

Embarcadero Conference 2024

Inovação faz parte do nosso DNA!

Técnicas de Engenharia de Prompt para desenvolvedores

{Carlos Eduardo Paulino }



{Carlos Eduardo Paulino }

- Desenvolvedor Delphi há mais de 25 anos.
- Criador dos canais **@DelphiCleanCode**
- Dev Backend com sistemas ERP e CRM 3º Setor
- Dev componentes TAsaas, TPagSeguro, TMercadoPago, TPicPay etc.





Trabalho com IA



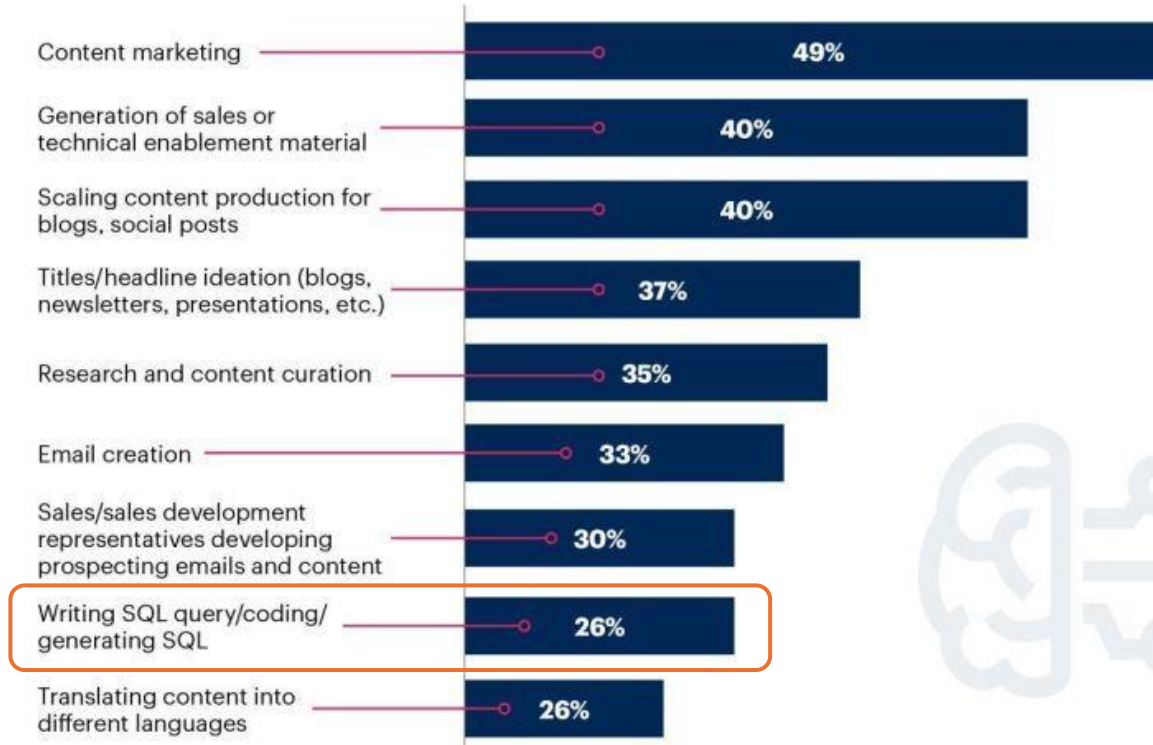
Qual área?



Sou Engenheiro de
Prompt.



Generative AI use cases



Q. What generative AI use case(s) is your company currently implementing/ planning to implement?
Source: Gartner
© 2024 Gartner, Inc. and/or its affiliates. All rights reserved. 2952678

Gartner®

Agenda

- IA Uma visão geral
- LLM conceitos e aplicações
- Arquitetura Transformers
- Engenharia de PROMPT
- Principais Técnicas de PROMPT
- Exemplos e dicas



Artificial Intelligence

A IA envolve técnicas que equipam os computadores para emular o comportamento humano, permitindo-lhes aprender, tomar decisões, reconhecer padrões e resolver problemas complexos de uma forma semelhante à inteligência humana.

Machine Learning

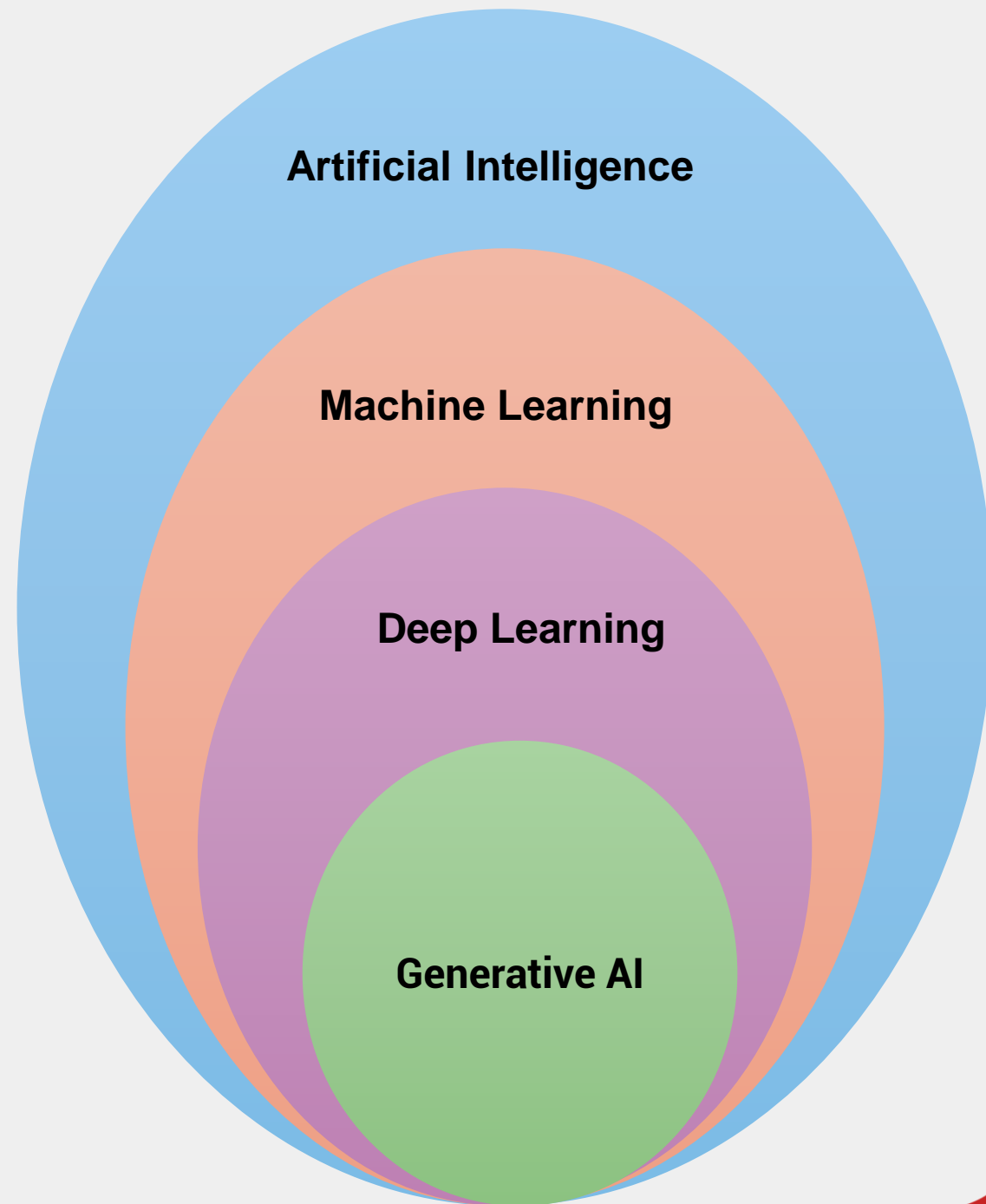
Criação e utilização de algoritmos avançados para detectar padrões em grandes conjuntos de dados, permitindo que as máquinas aprendam e se adaptem.

Deep Learning

Utilização de redes neurais para processamento aprofundado de dados e tarefas analíticas. DL aproveita múltiplas camadas de redes neurais artificiais para extrair recursos de alto nível de dados brutos de entrada, simulando a maneira como o cérebro humano percebe e entende o mundo.

Generative AI

IA generativa é um subconjunto de modelos de Deep Learning que gera conteúdo como texto, imagens ou código com base nas informações fornecidas. Treinados em vastos conjuntos de dados, estes modelos detectam padrões e criam resultados sem instrução explícita, utilizando uma combinação de aprendizagem supervisionada e não supervisionada.

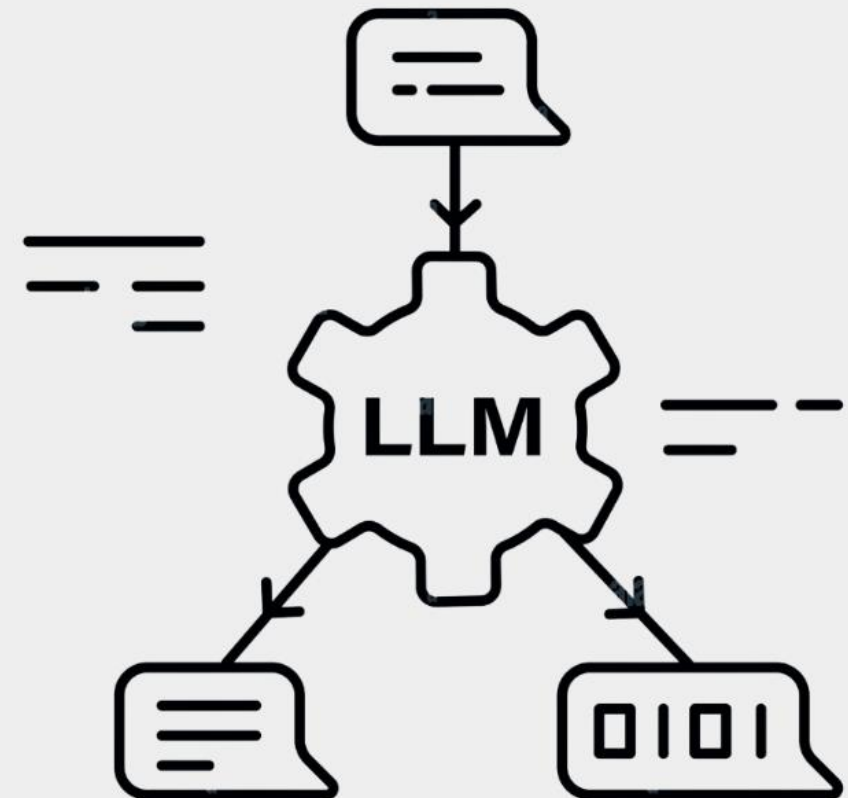
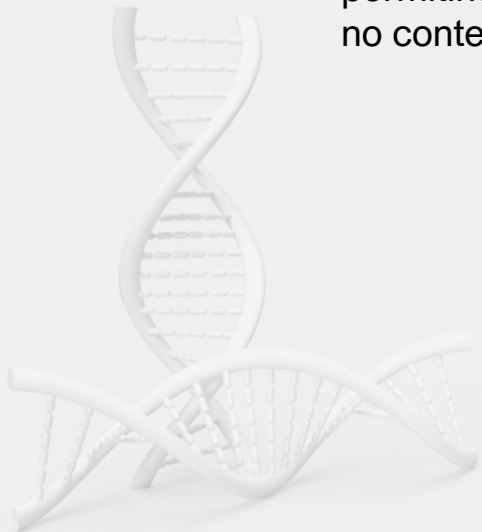


LLM (Large Language Model)

LLM é um modelo de linguagem avançado que utiliza algoritmos de processamento de linguagem natural para entender e gerar texto. Ele é treinado com grandes volumes de dados textuais, para aprender a estrutura da linguagem e as relações semânticas entre palavras e frases.

Por exemplo, o GPT (Generative Pre-trained Transformer) é um modelo de linguagem avançado que utiliza a arquitetura de transformadores para processar texto de maneira eficiente, capturando dependências de longo alcance.

Ele é treinado em tarefas de previsão de palavras ou frases, permitindo que gere texto coerente e de qualidade com base no contexto fornecido



Onde um LLM é aplicado?



- **Geração de texto:** Gerar automaticamente resumos de texto, artigos, redações e até mesmo diálogos fictícios.



- **Tradução automática:** Traduzir texto entre diferentes idiomas, fornecendo resultados mais precisos e fluentes.



- **Assistentes virtuais:** Chatbots e sistemas de perguntas e respostas são alimentados por LLMs para entender e gerar respostas coerentes em tempo real.



- **Análise de sentimentos:** Identificação de textos como positivos, negativos ou neutros. Isso é útil em áreas como análise de mídias sociais e pesquisa de opinião.



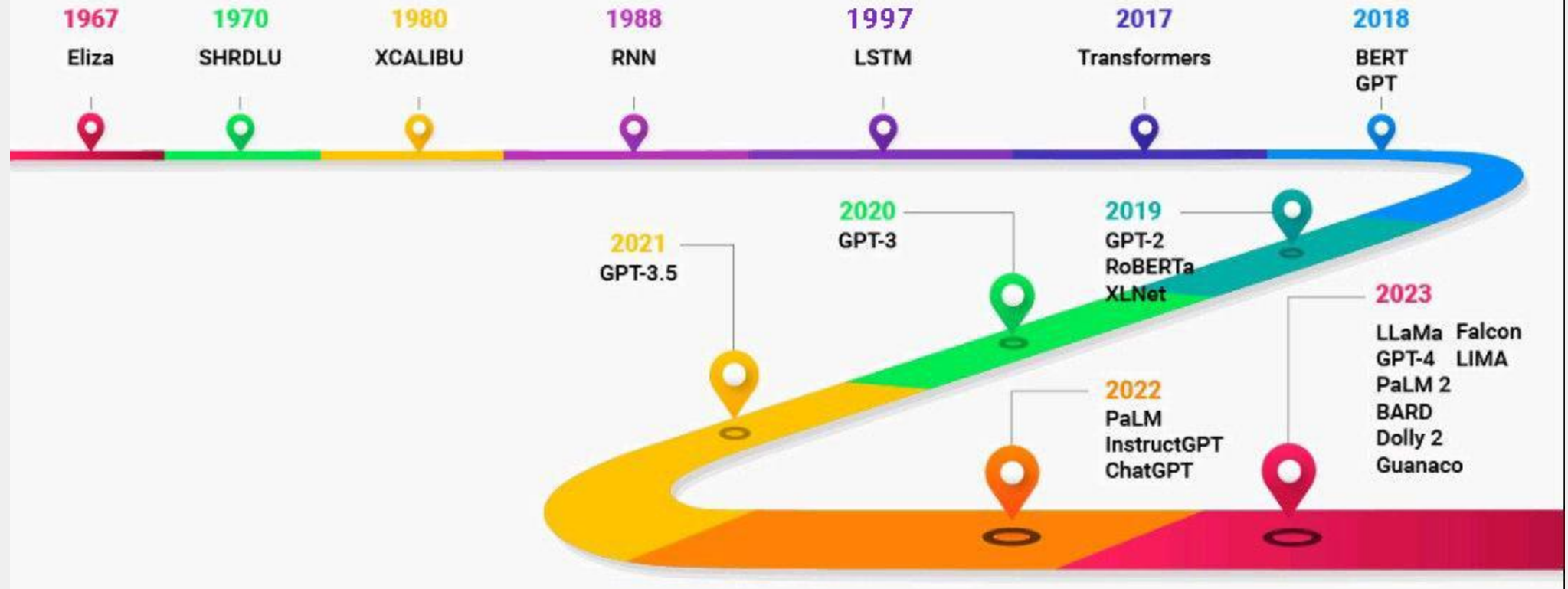
- **Recomendação de conteúdo:** Plataformas de streaming, comércio eletrônico e mídias sociais utilizam LLMs para recomendar conteúdo personalizado com base nos interesses e comportamentos dos usuários.



- **Correção automática:** Teclados e editores de texto com sugestões de palavras e corretor ortográficos e gramaticais.



Evolution of Large Language Models



1988 - 1997 – RNN e LSTM

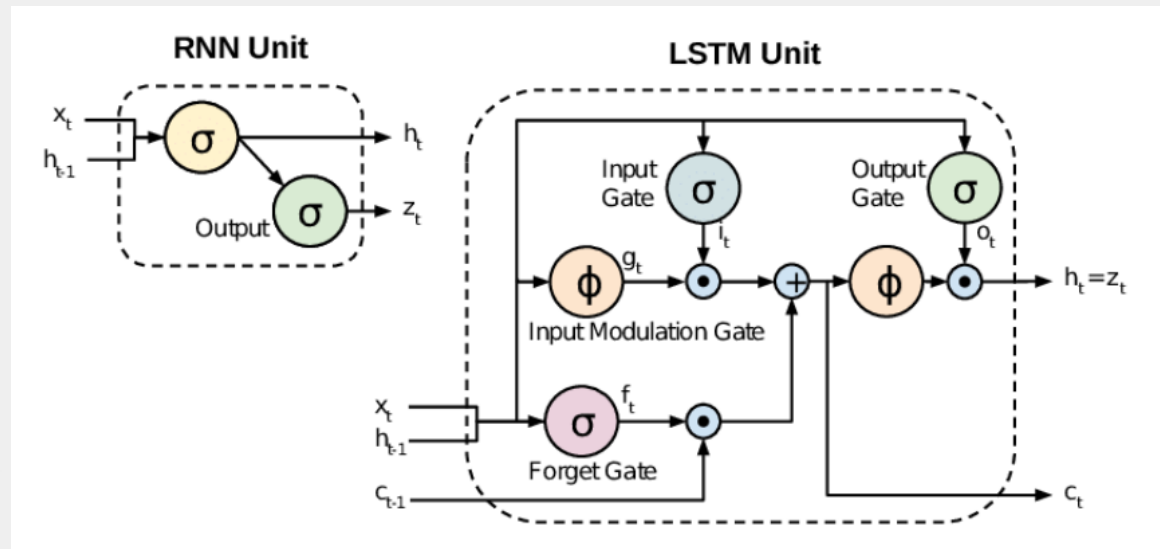
RNNs são um tipo de arquitetura de rede neural projetada para lidar com dados sequenciais, como texto.

Elas são capazes de manter uma “memória” interna que permite processar sequências de entrada e gerar sequências de saída.

No entanto, as RNNs tradicionais sofrem de um problema conhecido como “vanishing gradient”, que dificulta a aprendizagem de dependências de longo prazo em sequências de texto.

LSTMs foram propostas. Elas são uma variação das RNNs que incorpora mecanismos de “portas” para controlar o fluxo de informações na rede.

Isso permite que as LSTMs capturem dependências de longo prazo em dados sequenciais de forma mais eficaz, tornando-as ideais para tarefas de processamento de linguagem natural.



1988..1997 - RNN e LSTM

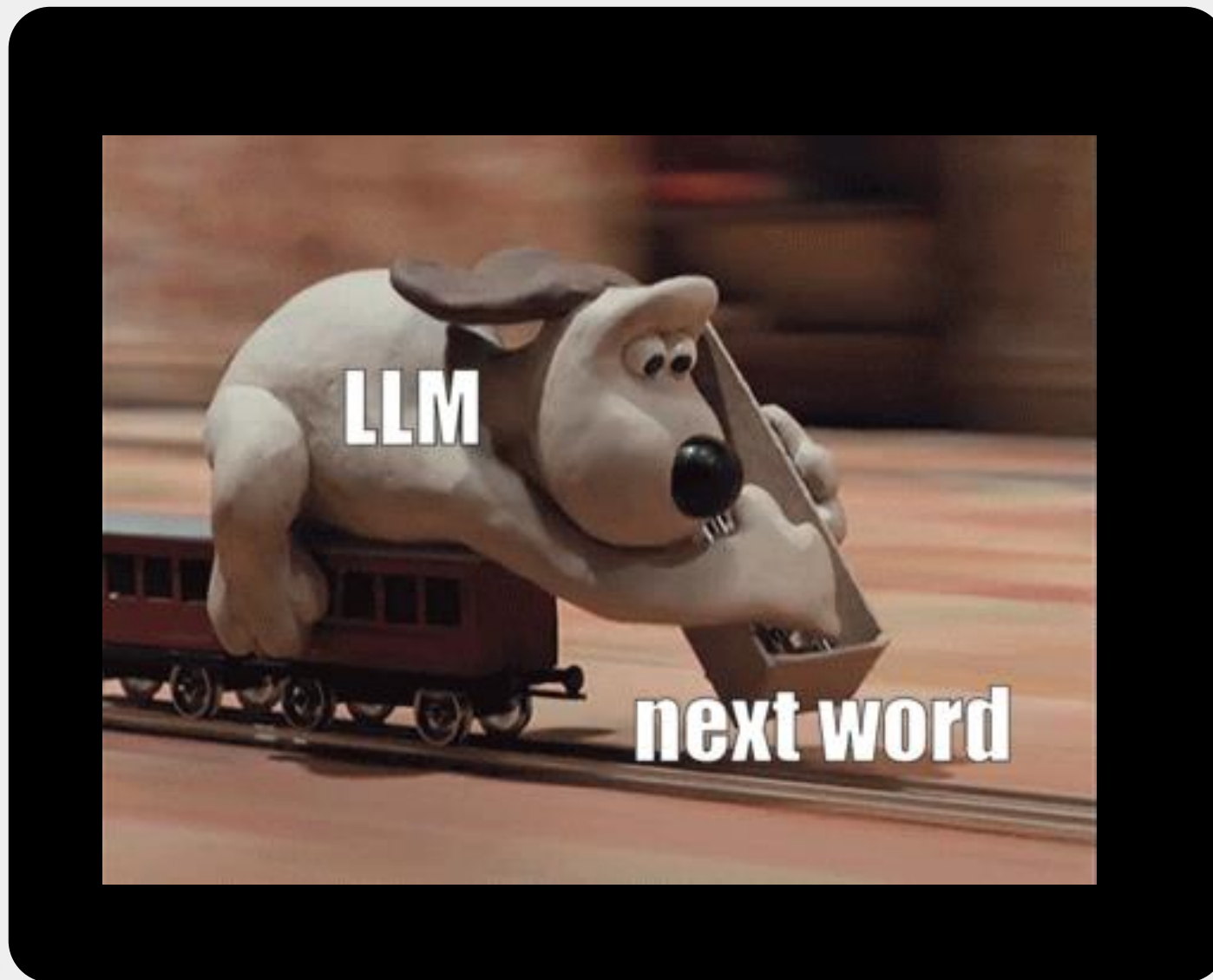
Eu desenvolvo em Delphi utilizando Linguagem
POO
Banco de Dados

Eu desenvolvo em Delphi utilizando

Eu desenvolvo em Delphi utilizando boas práticas

Eu desenvolvo em Delphi utilizando boas práticas de programação





2017 – Arquitetura Transformer

Os Transformers são um tipo de modelo baseado em atenção que gerencia melhor o processamento de sequências de dados, como texto.

A arquitetura Transformer permitiu grandes avanços na geração de texto, resultando em modelos como o GPT.

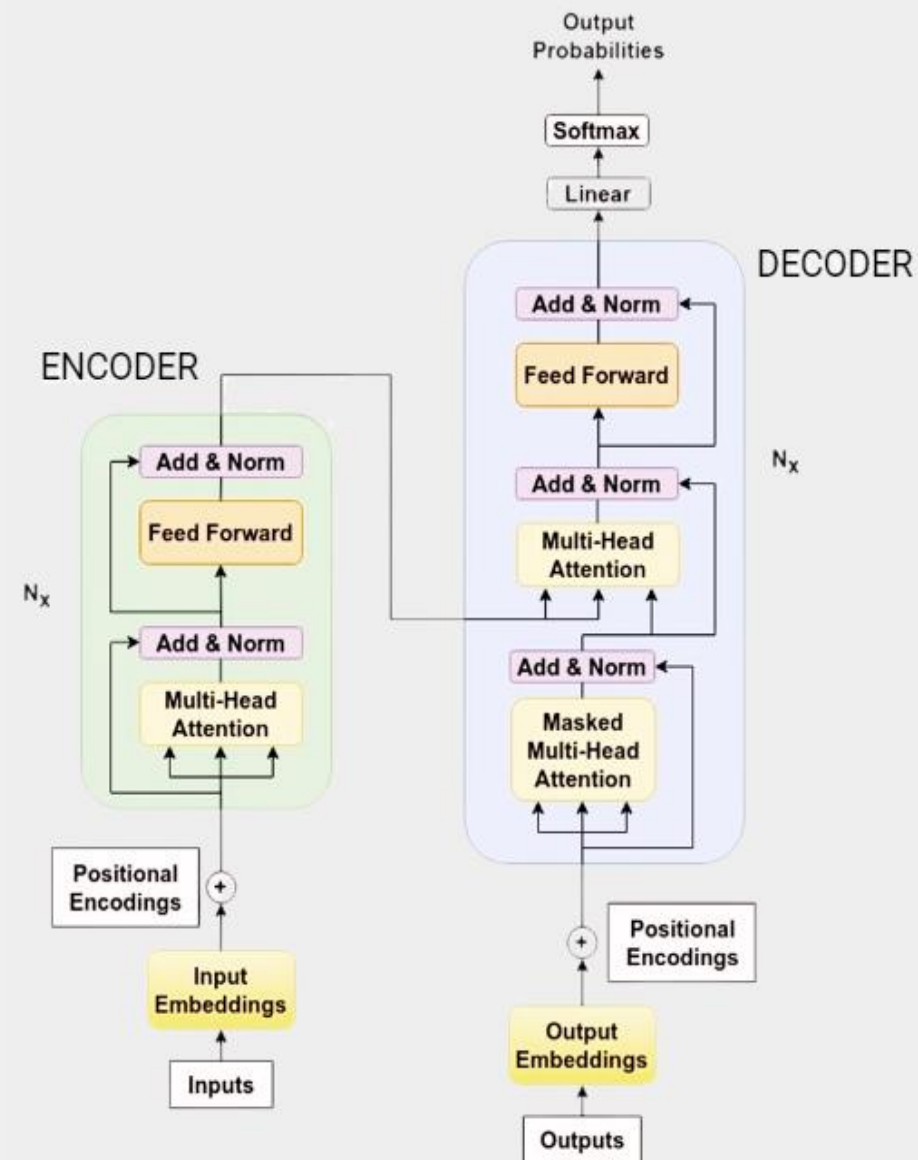
Esses modelos podem compreender e gerar texto com uma coerência, relevância e diversidade sem precedentes, impactando diretamente aplicações como tradução automática, geração de conteúdo e chatbots.



2017 – Arquitetura Transformer

Os Transformers são um tipo de modelo baseado em atenção que gerencia melhor o processamento de sequências de dados, como texto. A arquitetura Transformer permitiu grandes avanços na geração de texto, resultando em modelos como o GPT.

Esses modelos podem compreender e gerar texto com uma coerência, relevância e diversidade sem precedentes, impactando diretamente aplicações como tradução automática, geração de conteúdo e chatbots.



Attention

Is

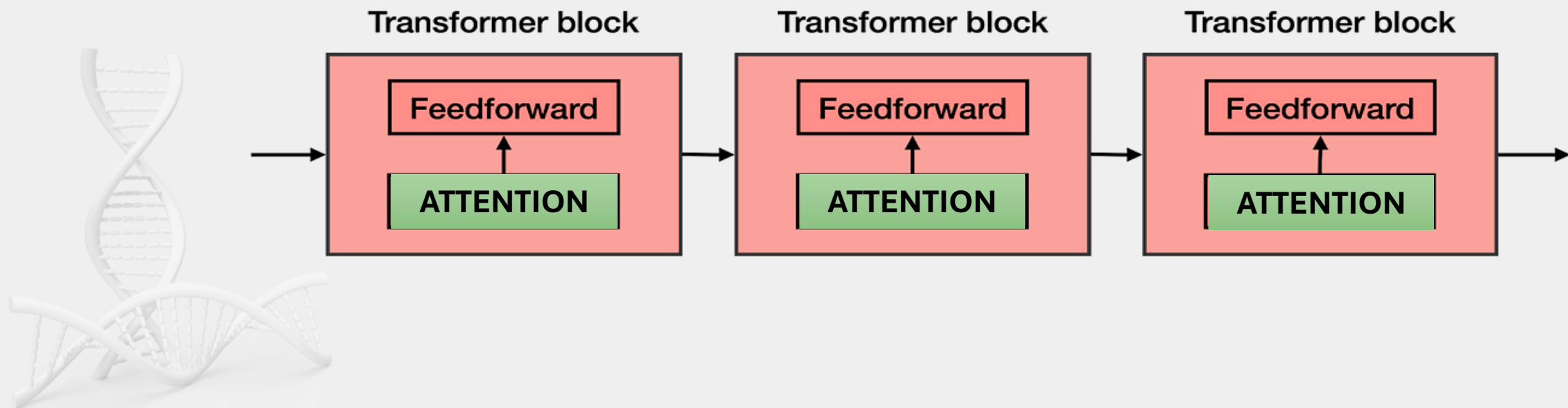
All

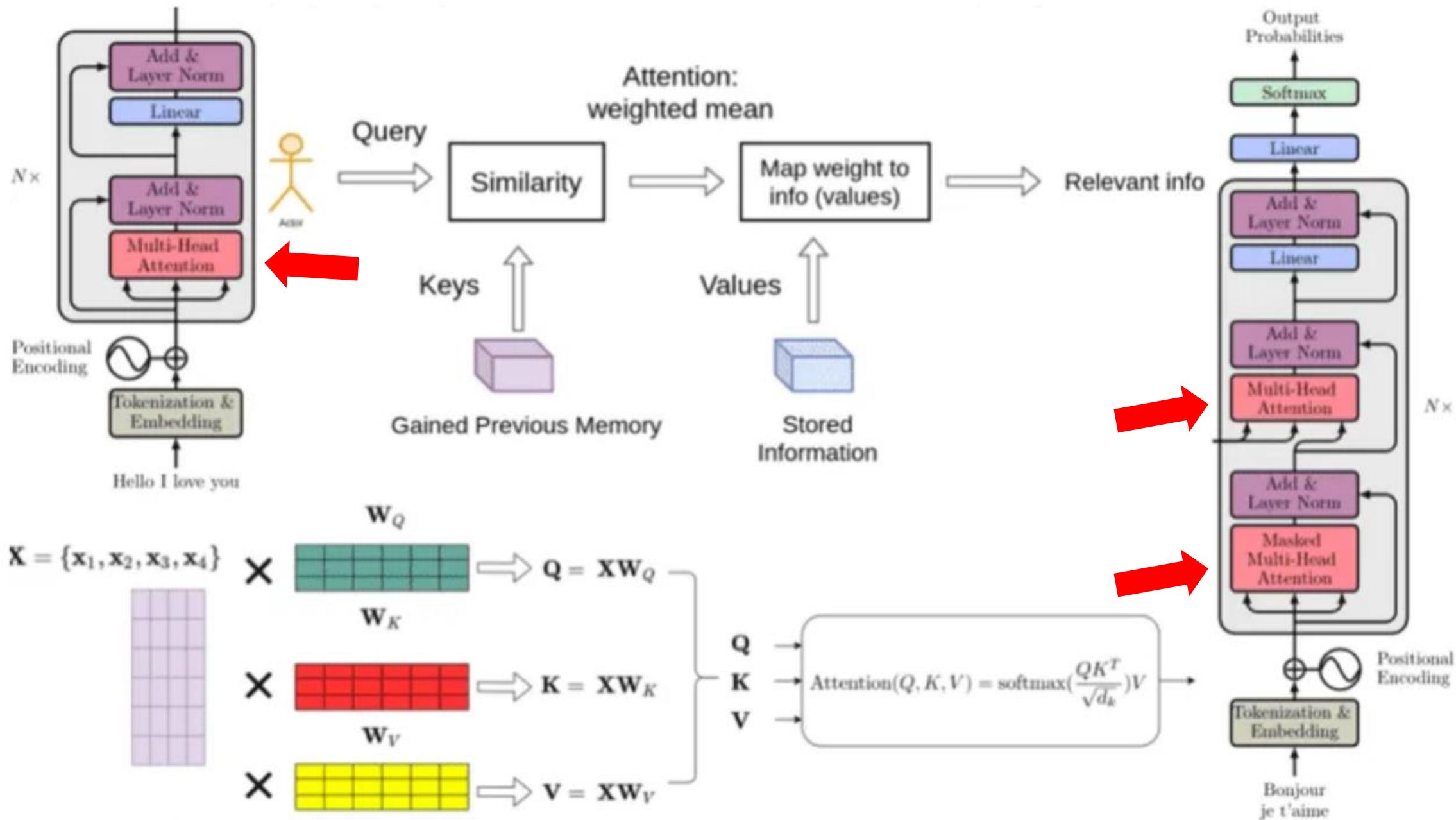
You

Need

Em 2017 um dos marcos mais importantes para a GenAI aconteceu:

A publicação do artigo “**Attention is All You Need**” por pesquisadores do Google Brain, marcou uma grande evolução nos modelos de linguagem, porque os Transformers conseguem capturar contextos mais amplos e realizar paralelismo em seu processamento, o que resulta em um treinamento mais eficiente em grandes conjuntos de dados





PROMPT

Gere método em Delphi para geração de um gráfico, utilizando o componente TChart, considerando os seguintes <valores>...</valores>

GPT-4

Tokens	Characters
33	130

Gere método em Delphi para geração de um gráfico, utilizando o componente TChart, considerando os seguintes <valores>...</valores>

GPT-3

Tokens	Characters
45	130

Gere método em Delphi para geração de um gráfico, utilizando o componente TChart, considerando os seguintes <valores>...</valores>



Calculation Type

Select 'Number' to directly enter the number of tokens, words, or characters. Choose 'Sample Text' to input text and automatically calculate the necessary tokens based on your input.

Number

Sample Text

Input Prompt

This is the system prompt followed by additional user or assistant messages.

Gere método em Delphi para geração de um gráfico, utilizando o componente TChart, considerando os seguintes <valores> Janeiro = 10, Fevereiro = 20, Março = 30</valores>

41.75 tokens

167 chars

Number of API calls

How many times the LLM API will be called with the input prompt to generate the output.

1

Output

This is what the LLM has generated given the input prompt.

```
YValues[2] := 30;  
  
// Cria o TChartSeries  
Series := TLineSeries.Create(Self);  
Series.Chart := Chart1.Chart;  
Series.XValues := XValues;  
Series.YValues := YValues;
```

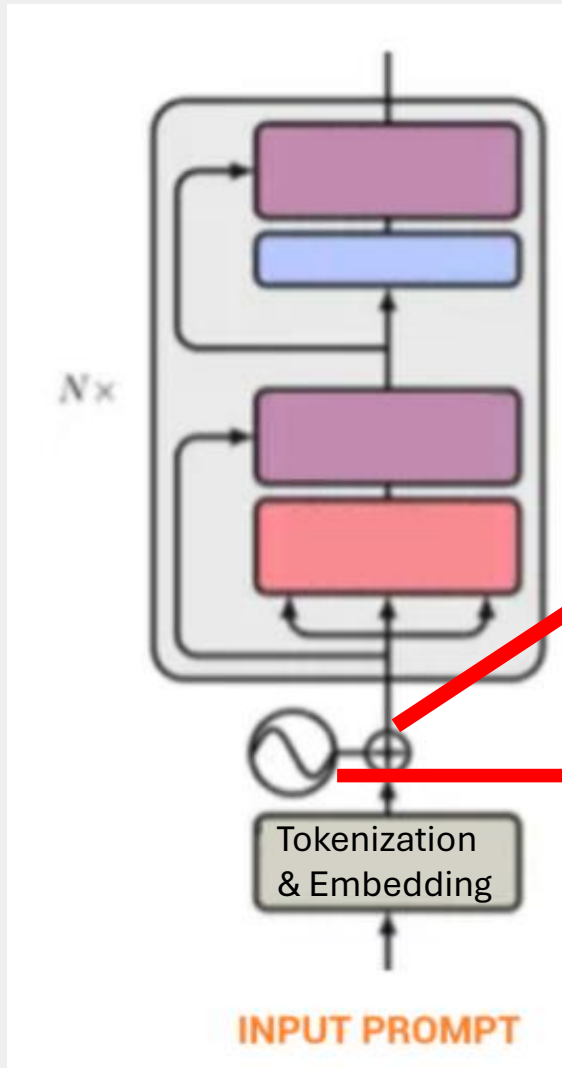
565.5 tokens

2262 chars



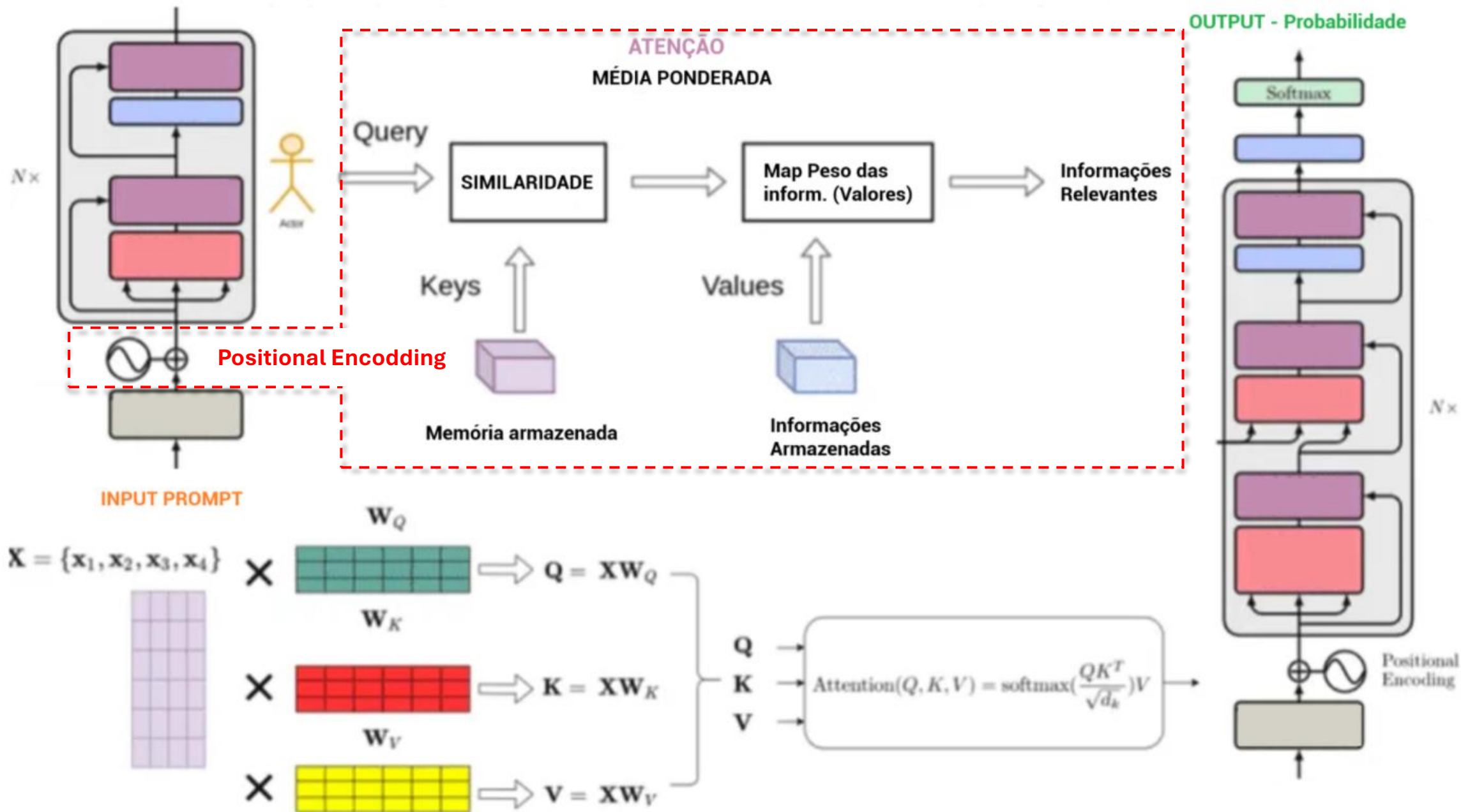
PROVIDER ▼	MODEL ▲	CONTEXT ▲	INPUT/PER TOKEN ▲	OUTPUT/PER TOKEN ▲	PER CALL ▲	TOTAL ▼	
AI Anthropic	Claude 3 Opus	200k	\$0.000015	\$0.000075	\$0.04304	\$0.04304	📌
🌀 openai	GPT-4	8k	\$0.00003	\$0.00006	\$0.03518	\$0.03518	📌
∞ Meta	Databricks Llama 3.1 (405B)	128k	\$0.00001	\$0.00003	\$0.01738	\$0.01738	📌
🌀 openai	GPT-4 Turbo	128k	\$0.00001	\$0.00003	\$0.01738	\$0.01738	📌
AI Anthropic	Claude 2.1	200k	\$0.000008	\$0.000024	\$0.01391	\$0.01391	📌
M Mistral AI	Mistral Large	33k	\$0.000008	\$0.000024	\$0.01391	\$0.01391	📌
G Google	Gemini 1.5 Pro	1m	\$0.000007	\$0.000021	\$0.01217	\$0.01217	📌
🌀 openai	GPT-4o	128k	\$0.000005	\$0.000015	\$0.00869	\$0.00869	📌
AI Anthropic	Claude 3 Sonnet	200k	\$0.000003	\$0.000015	\$0.00861	\$0.00861	📌
🍎 Cohere	Command-R+	128k	\$0.000003	\$0.000015	\$0.00861	\$0.00861	📌
∞ Meta	Llama 3.1 (405B)	128k	\$0.000003	\$0.000009	\$0.00521	\$0.00521	📌
M Mistral AI	Mistral Medium	33k	\$0.0000027	\$0.0000081	\$0.00469	\$0.00469	📌
M Mistral AI	Mistral Small	33k	\$0.0000015	\$0.0000045	\$0.00261	\$0.00261	📌
∞ Meta	Databricks Llama 3.1 (70B)	128K	\$0.000001	\$0.000003	\$0.00174	\$0.00174	📌
AI Anthropic	Claude Instant	100k	\$0.0000008	\$0.0000024	\$0.00139	\$0.00139	📌
🌀 openai	GPT-3.5 Turbo Instruct	4k	\$0.0000015	\$0.000002	\$0.00119	\$0.00119	📌
🍎 Cohere	Command-R	128k	\$0.0000005	\$0.0000015	\$0.00087	\$0.00087	📌
G Google	Gemini 1.0 Pro	33k	\$0.0000005	\$0.0000015	\$0.00087	\$0.00087	📌

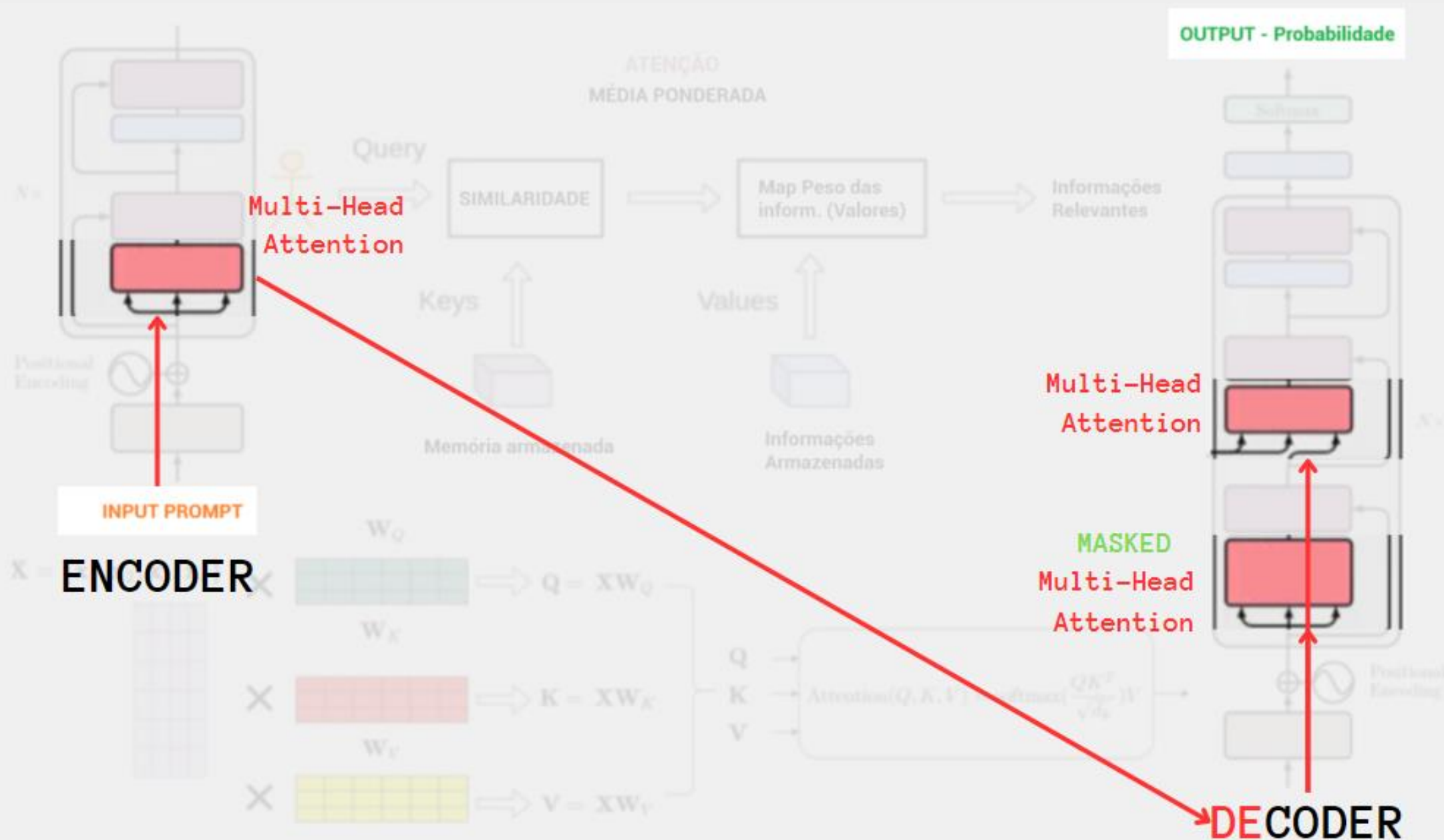
<https://www.tribalscale.com/our-services/llm-calculator>



Eu	Desenvolvo	em	Delphi	Utilizando
Pos = 1	Pos = 2	Pos = 3	Pos = 4	Pos = 5
e_0	e_1	e_2	e_3	e_4
p_0	p_1	p_2	p_3	p_4
0.32	0.23	0.12	-0.9	0.55
0.11	0.10	0.21	0.32	0.24
0.89	0.19	0.03	0.81	0.22
0.22	-0.2	0.6	0.37	0.00
0.03	0.91	0.08	0.62	0.58
-0.1	0.52	0.35	0.75	0.05
0.32	0.22	0.46	0.32	0.03
-0.01	0.70	-0.1	-0.1	-0.3
0.21	0.62	-0.21	-0.2	0.75
0.11	-0.32	-0.5	0.0	0.01

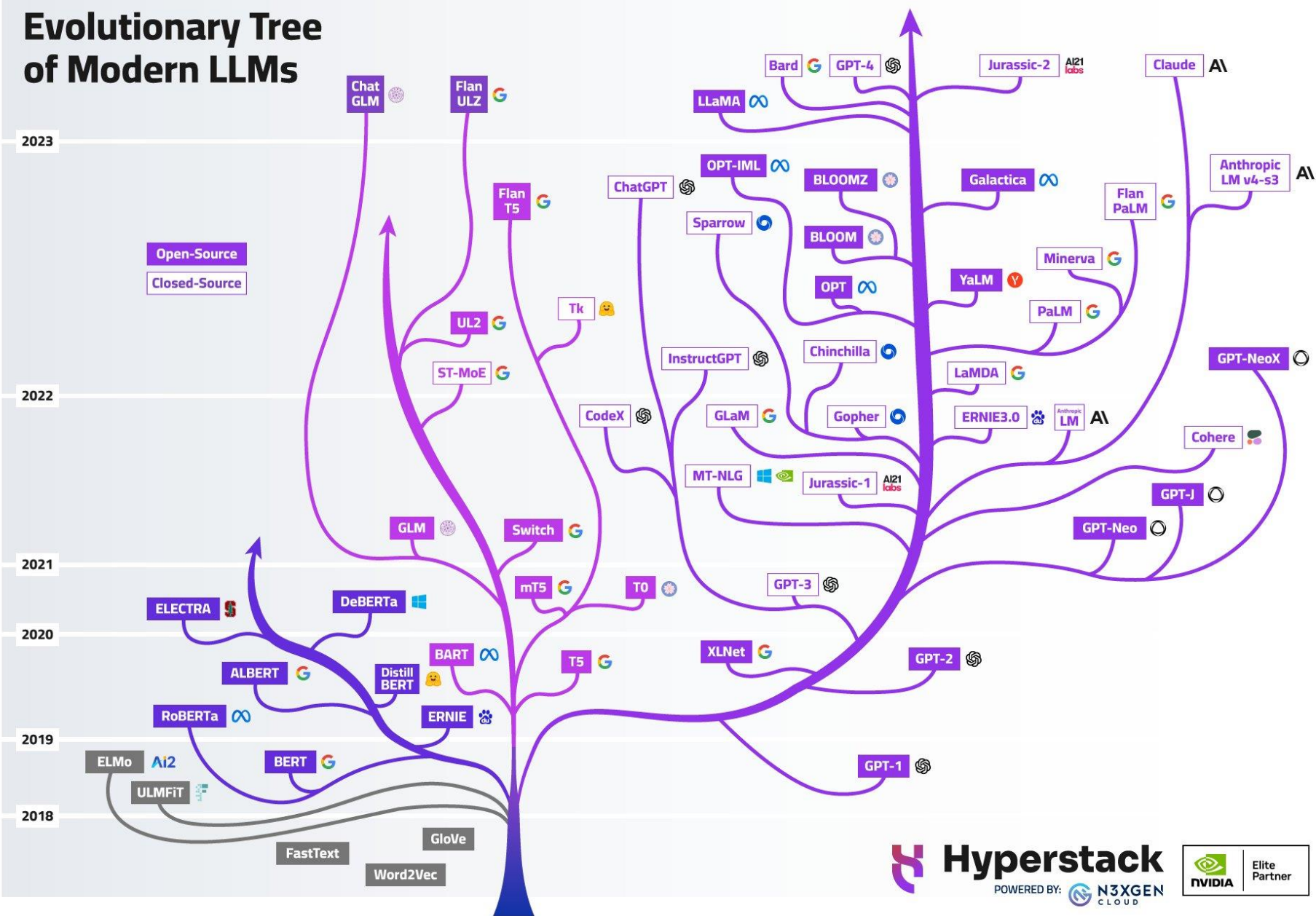








Evolutionary Tree of Modern LLMs



[Tasks](#) [Libraries](#) [Datasets](#) [Languages](#) [Licenses](#) [Other](#)

Multimodal

[Image-Text-to-Text](#)
[Visual Question Answering](#)
[Document Question Answering](#)
[Video-Text-to-Text](#)


Computer Vision





[Depth Estimation](#)
[Image Classification](#)
[Object Detection](#)
[Image Segmentation](#)
[Text-to-Image](#)
[Image-to-Text](#)
[Image-to-Image](#)
[Image-to-Video](#)
[Unconditional Image Generation](#)
[Video Classification](#)
[Text-to-Video](#)
[Zero-Shot Image Classification](#)
[Mask Generation](#)
[Zero-Shot Object Detection](#)
[Text-to-3D](#)
[Image-to-3D](#)
[Image Feature Extraction](#)


Natural Language Processing




[Text Classification](#)
[Token Classification](#)
[Table Question Answering](#)
[Question Answering](#)
[Zero-Shot Classification](#)
[Translation](#)


Models 901,823




 **black-forest-labs/FLUX.1-dev**


 Text-to-Image • Updated 19 days ago •  688k •  •  3.69k




 **THUDM/CogVideoX-5b**


 Text-to-Video • Updated 6 days ago •  14.3k •  311




 **meta-llama/Meta-Llama-3.1-8B-Instruct**


 Text Generation • Updated 14 days ago •  2.88M •  •  2.21k




 **CausalLM/miniG**


 Text Generation • Updated 3 days ago •  275k •  177




 **CohereForAI/c4ai-command-r-08-2024**


 Text Generation • Updated about 11 hours ago •  792 •  108


 **CohereForAI/c4ai-command-r-plus-08-2024**




 Text Generation • Updated 4 days ago •  387 •  91


 **yifeihu/TB-OCR-preview-0.1**





 Image-Text-to-Text • Updated 5 days ago •  675 •  88


 **microsoft/Phi-3.5-vision-instruct**




 **Qwen/Qwen2-VL-7B-Instruct**


 Image-Text-to-Text • Updated about 22 hours ago •  20.6k •  248




 **black-forest-labs/FLUX.1-schnell**


 Text-to-Image • Updated 19 days ago •  1.88M •  •  2.07k





 **Shakker-Labs/AWPortrait-FL**


 Text-to-Image • Updated about 21 hours ago •  1.57k •  140




 **Shakker-Labs/FLUX.1-dev-LoRA-blended-realistic**


 Text-to-Image • Updated 3 days ago •  1.41k •  •  116


 **ByteDance/Hyper-SD**


 Text-to-Image • Updated 7 days ago •  115k •  •  785

 **Qwen/Qwen2-VL-2B-Instruct**

 Image-Text-to-Text • Updated about 8 hours ago •  14.7k •  90

 **gpt-omni/mini-omni**

Updated 5 days ago •  80

 **microsoft/Phi-3.5-mini-instruct**

O que é Engenharia de PROMPT

É a arte de **formular instruções claras** para **obter respostas precisas** de modelos de IA, combinando habilidades linguísticas e técnicas específicas.

A chave é fornecer o máximo de contexto possível em um prompt bem estruturado.



Porque eu preciso de engenharia de PROMPT?





como calibrar pneu de trem?




Calibrar os pneus de um trem é um processo importante para garantir a segurança e a eficiência do veículo ferroviário. O processo de calibração dos pneus de um trem é semelhante ao de um veículo comum, mas existem algumas diferenças importantes a serem observadas.

1. Verifique a pressão recomendada: Antes de começar a calibrar os pneus, verifique a pressão recomendada pelo fabricante do trem. Isso geralmente pode ser encontrado no manual do proprietário ou nas especificações técnicas do trem.
2. Prepare os equipamentos de calibração: Você precisará de um medidor de pressão de pneus e um compressor de ar para calibrar os pneus do trem. Certifique-se de que o medidor de pressão esteja calibrado corretamente antes de começar a medição.





3. Remova as tampas de válvula: Remova as tampas de válvula dos pneus do trem. Essas tampas são geralmente encontradas na parte superior do pneu e devem ser removidas com uma chave de válvula.
4. Conecte o medidor de pressão: Conecte o medidor de pressão à válvula do pneu e pressione-o firmemente para garantir uma conexão segura.
5. Meça a pressão dos pneus: Verifique a pressão dos pneus do trem e anote a leitura. Repita o processo em cada um dos pneus para garantir que todos estejam calibrados corretamente.
6. Adicione ou libere ar: Se a pressão estiver abaixo do recomendado, adicione ar aos pneus usando um compressor de ar. Se a pressão estiver acima do recomendado, libere ar dos pneus usando a válvula de liberação.
7. Verifique novamente a pressão: Após adicionar ou liberar ar, verifique novamente a pressão dos pneus e ajuste conforme necessário.
8. Coloque as tampas de válvula de volta: Depois de calibrar todos os pneus do trem, coloque as tampas de válv  de volta.



É importante lembrar que a calibração dos pneus deve ser feita regularmente para garantir que o trem esteja operando de maneira segura e eficiente.



Principais Técnicas de Engenharia de Prompt

Zero Shot

Essa técnica envolve fornecer ao modelo uma tarefa que ele não viu durante seu treinamento.

Ela testa a capacidade do modelo de generalizar e produzir saídas relevantes sem depender de exemplos anteriores.



Zero Shot - Exemplo

##PROMPT

Dado o seguinte texto:

“Hoje foi um dia incrível! Acordei cedo, tomei um café e fui programar.
Logo na primeira tarefa, resolvi um BUG milenar!
Finalmente o cliente vai me dar paz rs.
Estou me sentindo o ninja da programação!”

##RESPOSTA

Classifique o sentimento do texto em uma escala de 1 a 5, sendo 1 muito negativo e 5 muito positivo, sem usar nenhum modelo de linguagem treinado para essa tarefa específica.



Zero Shot - Exemplo

Análise de Sentimento

Para classificar o sentimento do texto apresentado, vamos considerar os elementos emocionais e as expressões utilizadas.

Elementos Positivos

1. **Entusiasmo:** O autor expressa um sentimento de alegria ao descrever seu dia como "incrível".
2. **Realização:** A resolução de um "BUG milenar" é mencionada com um tom de conquista, o que indica satisfação e orgulho.
3. **Humor:** A frase "Finalmente o cliente vai me dar paz rs" sugere um alívio e uma leveza na situação, mostrando que o autor está lidando bem com a pressão.
4. **Confiança:** A expressão "me sentindo o ninja da programação" indica uma autoconfiança elevada e um sentimento de competência.

Elementos Negativos

Não há elementos claramente negativos no texto. A única menção que poderia ser interpretada como negativa é a referência ao "BUG milenar", mas isso é contrabalançado pela alegria de ter resolvido o problema.



Zero Shot - Exemplo

Classificação Final

Com base na análise acima, o sentimento do texto pode ser classificado como **5** na escala de 1 a 5, onde 1 é muito negativo e 5 é muito positivo. O texto reflete um estado de espírito otimista e satisfeito, típico de um dia produtivo e gratificante.



Principais Técnicas de Engenharia de Prompt

Few-shot

A técnica de Few-shot prompting, ou aprendizado em contexto, envolve fornecer exemplos que ajudam a guiar a resposta do modelo.



Few-shot - Exemplo

Prompt

Instruções: Abaixo estão alguns exemplos de sentimentos expressos em textos. Depois dos exemplos, forneça um novo texto e classifique o sentimento em uma escala de 1 a 5, onde 1 é muito negativo e 5 é muito positivo.

Exemplo 1:

- **Texto:** "Hoje foi um dia muito difícil. Enfrentei muitos problemas no trabalho e me senti sobrecarregado."
- **Sentimento:** 2

Exemplo 2:

- **Texto:** "Tive um ótimo final de semana! Fui ao parque com amigos e nos divertimos muito."
- **Sentimento:** 5

Exemplo 3:

- **Texto:** "Estou me sentindo um pouco ansioso com a apresentação de amanhã, mas acredito que tudo vai dar certo."
- **Sentimento:** 3



Few-shot - Exemplo

Novo Texto: "Finalmente consegui terminar meu projeto! Estou tão feliz e aliviado, mal posso esperar para mostrar ao meu chefe."

Sentimento: 5



Principais Técnicas de Engenharia de Prompt

Chain-of-Thought

A técnica de Chain-of-Thought (Cadeia de Pensamento) envolve guiar o modelo a pensar passo a passo, detalhando seu raciocínio antes de chegar a uma conclusão.

Essa técnica é útil para resolver problemas complexos ou para classificar sentimentos de forma mais precisa.



Chain-of-Thought - Exemplo

Prompt

Instruções: Leia o texto abaixo e, em seguida, siga os passos para analisar o sentimento. Explique seu raciocínio antes de fornecer a classificação final em uma escala de 1 a 5, onde 1 é muito negativo e 5 é muito positivo.

Texto: "Hoje foi um dia desafiador. Tive que lidar com várias tarefas ao mesmo tempo e, no final, me senti exausto. No entanto, consegui completar tudo e isso me deixou um pouco satisfeito."

Passo a Passo

1. **Identificação de Emoções:** Comece identificando as emoções expressas no texto. O autor menciona um "dia desafiador" e "me senti exausto", o que indica um sentimento negativo inicial.
2. **Análise de Conflito:** Apesar da exaustão, o autor também menciona que "consegui completar tudo", o que sugere um sentimento de realização. Essa é uma emoção positiva que contrabalança a exaustão.
3. **Peso das Emoções:** Avalie a intensidade das emoções. A exaustão parece ser forte, mas a satisfação por ter completado as tarefas também é significativa. O autor se sente "um pouco satisfeito", o que é um sentimento positivo, mas não tão forte quanto o negativo.



Chain-of-Thought - Exemplo

Classificação Final: Considerando que o texto expressa tanto um sentimento negativo (exaustão) quanto um positivo (satisfação), mas que a exaustão parece dominar, eu classificaria o sentimento como **3**. Isso indica um estado neutro, onde as emoções se equilibram.

Sentimento: 3



Linguagem para PROMPT.

MARKDOWN

<https://www.markdownguide.org/cheat-sheet/>

LLM's Aprenderam com base em textos e replicam isso.

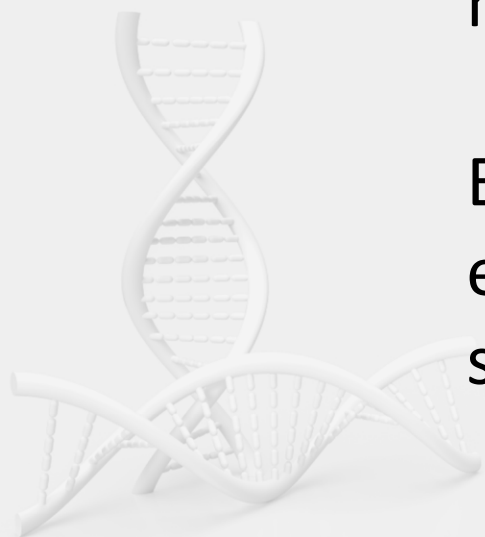


Linguagem para PROMPT.

<XML>

O uso de XML em prompts não apenas melhora a organização e a clareza das instruções, mas também aumenta a precisão e a flexibilidade na interação com modelos de linguagem.

Esses benefícios tornam o XML uma escolha ideal para estruturar prompts complexos e garantir que as saídas sejam de alta qualidade e facilmente manipuláveis.



Linguagem para PROMPT.

{{Variáveis}}

Utilizar variáveis facilita a sobreposição de respostas.



Linguagem para PROMPT.

```` CÓDIGO ````

Um prompt com instruções de código caracteriza o formato de saída.

##Exemplo de código

```

unit uCliente;

interface

Uses System.SysUtils;

type

  TCliente = class

  ...  
````



Como criar meus prompts?

groqcloud

<https://console.groq.com/playground>

ANTHROPIC

<https://console.anthropic.com>



<https://platform.openai.com/playground/>



Exemplos de PROMPT







Consultas Dinâmicas



Questão:



Gerar Resultado

DATA_VENDA	NOME	TOTAL_REAL
▶ 25/07/2024	João Silva	9650

SQL Gerado

```
SELECT VDA.DATA_VENDA, VEN.NOME, VDA.TOTAL_REAL FROM Vendas VDA INNER JOIN
Vendedores VEN ON VDA.ID_VENDEDOR = VEN.ID_VENDEDOR WHERE VDA.DATA_VENDA
BETWEEN '2024-01-01' AND '2024-12-31' ORDER BY VDA.TOTAL_REAL DESC LIMIT 1
```

Executar SQL Novamente

Modelo

Temperatura





Consultas Dinâmicas



Questão:

Gerar Resultado

NOME	Total_Estoque
▶ Acessórios	142
Computadores	132
Escritórios	73
Gaming	27
Impressoras	108
Notebooks	30
Periféricos	130
Smartphones	82
Software	136

SQL Gerado

```
SELECT CAT.NOME, SUM(PRD.ESTOQUE) AS Total_Estoque FROM Produtos PRD INNER JOIN
Produtos_Categorias PCT ON PRD.ID_PRODUTO = PCT.ID_PRODUTO INNER JOIN Categorias CAT
ON PCT.ID_CATEGORIA = CAT.ID_CATEGORIA GROUP BY CAT.NOME
```

Executar SQL Novamente


Modelo

Temperatura

Construtor de Consultas

SQL

Consultas Dinâmicas



Questão:

apresente as vendas por produtos, listando o nome do produto e quantidade vendida

Gerar Resultado

Mes

▶ 2023-01

2023-02

2023-03

2023-04

SQL Gerado

SELECT strftime('%Y-%m', VDA.DATA_VENDA) AS Mes, SUM(VDA.TOTAL_REAL) AS TotalVendas
FROM Vendas VDA WHERE strftime('%Y', VDA.DATA_VENDA) = '2023' GROUP BY strftime('%Y-%m',
VDA.DATA_VENDA) ORDER BY Mes;

PowerGroq (Chart)

Modelo

llama3-8b-8192

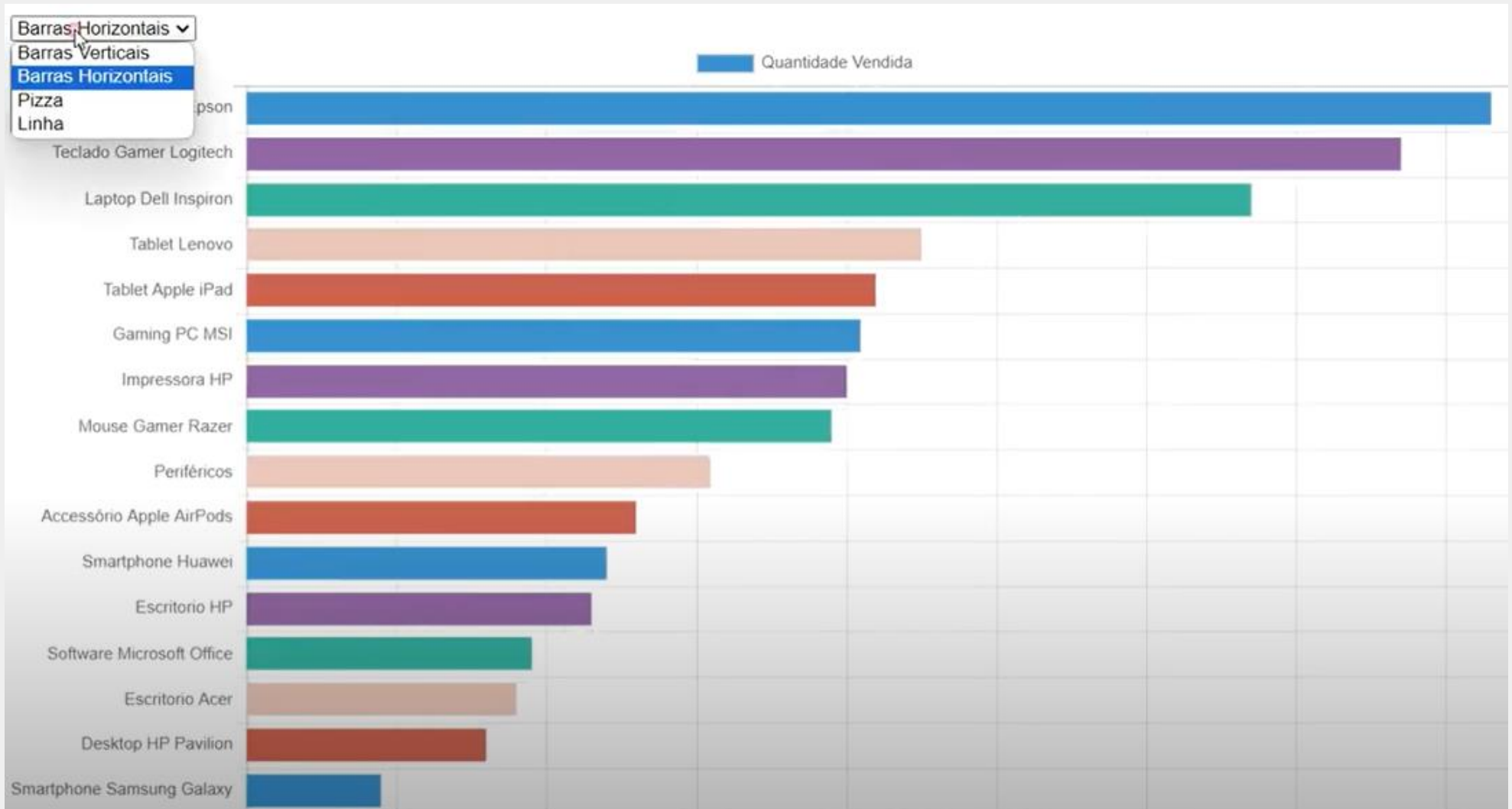
Temperatura

0.03

GROQ APIKey

#####





<https://lmarena.ai>



Embarcadero Conference 2024


Inovação faz parte do nosso DNA!





Quer me ver na
#ECON25?
Acesse o QRCode
e avalie minha palestra!




Carlos Eduardo Paulino

 [@DelphiCleanCode](https://www.instagram.com/DelphiCleanCode)

 [linkedindopalestrante](https://www.linkedin.com/in/linkedindopalestrante)

 delphicleancode@gmail.com

 [\(14\) 9 9195 4605](tel:(14)991954605)

