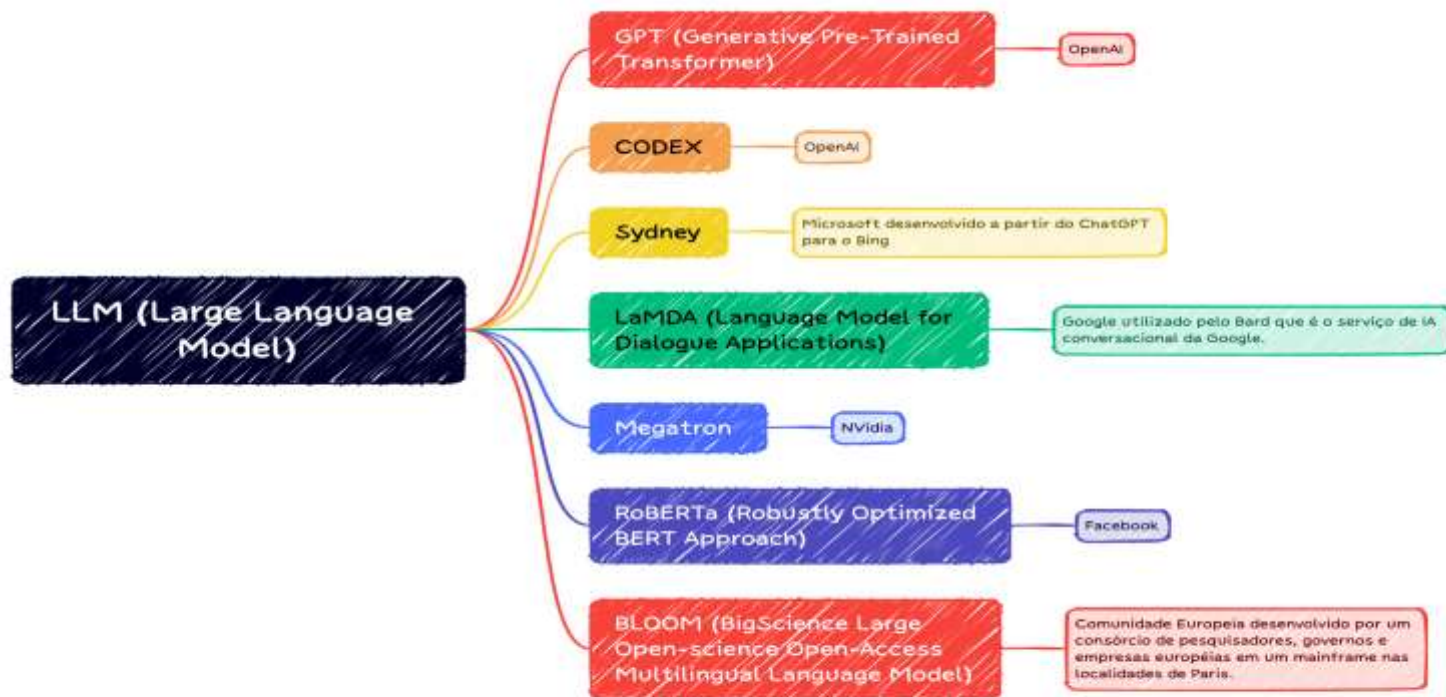
processa as representações de cada palavra independentemente, transformando-as em uma nova representação.

Decoder

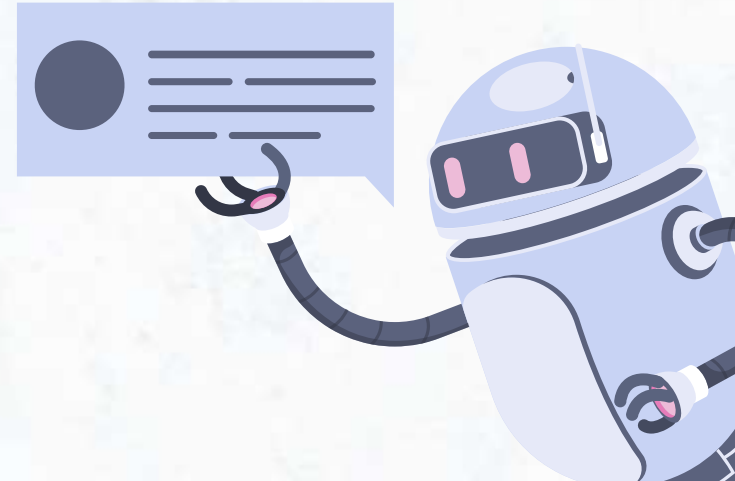
semelhante à camada de encoder, mas também possui uma terceira subcamada de self-attention. Essa máscara garante que, durante a geração de texto, o modelo só tenha acesso às palavras anteriores, evitando informações futuras.

Saída

é uma distribuição de probabilidade sobre o vocabulário, onde cada palavra recebe um valor de probabilidade. Com base nessa distribuição, é possível gerar sequências de palavras coerentes ou prever a próxima palavra em uma tarefa de preenchimento de lacunas, por exemplo.

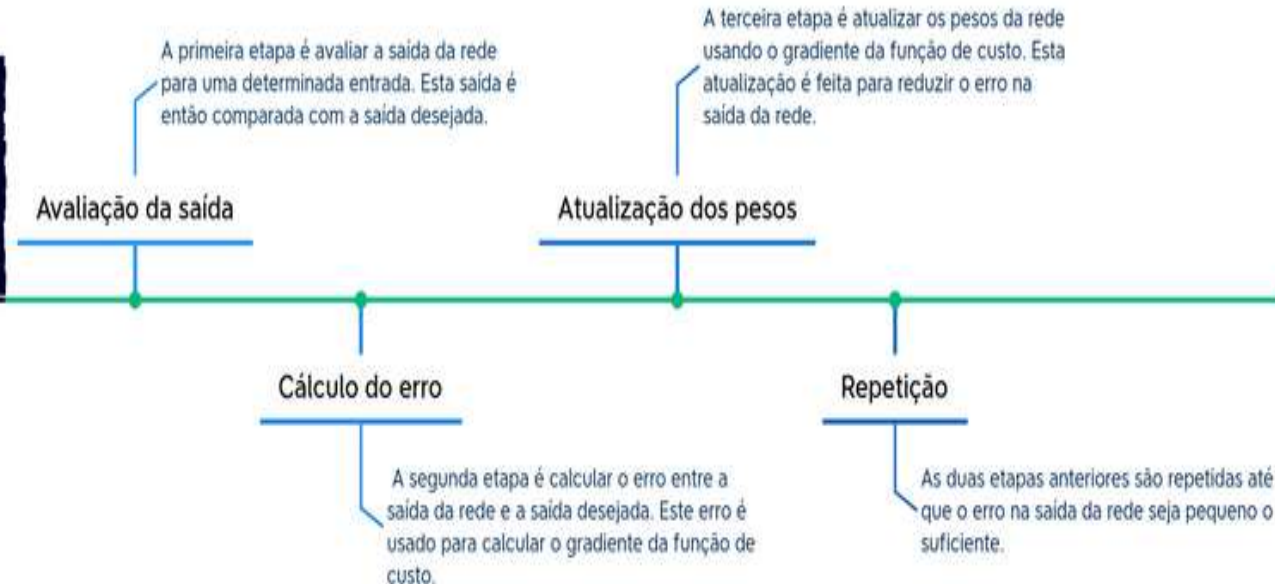


4 → Treinamento



Backpropagation

Sobre o processo de backpropagation (Retroalimentação)

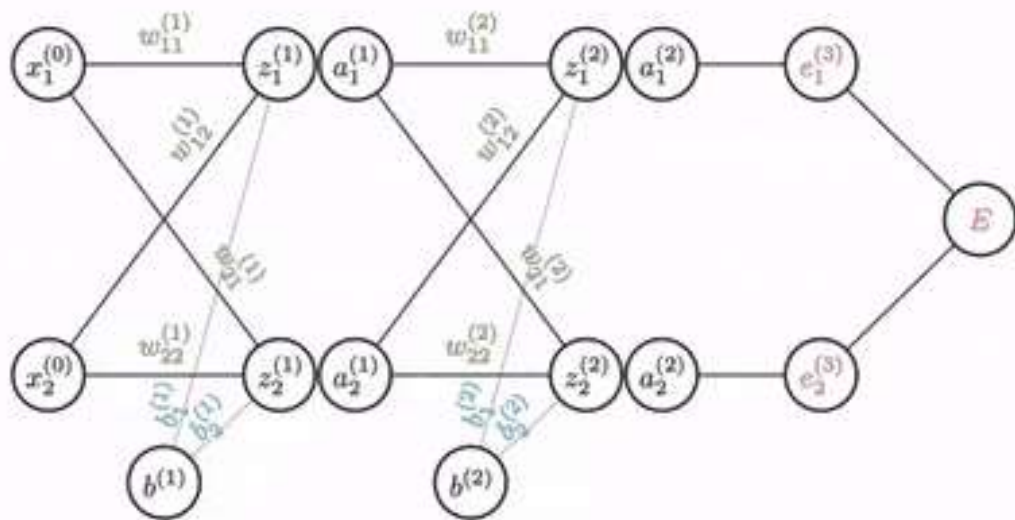


$$\frac{\partial E}{\partial w_{11}^{(2)}} = \frac{\partial e_1^{(3)}}{\partial a_1^{(2)}} \frac{\partial a_1^{(2)}}{\partial z_1^{(2)}} \frac{\partial z_1^{(2)}}{\partial w_{11}^{(2)}}$$

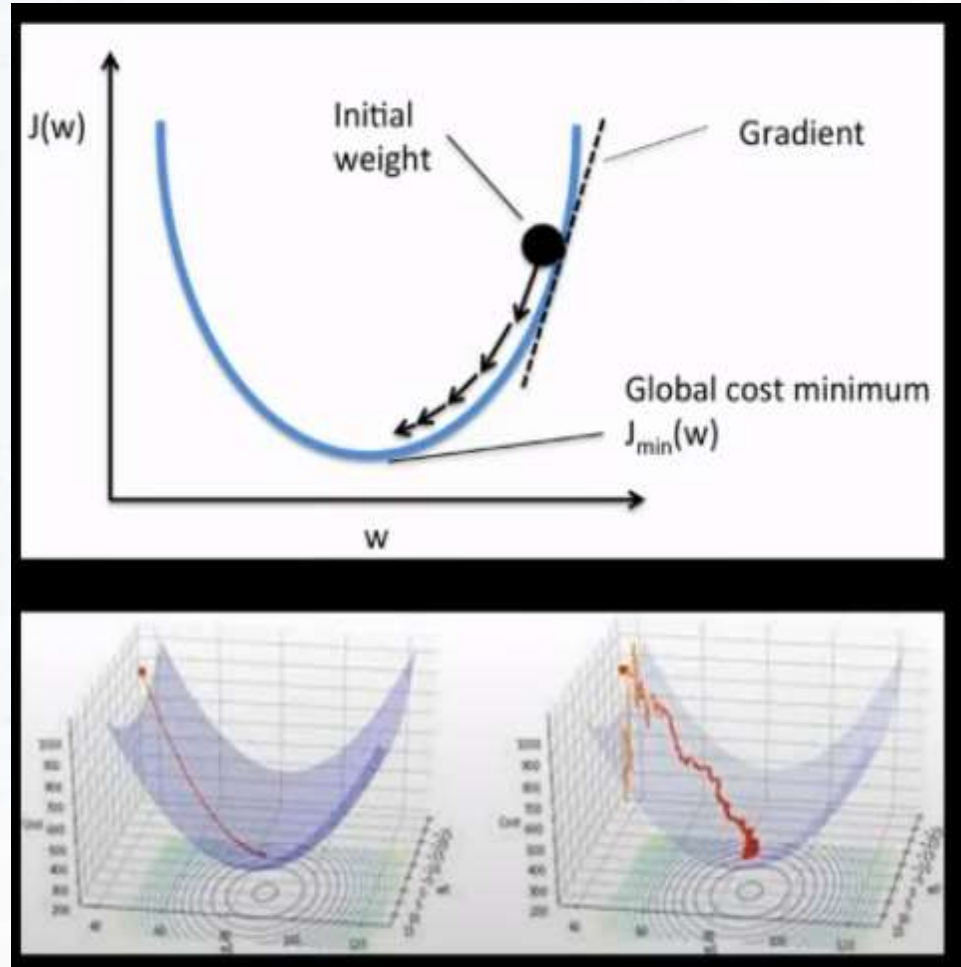
$$\frac{\partial E}{\partial w_{12}^{(2)}} = \frac{\partial e_1^{(3)}}{\partial a_1^{(2)}} \frac{\partial a_1^{(2)}}{\partial z_1^{(2)}} \frac{\partial z_1^{(2)}}{\partial w_{12}^{(2)}}$$

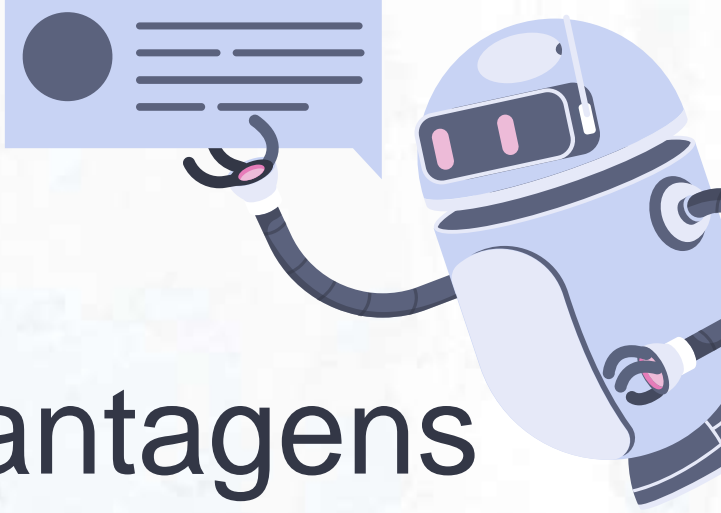
$$\frac{\partial E}{\partial w_{21}^{(2)}} = \frac{\partial e_2^{(3)}}{\partial a_2^{(2)}} \frac{\partial a_2^{(2)}}{\partial z_2^{(2)}} \frac{\partial z_2^{(2)}}{\partial w_{21}^{(2)}}$$

$$\frac{\partial E}{\partial w_{22}^{(2)}} = \frac{\partial e_2^{(3)}}{\partial a_2^{(2)}} \frac{\partial a_2^{(2)}}{\partial z_2^{(2)}} \frac{\partial z_2^{(2)}}{\partial w_{22}^{(2)}}$$



Na realidade, o backpropagation é apenas uma aplicação da regra da cadeia (chain rule). Como as redes neurais são estruturas de modelo de aprendizado de máquina multicamadas complicadas, cada peso “contribui” para o erro geral de uma maneira mais complexa e, portanto, as derivadas reais exigem muito esforço para serem produzidas. No entanto, uma vez que passamos pelo cálculo, o backpropagation das redes neurais é equivalente à descida de gradiente típica para regressão logística / linear.



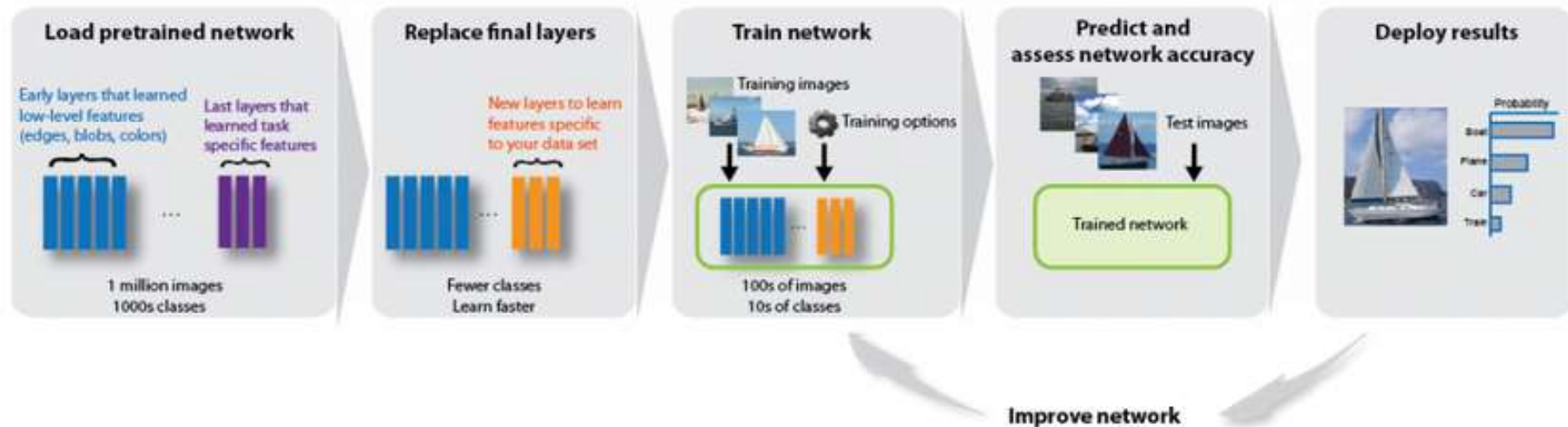


Vantagens e desvantagens da utilização de redes neurais

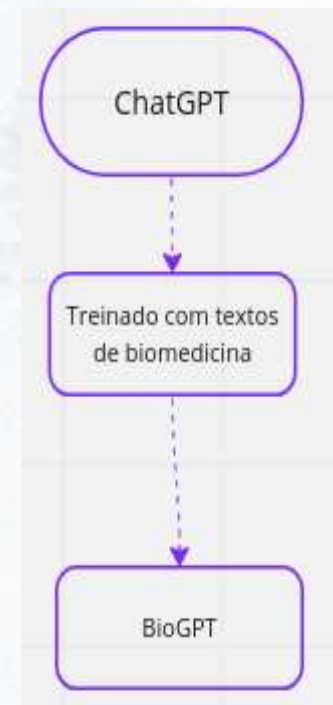
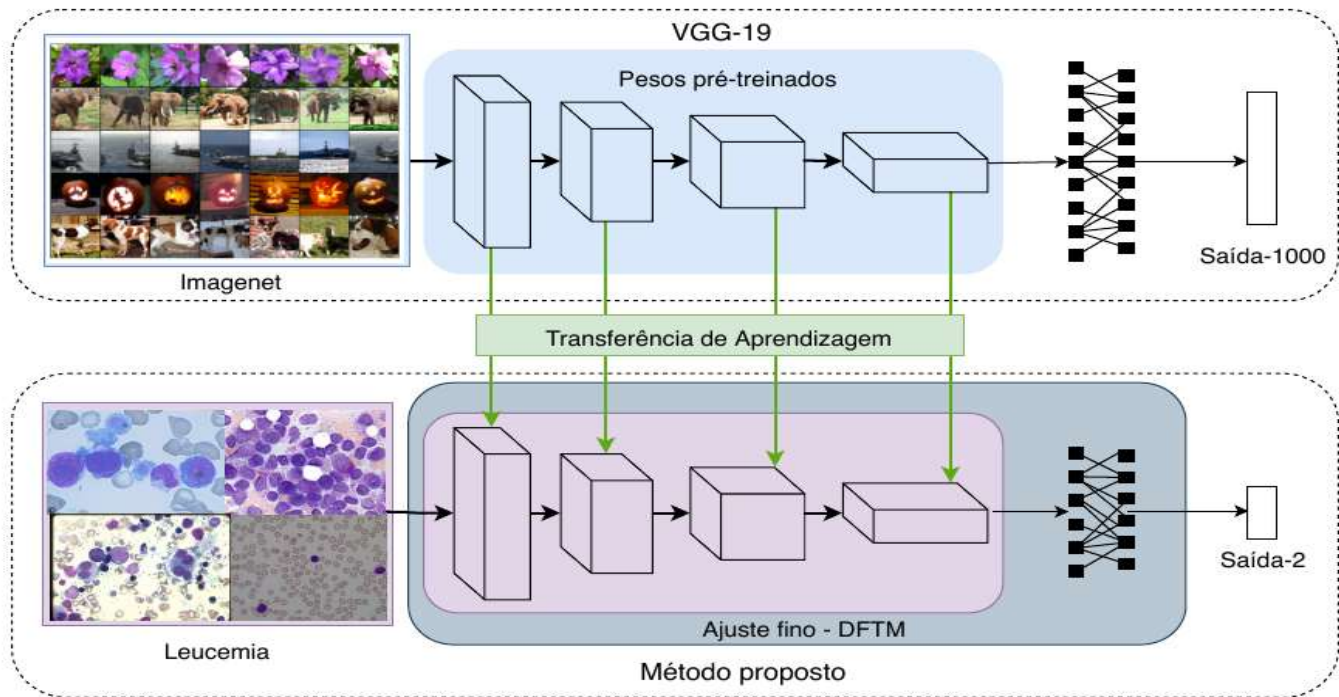
Vantagens das redes neurais

- (+) Pode ser usada como base para construir outra rede
- (+) Tomada de decisões
- (+) Paralelismo

Reuse Pretrained Network



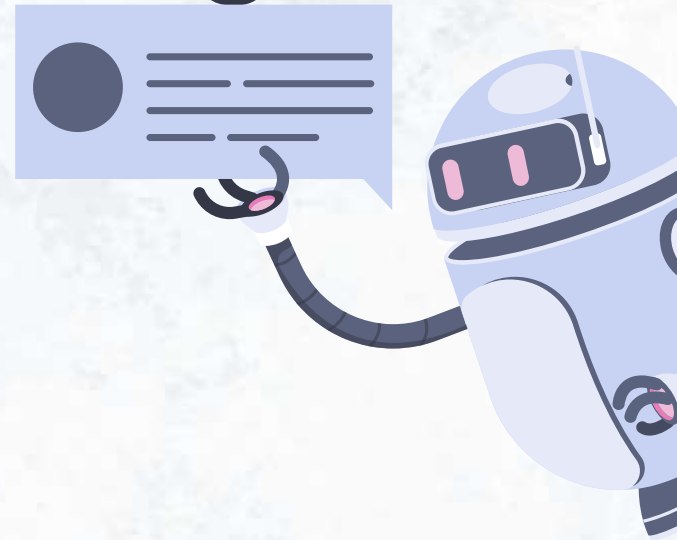
Exemplo de rede pré-treinadas



Desvantagens das redes neurais

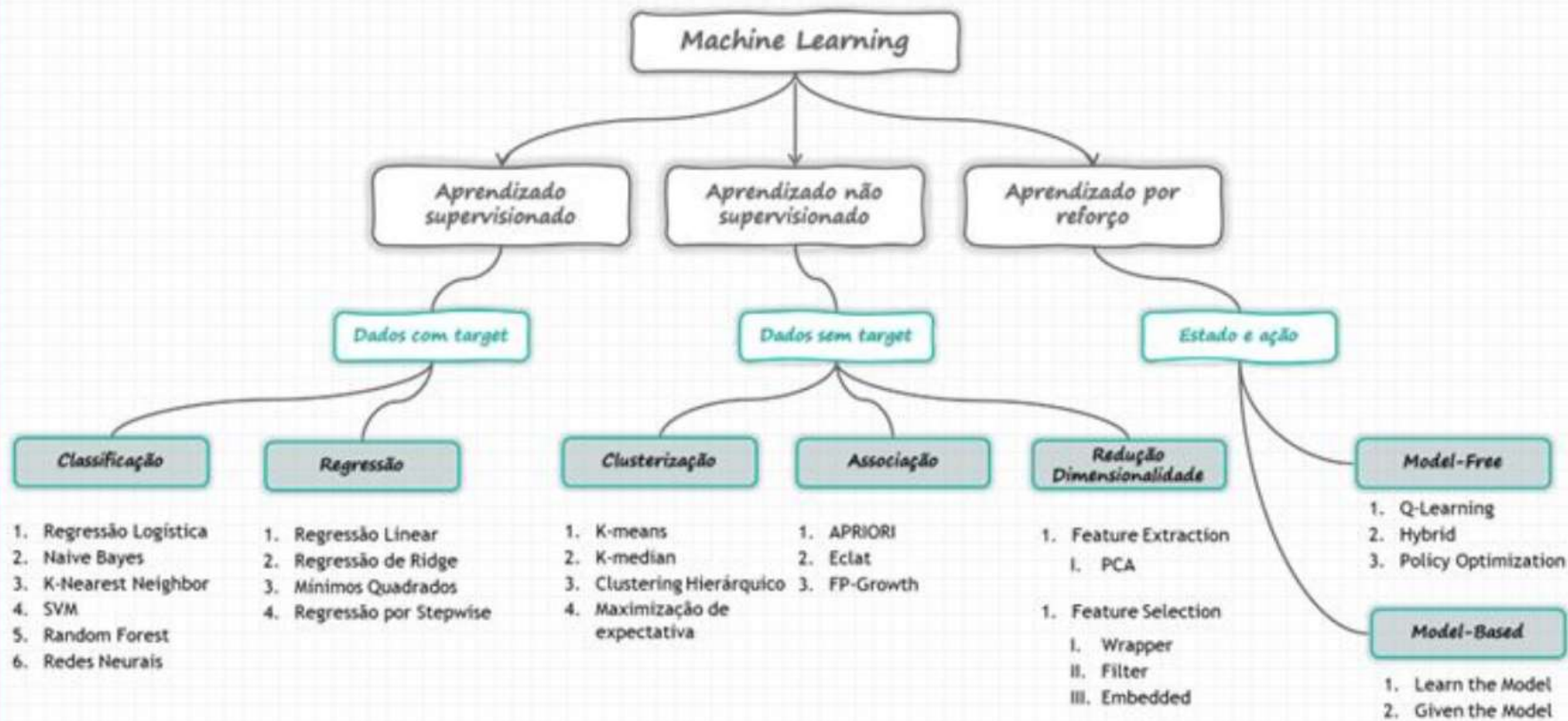
- (+) Treinamento demorado do modelo (especialmente se não tiver uma gpu)
- (+) Caixa preta (já que não se tem controle dos pesos sinápticos)
- (+) Grande volume de dados para que a rede neural aprenda
- (+) Preparação de dados criteriosa

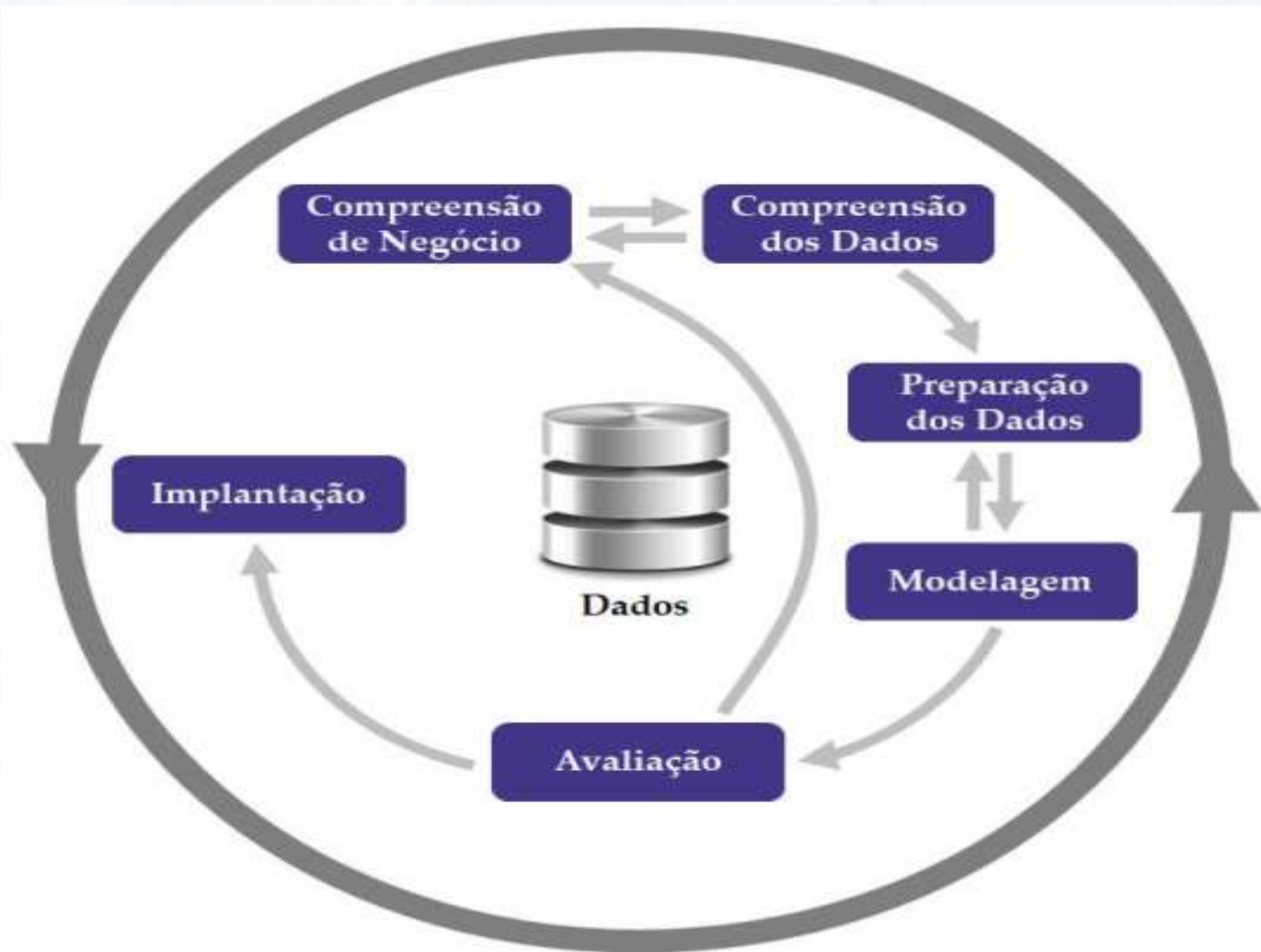
4 → Mãos ao Código



Problema

- Dado um título de notícia, será construído um modelo na qual vai classificar se a mesma é uma notícia verdadeira (no caso, o modelo retornará como 0) ou se aquela notícia é fake news (no caso, o modelo retornará como 1).

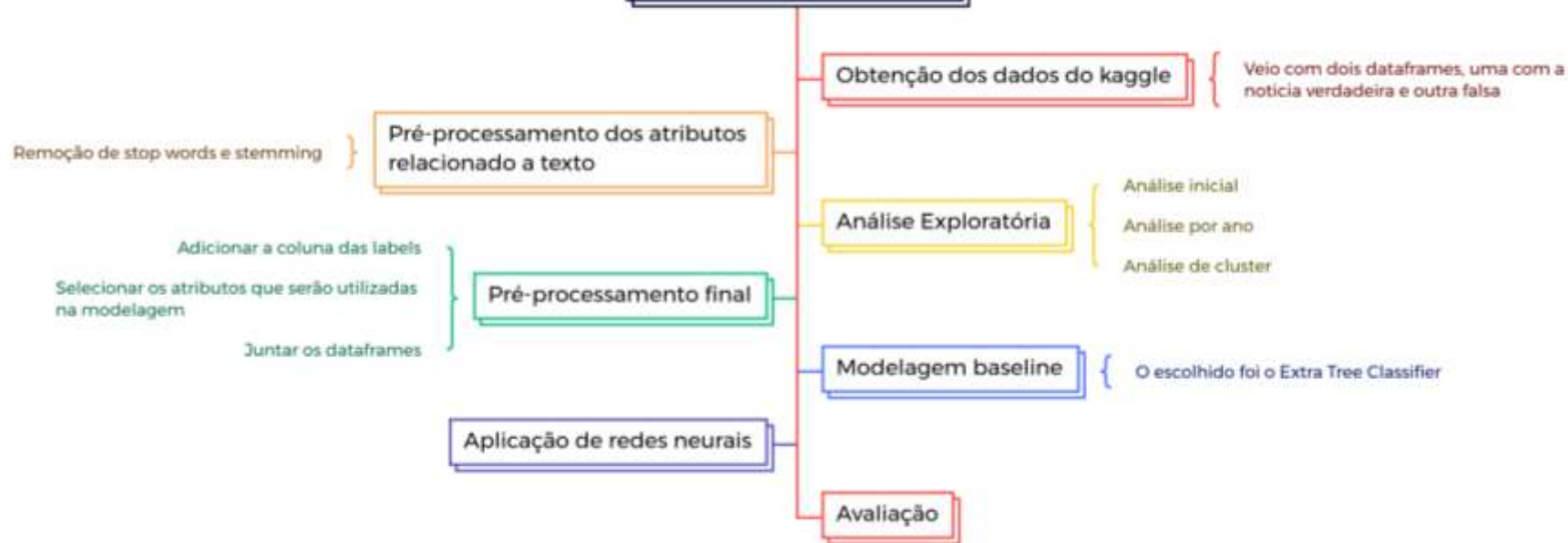




Principais Bibliotecas



Fluxograma do Projeto



	Extra tree	MLP	RNN
Acurácia	0.9421985815602837	0.9460992813110352	0.9522458910942078
Precisão	0.9464505035617784	0.945361852645874	0.9503632187843323
Recall	0.9342870999030067	0.9439864158630371	0.951745867729187
F1-score	0.9403294691885296	0.944673633598654	0.9510540407308745

Obrigado



Você tem alguma pergunta?

@gustavoramos82

CRÉDITOS: Este modelo de apresentação foi criado pelo **Slidesgo**, e inclui ícones da **Flaticon**, infográficos e imagens da **Freepik** e conteúdo de **Eliana Delacour**

Repositório do Github



Perfil do LinkedIn

