

Machine Learning, Deep Learning e IA Generativa com o Watsonx na prática

Aula 1: HackTruck

Jorge D. B. Chagas
Senior AI Engineer
Client Engineering – Watsonx



LTU robotic vehicle performs well at IGVC

Release Date: June 16, 2014

Lawrence Technological University's robotic vehicle, iWheels 2, finished fifth in Inter-Operability Profiles (IOP) challenge and sixth in the Auto-Nav Challenge at the 22nd



Representing Lawrence Tech at the Intelligent Ground Vehicle Competition were (left to right) Gabriel Laguardia de lima, Adjunct Instructor Jonathan Ruszala, Gordon Stein, Jorge Damiao Barbosa das Chagas, and Professor CJ Chung.

annual Intelligent Ground Vehicle Competition (IGVC) held at Oakland University June 6-9.

IGVC promotes the research and development of automated and intelligent vehicles that can have both civilian and military applications. Forty-nine teams from around the nation and as far away as India and Japan entered the prestigious international competition.

Computer science undergraduate student Gordon Stein, and two students from Brazil, Gabriel Laguardia de lima and Jorge Damiao Barbosa das Chagas, competed for LTU and won \$1,000 in prizes. Professor CJ Chung and Adjunct Instructor Jonathan Ruszala were the faculty co-advisors. Two former IGVC participants Jonathan Nabozny and Christopher Kawatsu, assisted as technical advisors.

Chung said the fifth place award in IOP challenge is significant, because it involved the key software technologies for smart cars of the future that will connect to each other and the transportation infrastructure.

Sponsors include the Joint Project Office for Robotic Systems of the U.S. Army, the Joint Ground Robotics Enterprise of the Department of Defense, TARDEC, and the Michigan chapter of the National Defense Industrial Association, Oakland University, and the Association for Unmanned Vehicle Systems International (both the foundation and the Great Lakes chapter).

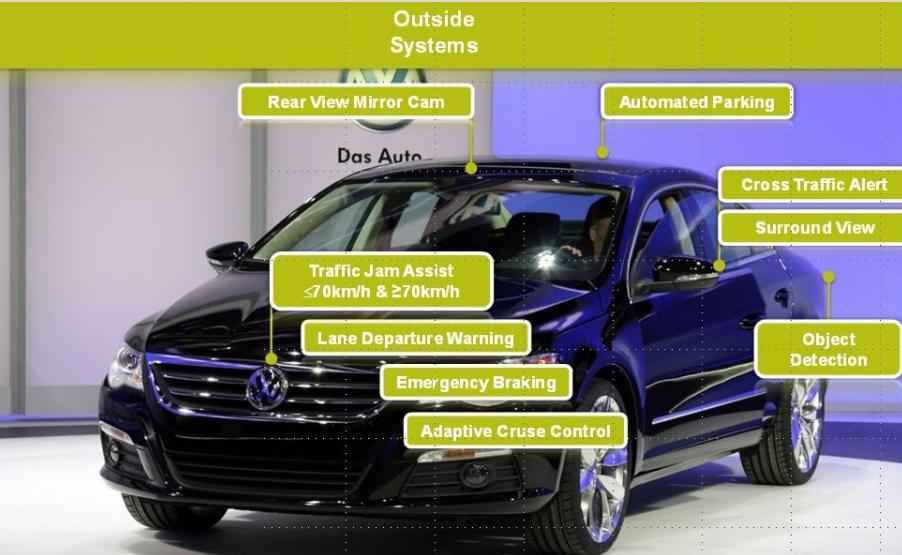
Corporate sponsors include Lockheed Martin, Continental, Valeo, Magna, MathWorks, and General Dynamics Land Systems.

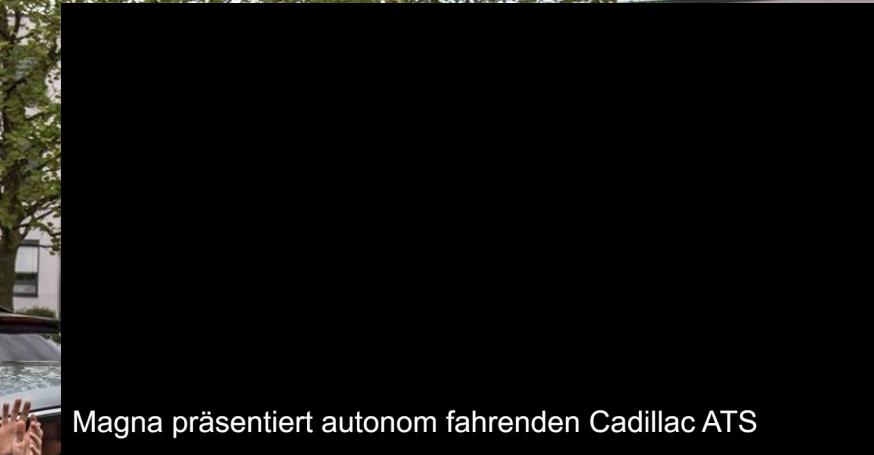


Autonomous Driving Vehicle

IBM

Pattern recognition for navigation optimization, control systems and object detection





Magna präsentiert autonom fahrenden Cadillac ATS



PROJETOS DESENVOLVIDOS NA EMBRAER





Fraud Detection



Credit Card Transactions



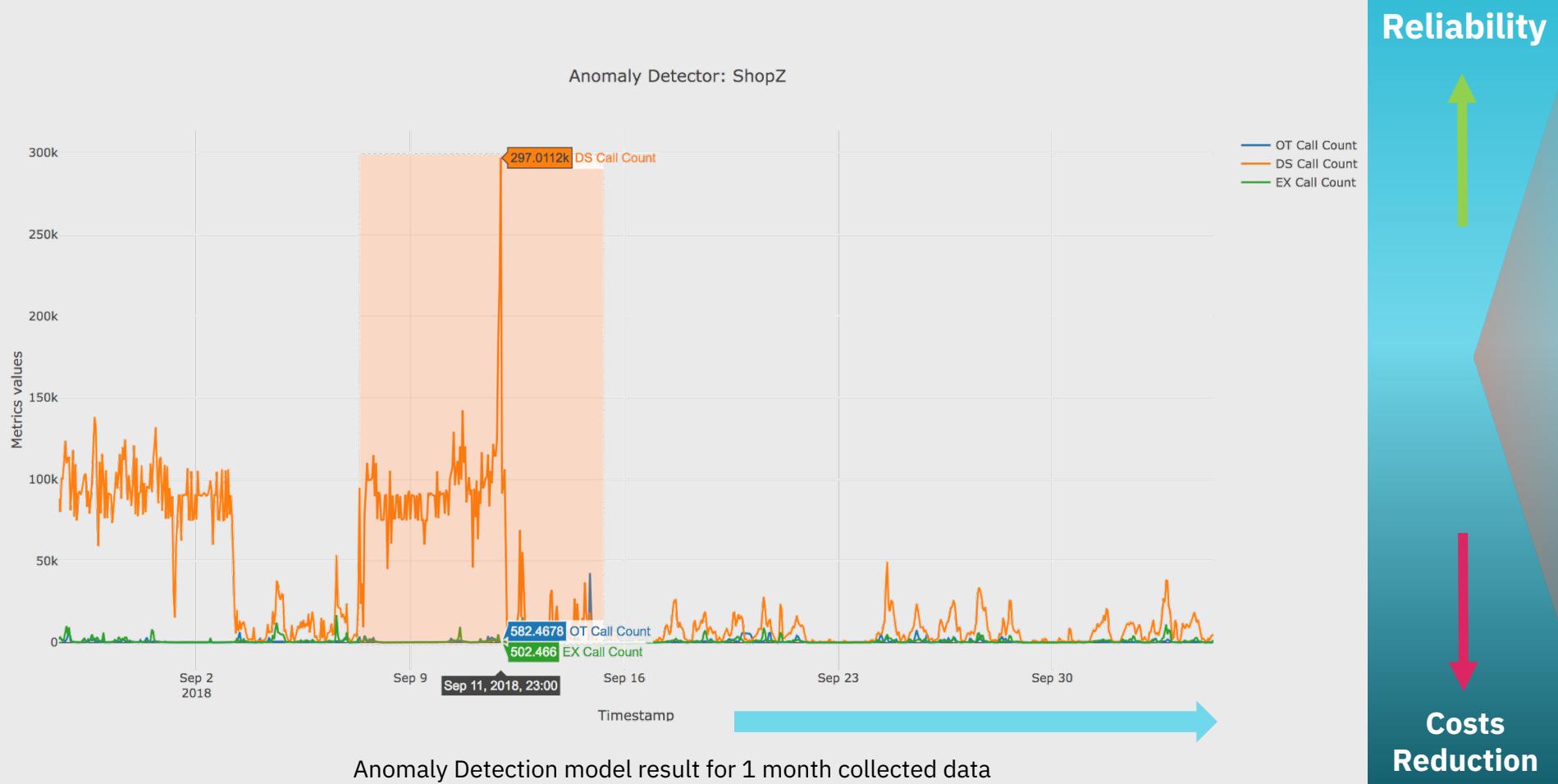
Previous Purchases



Profile Analysis



Automatically detect anomalous behavior



O que é **BIG DATA**?



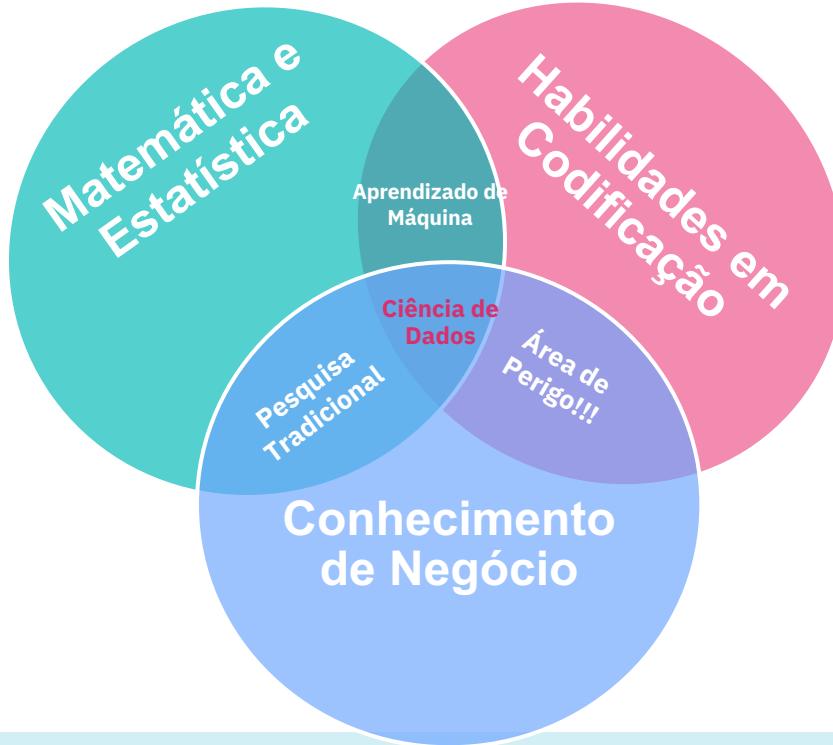
BIG DATA



Big Data é um termo adotado para representar atividades analíticas em um espectro mais amplo do dado, em sua maioria, para operações que envolvem um grande volume de informações.

Referimos sempre ao termo como uma agregação dos V's: **Volume**, **Veracidade**, **Variedade** e **Velocidade**. Recentemente, refere-se Big Data as atividades analíticas que englobam Volume, Valor, Veracidade, Visualização, Variedade, Velocidade e Viralidade

O que é Ciência de Dados?



 **Josh Wills**
@josh_wills [Seguir](#)

Data Scientist (n.): Person who is better at statistics than any software engineer and better at software engineering than any statistician.

Traducir del inglés

9:55 - 3 may. 2012

1.686 Retweets 1.417 Me gusta 

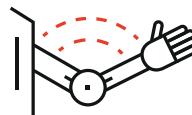
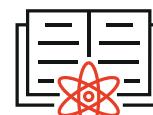
51 1,7K 1,4K

Ciência de dados é um área interdisciplinar que envolve o estudo dos dados e informações inerentes ao negócio, visando a extração de conhecimento, detecção de padrões, processamento, otimização, automação, transformação e análise de dados. É uma área que envolve as disciplinas de matemática, estatística, computação e conhecimento do negócio.

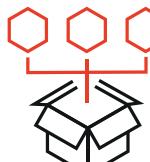
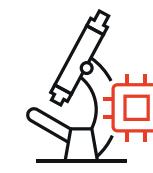
**Quais problemas podem ser resolvidos
utilizando Ciência de Dados?**



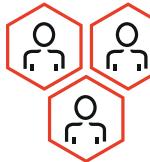
COGNITION

SENSORIMOTOR
SKILLSAI
KNOWLEDGE

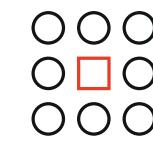
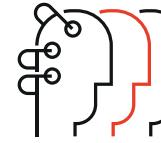
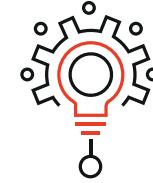
EXPERT SYSTEM

KNOWLEDGE
REPRESENTATIONAUTOMATED
PLANNINGCOMPUTATIONAL
INTELLIGENCE

DEEP LEARNING

MULTI-AGENT
SYSTEM

AI APPLICATIONS

PATTERN
RECOGNITIONINTELLIGENT
AGENTSARTIFICIAL
NEURAL
NETWORKHUMAN-COMPUTER
INTERACTION

AI RESEARCH

EMERGENT
BEHAVIOR

Prever o que irá acontecer...

- ...baseado em **dados históricos**.



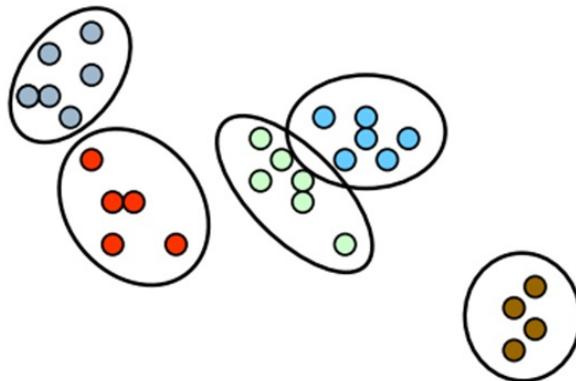
- ...baseado em **tendência** de dados antigos.

EXEMPLOS

- **Quantos** incidentes espera-se que sejam criados na **próxima segunda?**
- **Quanto tempo** leva-se geralmente para criar **esse tipo de incidente?** (Problemas de estimativa)
- **Prever a necessidade computacional** necessária em um futuro próximo.
- **Previsão de falhas**

Categorizar coisas automaticamente...

- ...através de **reconhecimento de padrões**.



EXEMPLOS

- **Agrupar itens similares**
 - Agrupamento de filas sendo trabalhadas em incidentes de problemas similares;
 - Classificação de clientes;
 - Reconhecimento de padrões.

Relacionamento entre itens...

- ...Através de **análise de redes**

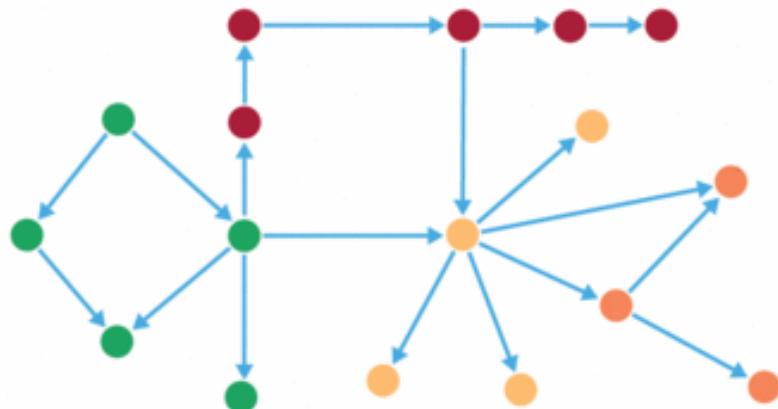


EXEMPLOS

- **Agrupamento de itens ou grupos que trabalham em conjunto**
 - Relacionamento de clientes ou times para recomendação
 - Entendimento do **impacto** entre um time ou outro na cadeia produtiva

Dependência entre coisas

- ...através de **redes Bayesianas**



EXEMPLOS

- Qual o **impacto** que **um ítem de configuração** pode ter? (Análise de suporte ao cliente)
- Quais **itens** pode ter **causado** uma falha em um determinado componente? (Análise de causa raiz).

Fornecer sugestões ou recomendações

EXEMPLOS

- ...através de **sistemas de recomendação**



- Recomendar soluções de resolução de problemas;
- Sugerir itens similares em um processo de vendas;

Detectar anormalidades

- ...através de algoritmos de **detecção de anomalias**



EXEMPLOS

- Detectar **anomalias** em **dados de monitoramento** ou de **desempenho**:
 - Auxilia na prevenção de falhas, manutenção preventiva, etc.

Tipos de algorítmos de aprendizado de máquina



Treinados com supervisão humana ou por rótulos/anotação nos dados

- Supervisionados
- Não-supervisionados
- Aprendizado por reforço

Podem aprender incrementalmente ou em tempo-real

**Online vs.
Batch Learning**

Como realiza o processo de generalização

- Baseado em instância
- Aprendizado baseado em modelo

Tipos de algoritmos de aprendizado de máquina



• ALGORÍTMOS SUPERVISIONADOS

- Há existência de uma variável alvo como rótulo ou sinal do que se deseja prever, classificar, recomendar, etc.
- Tem-se conhecimento prévio nos dados de treinamento sobre a “resposta”, sendo possível validar o resultado entre o valor observado e o previsto.

• ALGORÍTMOS NÃO-SUPERVISIONADOS

- Não há nenhum rótulo ou anotação sobre os dados.
- Nesse caso, procura-se por alguma estrutura ou padrão no dado de acordo com algum critério ou característica específica.

Exemplo de problemas: Supervisionado

Classificação de câncer de pele



SPAM FILTERING



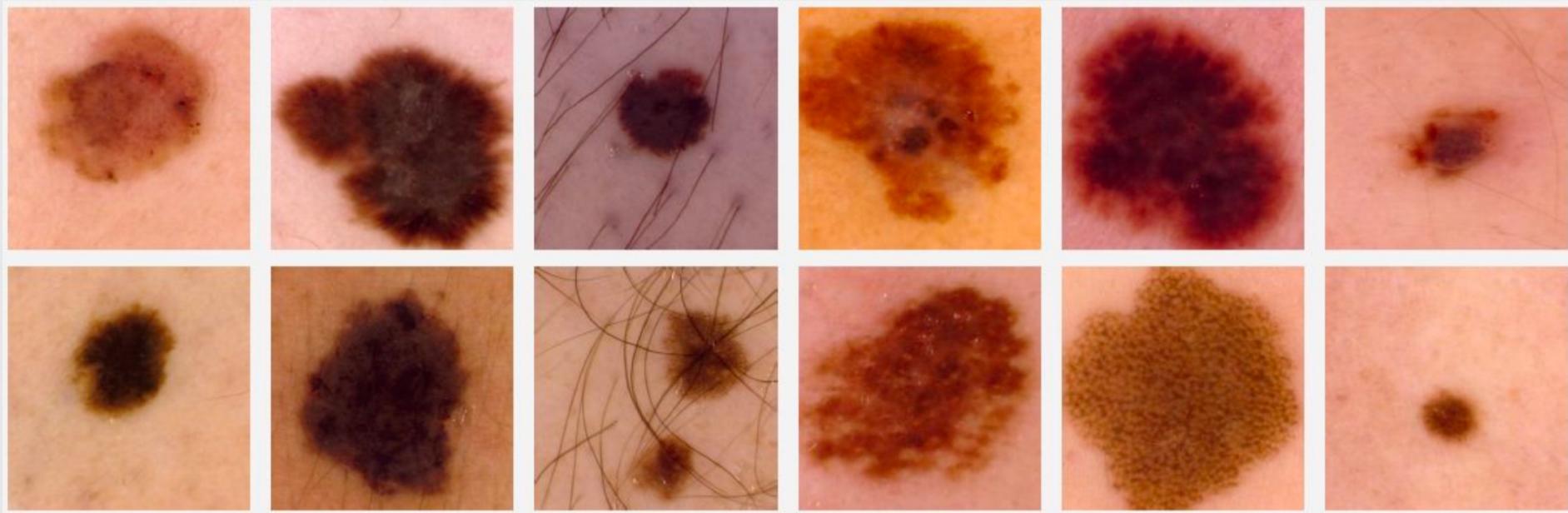
Bad Cures fast and effective! - Canadian *** Pharmacy #1
Internet Inline Drugstore Viagra Cheap Our price \$1.99

...

Good Interested in your research on graphical models - Dear Prof., I have read some of your papers on probabilistic graphical models. Because I ...

Exemplo de problemas: Supervisionado

Classificação de câncer de pele

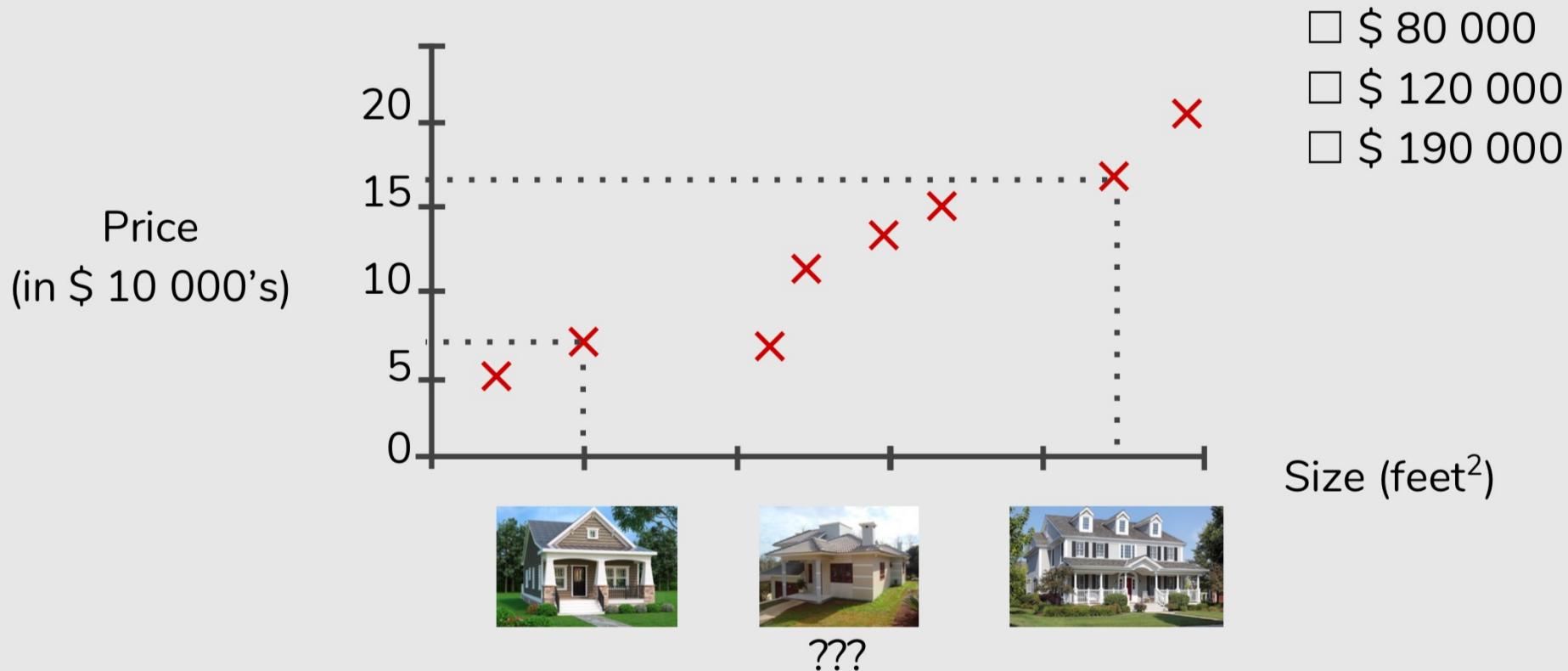


Melanomas (linha superior) e lesões **benignas** (linha inferior)

Exemplo de problemas: Supervisionado

Estimativa de preços de imóveis

Problemas de estimativa: Regressão



Tipos de algoritmos de aprendizado de máquina

Supervisionados

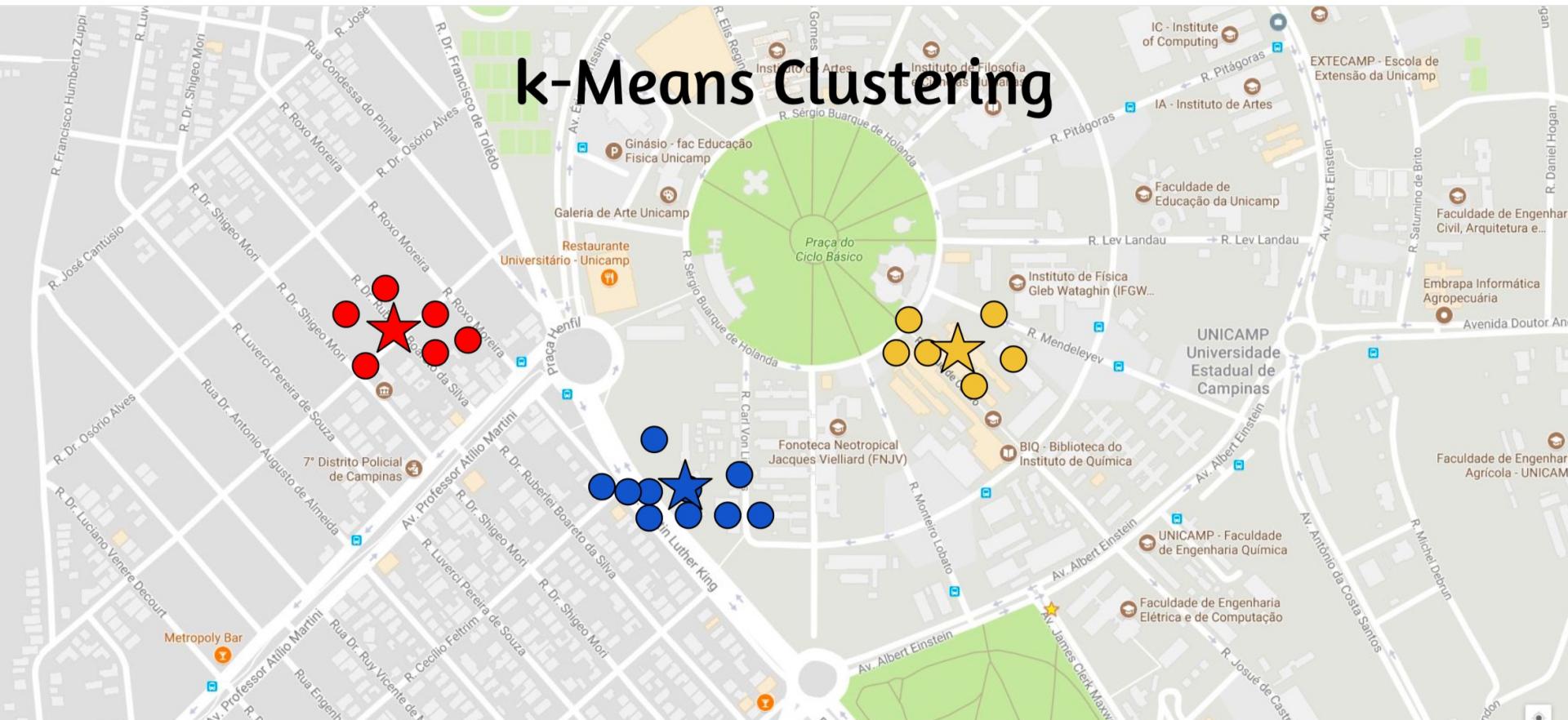


- Naïve Bayes;
- Linear Regression;
- Logistic Regression;
- k-Nearest Neighbors;
- Support Vector Machines (SVMs);
- Neural Networks;
- Decision Trees and Random Forests.

Exemplo de problemas: Não-Supervisionado

Agrupamento de restaurantes por tipo de cardápio

k-Means Clustering

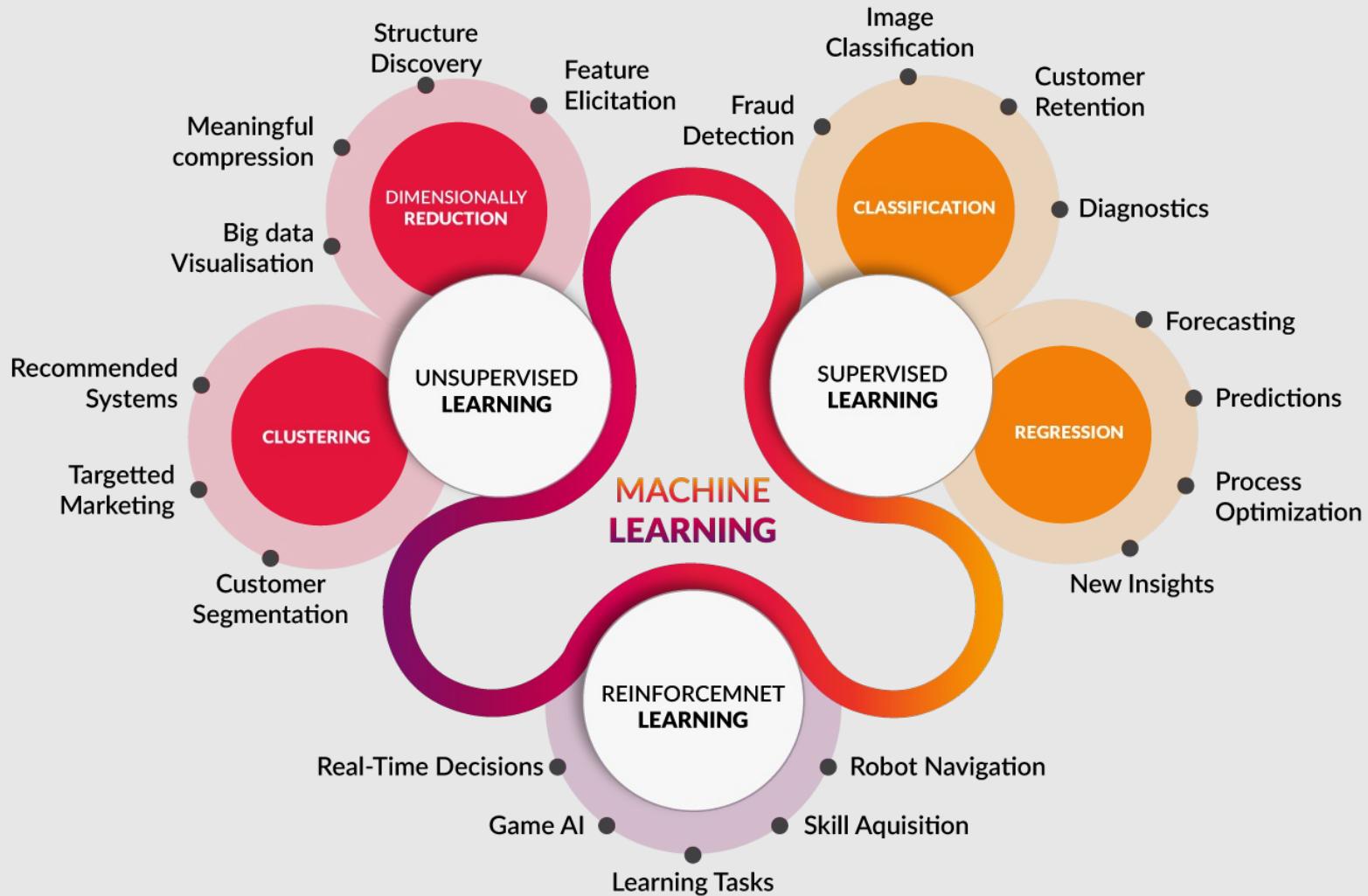


Tipos de algoritmos de aprendizado de máquina

Não-Supervisionados

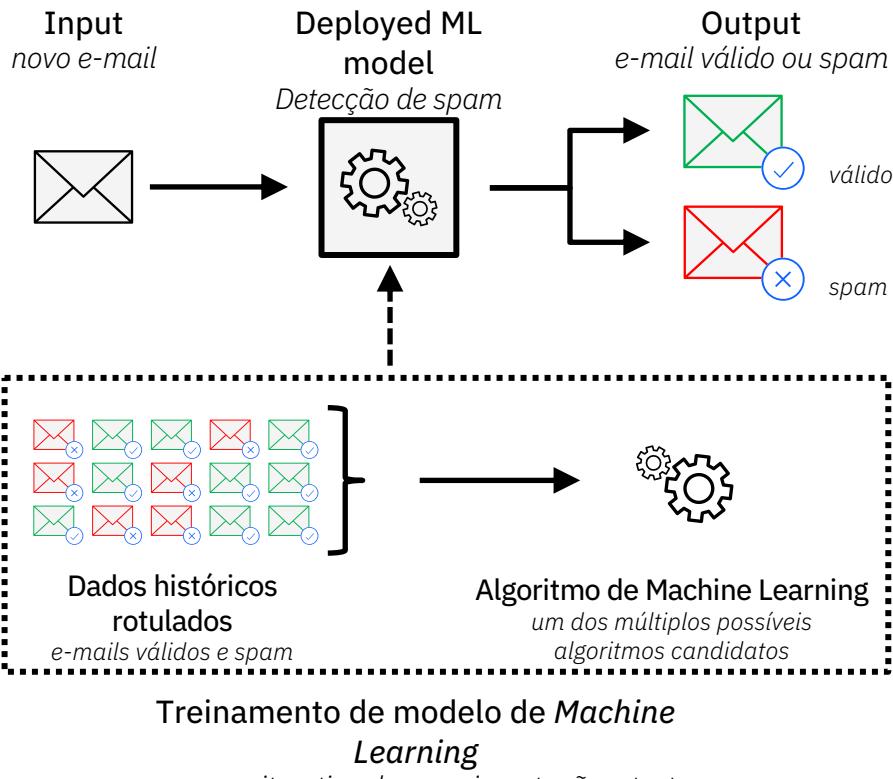


- k-Means
- Hierarchical Cluster Analysis (HCA)
- Expectation Maximization
- Principal Component Analysis (PCA)
- Kernel PCA
- t-distributed Stochastic Neighbor Embedding (t-SNE)



O que é um modelo de machine learning (ML)?

Exemplo: Detecção de spam em e-mail



Tipos de Machine Learning (ML)

Modelos de classificação

Os modelos de classificação atribuem rótulos às entradas do modelo ou os atribuem a categorias específicas.

Os casos de uso comuns incluem::

- Detecção de fraude: prever se uma transação é fraudulenta com base em padrões nos dados
- Análise de sentimento: classificar o texto como positivo, negativo ou neutro
- Diagnóstico médico: atribuir um rótulo de doença ao caso de um paciente, com base nos sintomas e no histórico médico
- Reconhecimento de imagem: reconhecer objetos ou identificar pessoas com base em características e traços visíveis

Tipos de Machine Learning (ML)

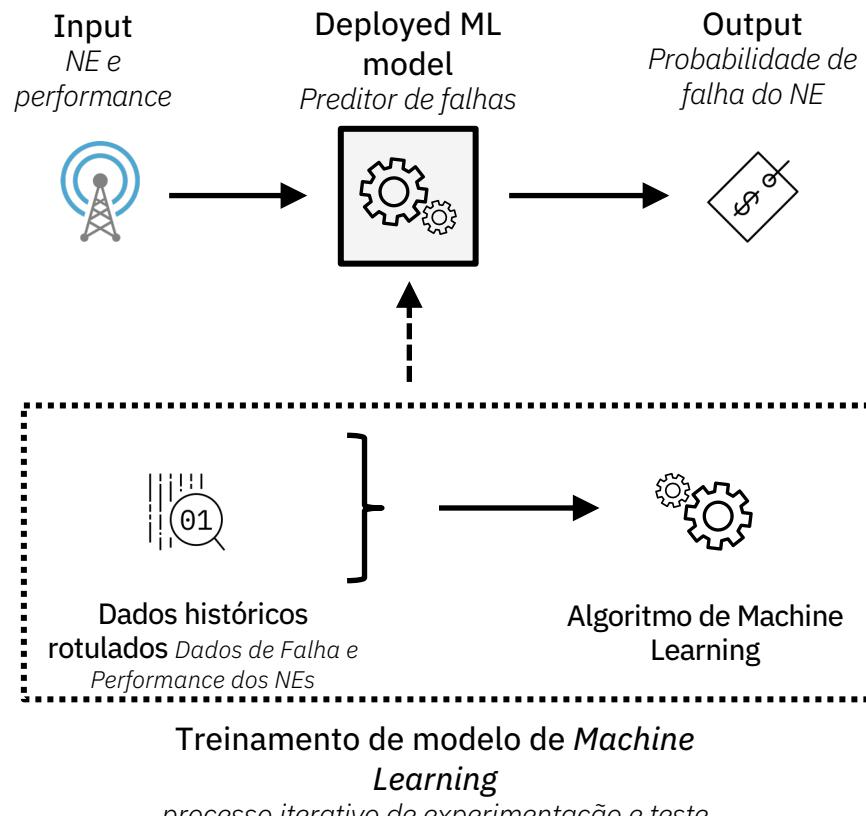
Modelos de Regressão

Modelos de regressão fazem previsões com base nas entradas do modelo.

Os casos de uso comuns incluem:

- Análise do mercado de ações: previsão de preços de títulos com base em dados históricos ou eventos de notícias
- Vendas: previsão com base em dados históricos ou tendências de mercado
- Cuidados de saúde: prever os resultados dos pacientes com base em fatores como idade, sexo, histórico médico ou planos de tratamento
- Análise do comportamento do cliente: preveja padrões de compra futuros do cliente com base em dados demográficos, histórico de compras anteriores e campanhas publicitárias
- Predição de falhas na rede: com base em dados de falha e performance da rede.

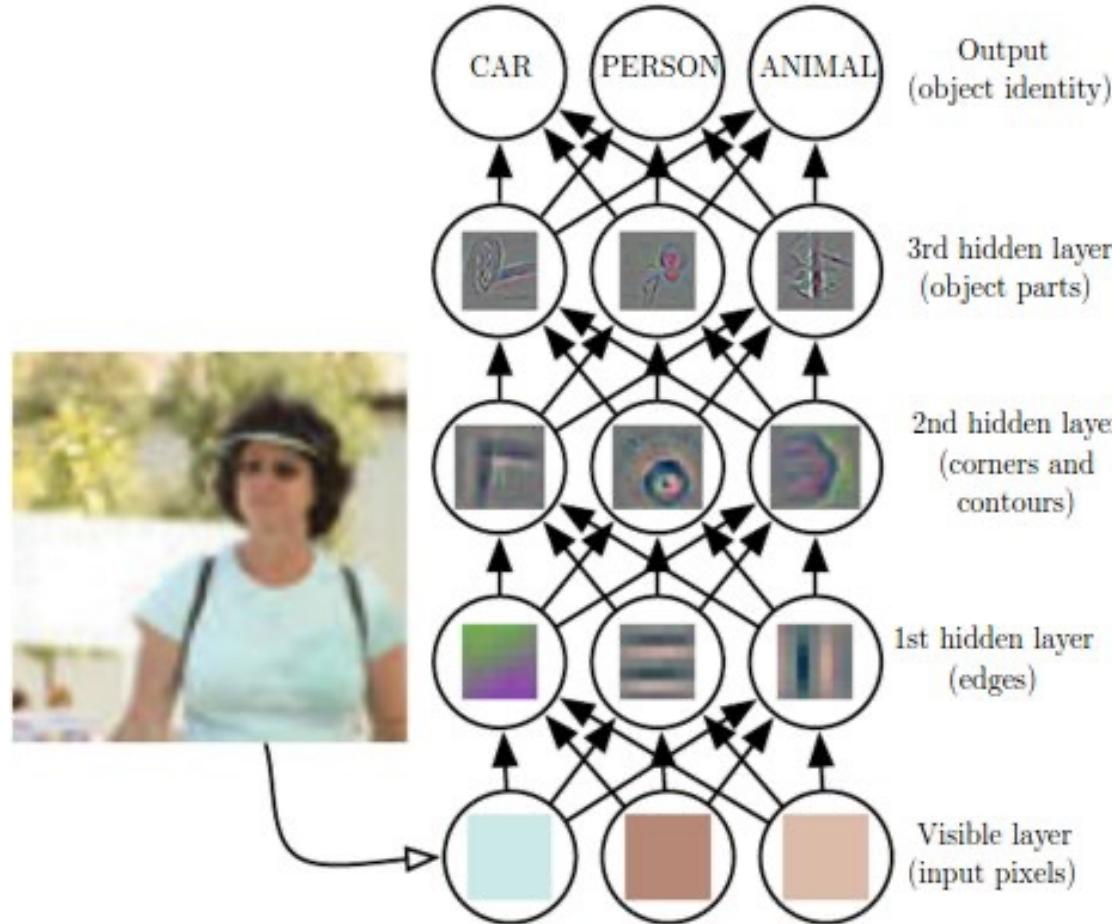
Exemplo: Previsão de preço de casa



Exemplo Complexo: Visão Computacional



Feature Extractions on a Deep Neural Network



Multi-class Classification



Cat



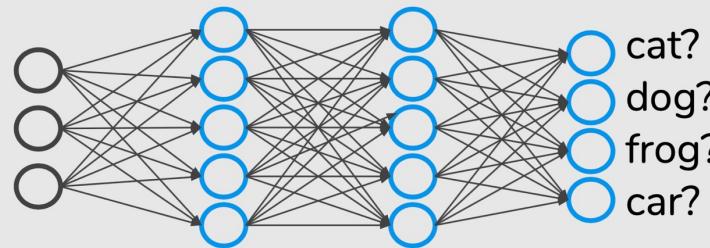
Dog



Frog



Car

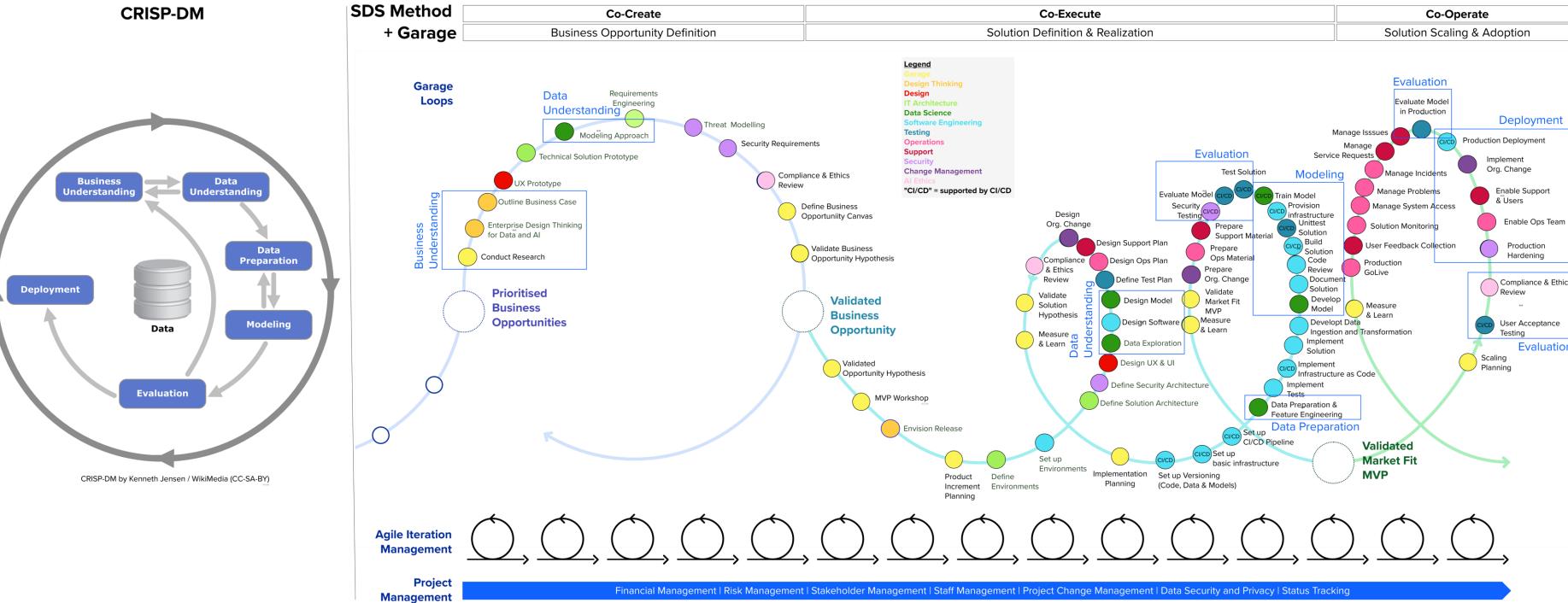


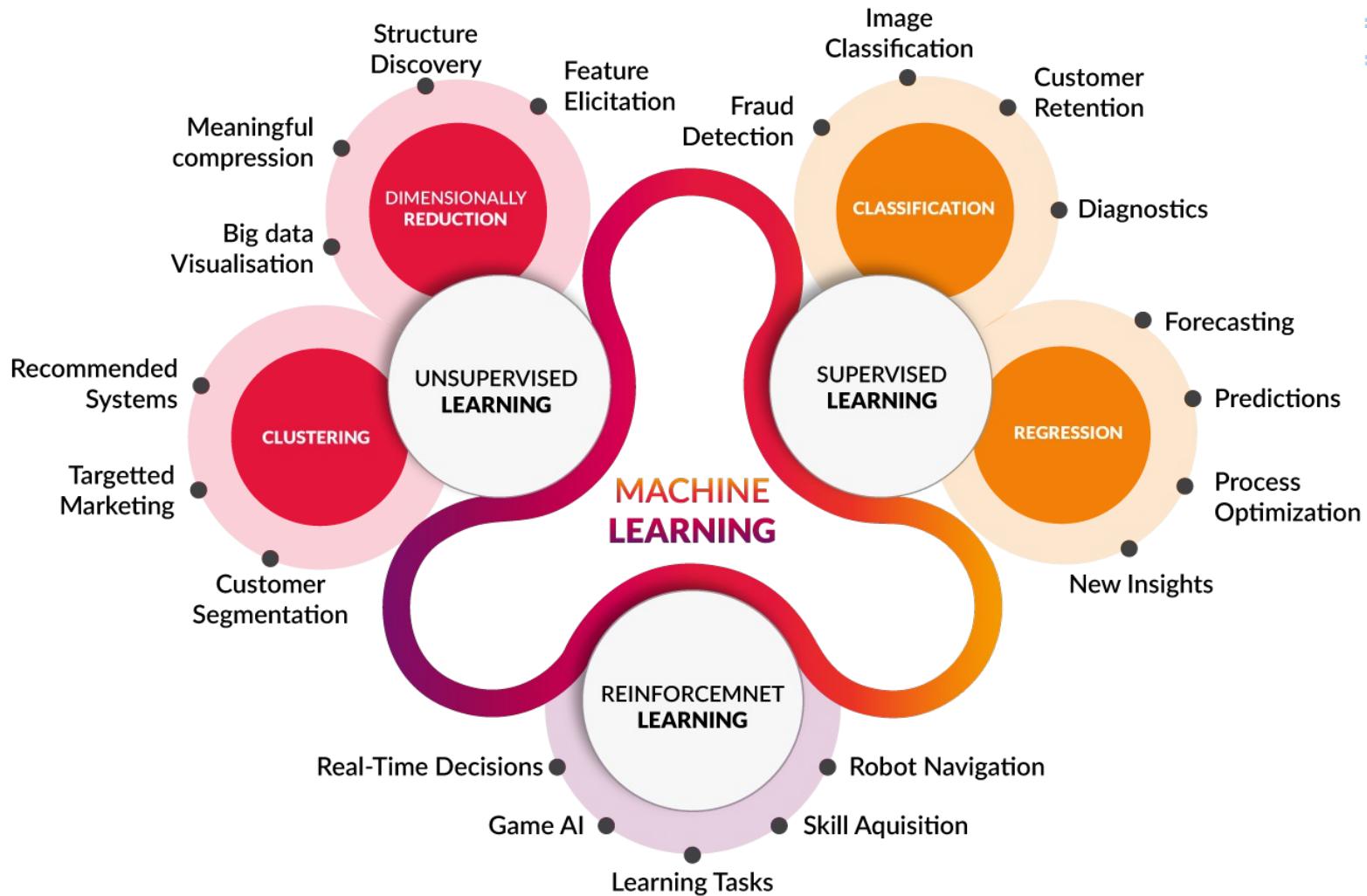
$$\text{Want } h_{\Theta}(x) \approx \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad h_{\Theta}(x) \approx \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \quad h_{\Theta}(x) \approx \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \quad h_{\Theta}(x) \approx \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

when cat when dog when frog when car

Crispier-DM

Scaled Data Science Method expands CRISP-DM with Software Engineering, Design Thinking, and many more



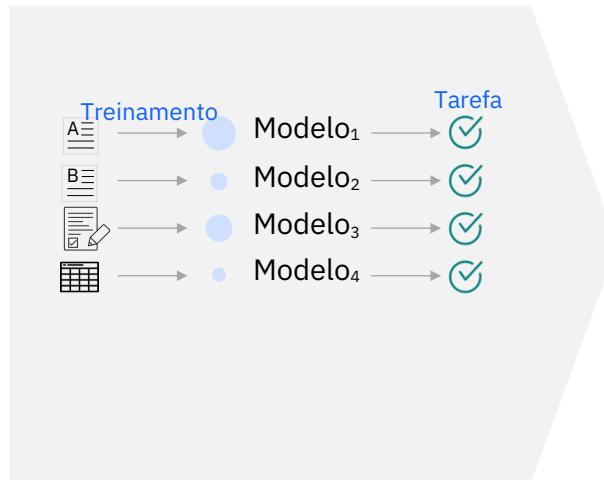


E com IA Generativa...

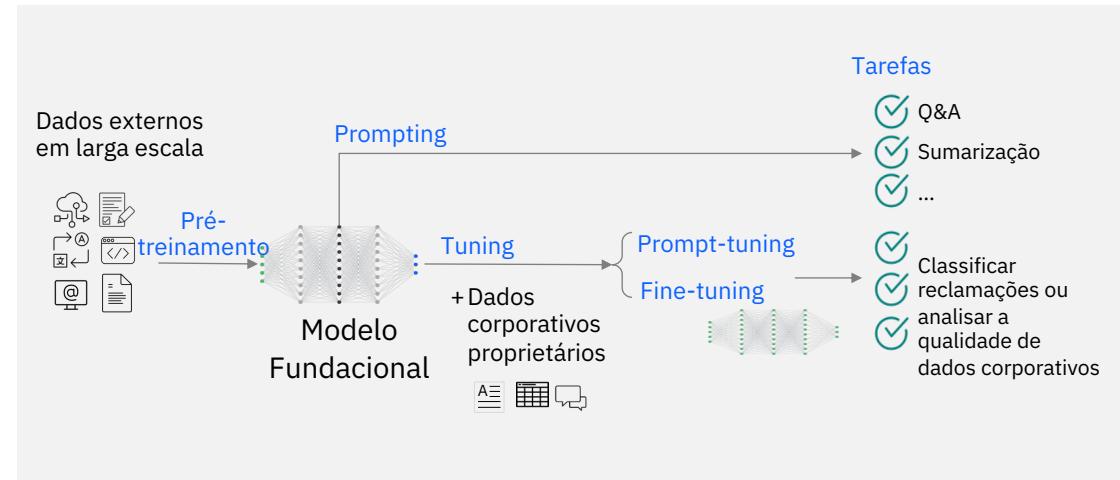
Como funciona?!

Os Modelos Fundacionais estão permitindo um novo paradigma no desenvolvimento eficiente de Inteligência Artificial em escala

Modelos de ML / DL convencionais



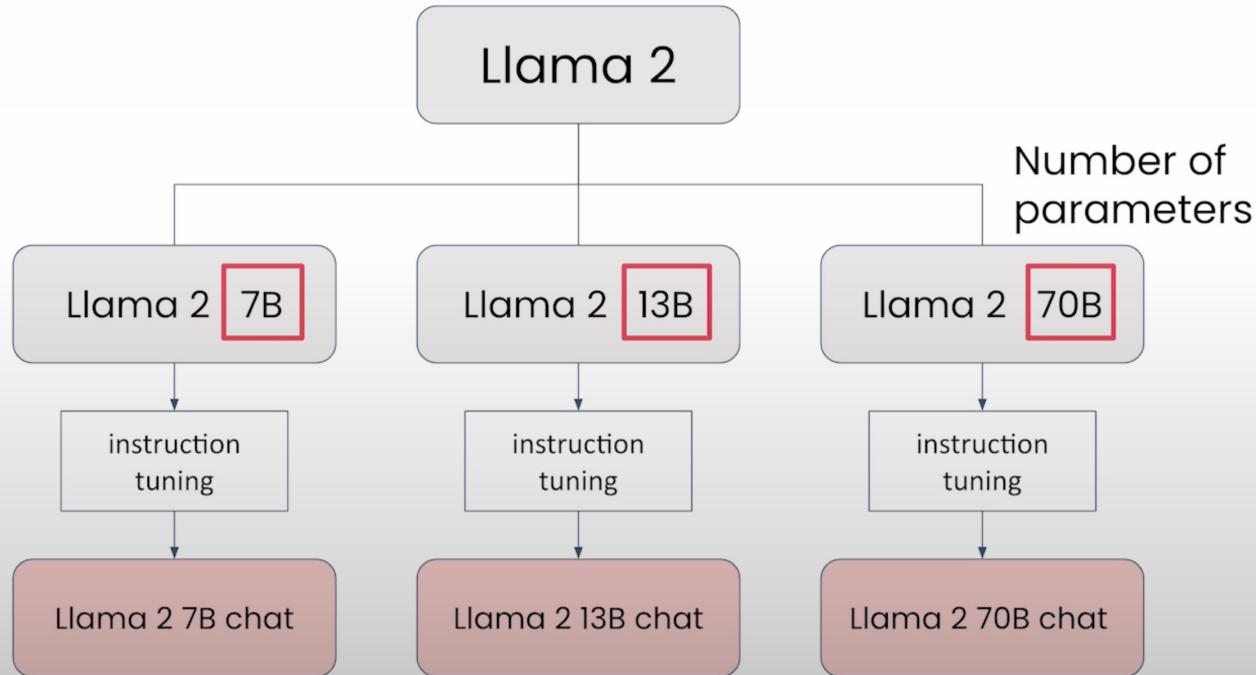
Modelos Fundacionais



- Modelos de silos individuais
- Requer treinamento específico para cada tarefa
- Requer treinamento supervisionado

- Adaptação rápida a várias tarefas com pequenas quantidades de dados específicos da tarefa
- Aprendizagem não supervisionada pré-treinada

Prompts e Geração de Texto



Transformers

Arquitetura simplificada

Encoder

Codifica a entrada (“prompts”) com um entendimento Contextual e produz um vetor para cada token de entrada.

Embeddings

Entradas

Encoder

Saída

Função Softmax

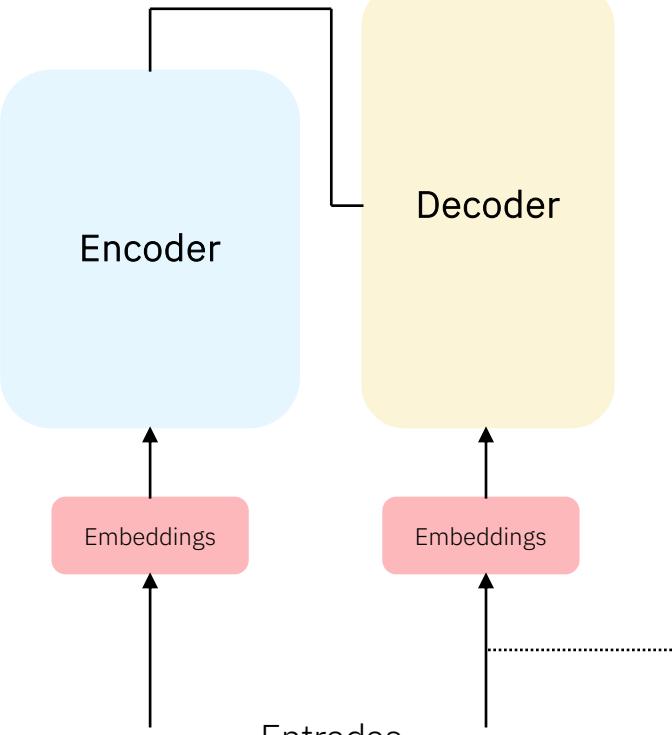
Decoder

Decoder

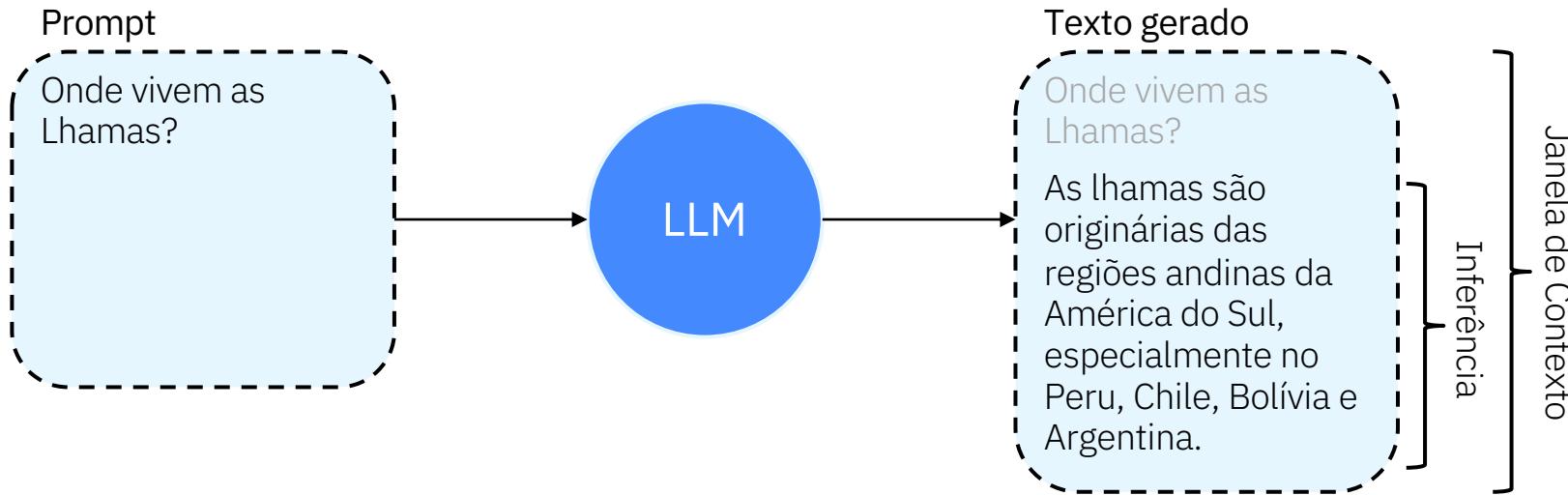
Aceita tokens de entrada e gera novos tokens.

Embeddings

Entradas

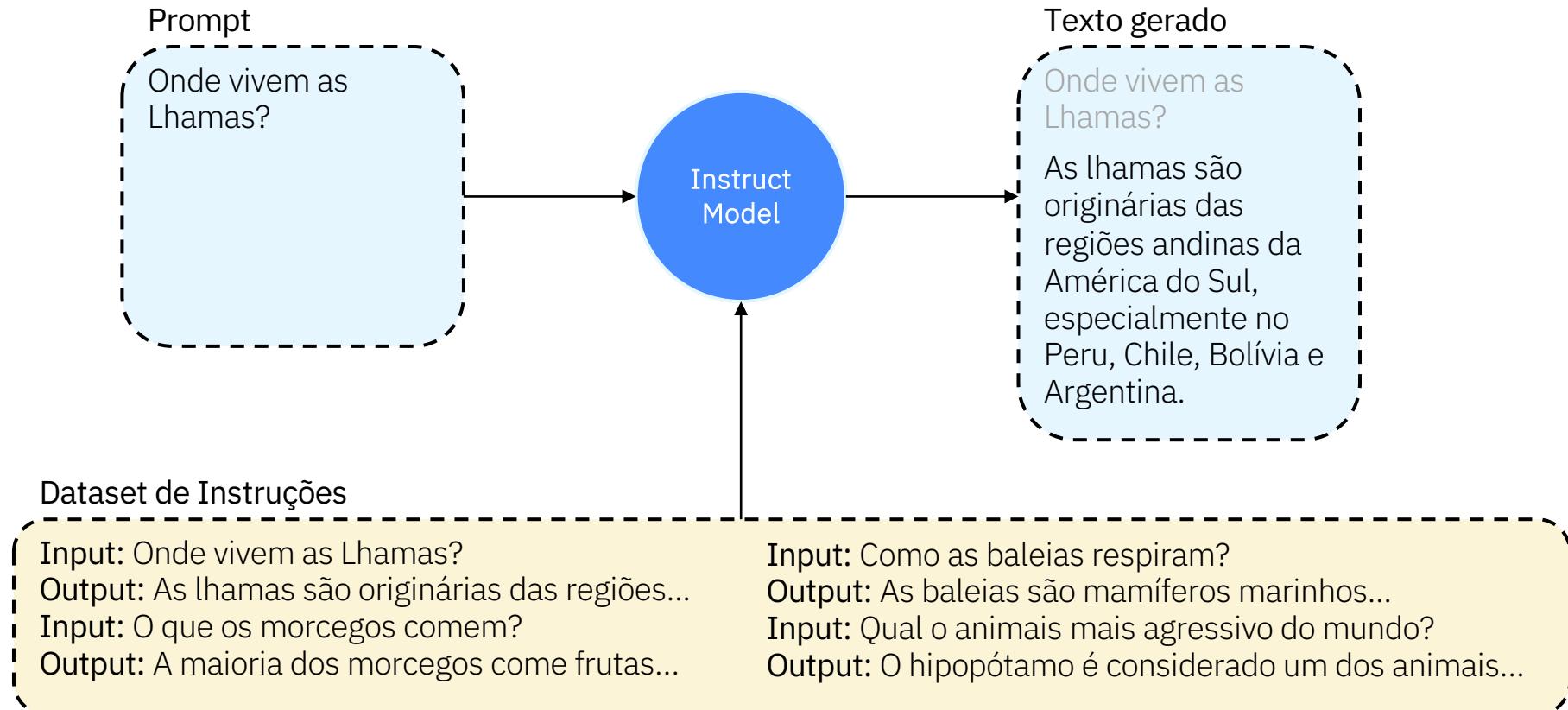


Prompts e Geração de Texto

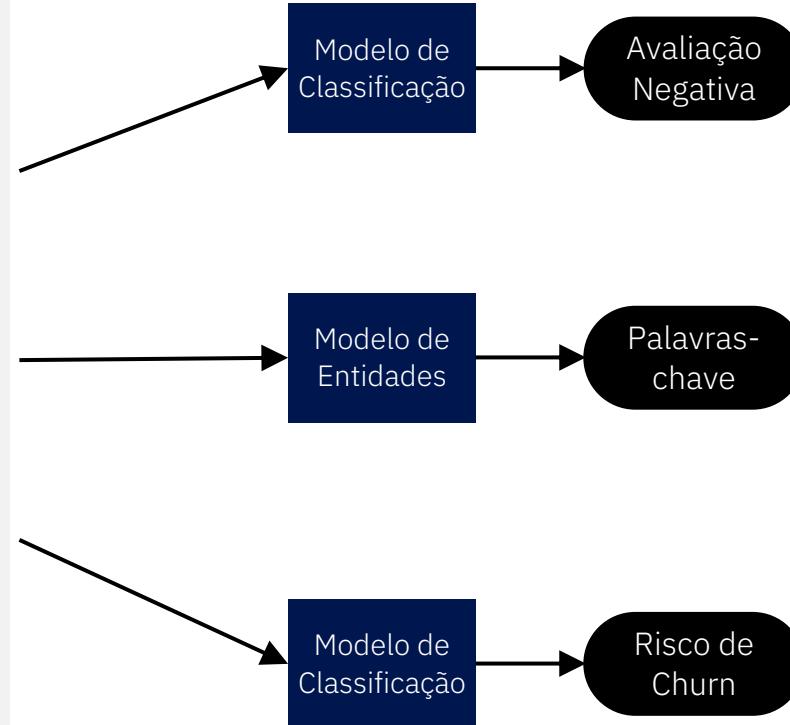


Diferentes modelos possuem diferentes tamanhos de *Context Window*:
2k ~ 16k tokens

Modelos *Instruct-tuned*



“ Apesar das praticidades do meu banco digital, tive uma péssima experiência via chat. Enfrentei demoras absurdas nas respostas, falta de empatia e cortesia, além de problemas técnicos.
(...) levando-me a considerar encerrar minha relação com o banco.”



“ - - - - -
Apesar das praticidades do meu banco digital, tive uma péssima experiência via chat. Enfrentei demoras absurdas nas respostas, falta de empatia e cortesia, além de problemas técnicos.
(...) levando-me a considerar encerrar minha relação com o banco.”



“
“id_cliente”: 2042357;
“classificacao”: “negativa”;
“pontos_positivos”: “praticidades”;
“pontos_negativos”: “péssima experiência via chat”;
“pontos_melhoria”: [“demora na resposta”, “falta de empatia e cortesia”, “problemas técnicos”];
“risco_churn”: True;
“resumo”: “Péssima experiência de atendimento via chat em um banco digital devido a demoras, atendentes despreparados, falta de empatia e problemas técnicos, levando à insatisfação do cliente e possível encerramento da relação com o banco.”

Tokens

Por palavra

LETRA	Indice	LETRA	Indice	LETRA	Indice
A	1	ELE	104165	PORQUE	252197
ABAIXO	2
ABALADO	3	ESTAVA	119840	PREGUIÇA	254276
...
ARQUITETO	26887	NÃO	213002	REINICIOU	267358
...
COM	76912	O	219195	SERVIDOR	281593
...	ZUMBIR	320094

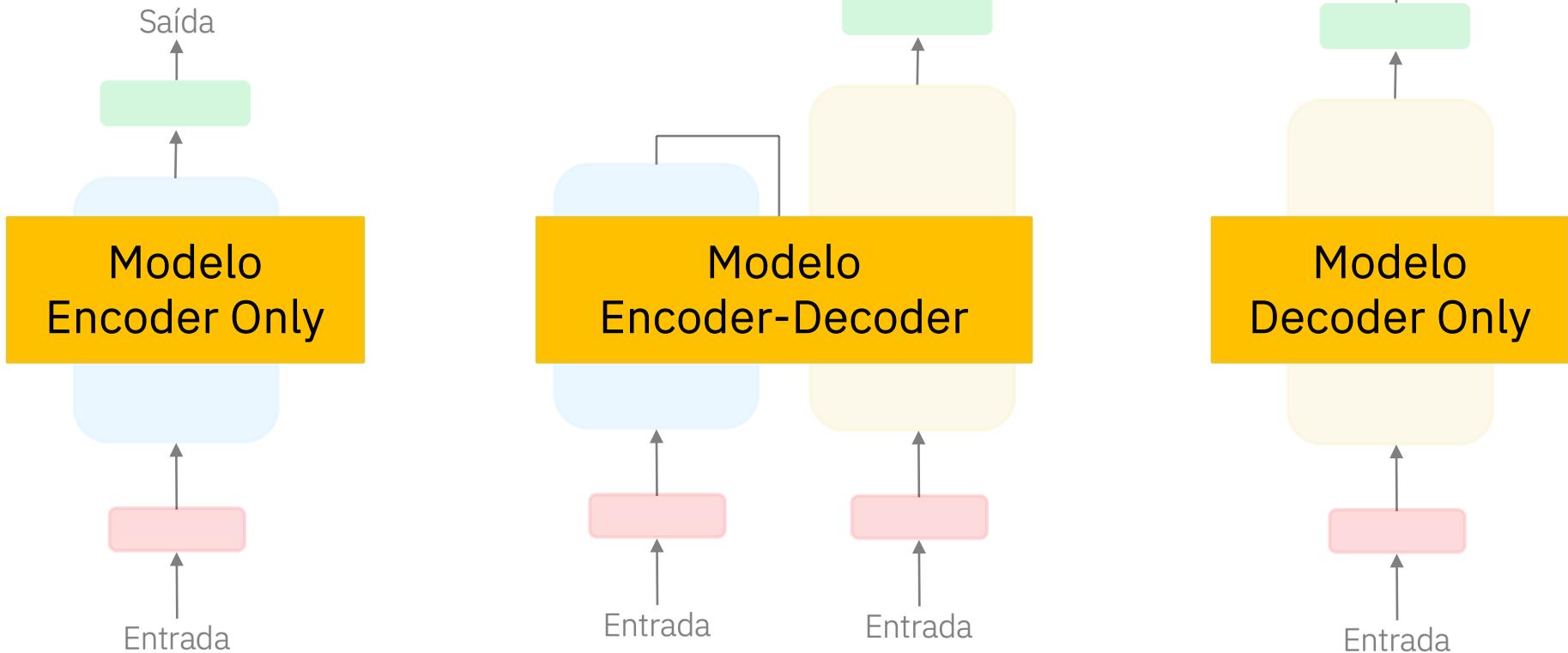
219195 26887 213002 267358 219195 281596
 O arquiteto não reiniciou o servidor
 252197 104165 119840 76912 254276
 porque ele estava com preguiça.

[219195, 26887, 213002, 267358, 219195, 281596, 252197, 104265, 119840, 76912, 254276]

Quantidade de Tokens: 11

Transformers

Diferentes Arquiteturas



Transformers

Arquitetura simplificada

Encoder

Codifica a entrada (“prompts”) com um entendimento Contextual e produz um vetor para cada token de entrada.

Embeddings

Encoder

Entradas

Saída

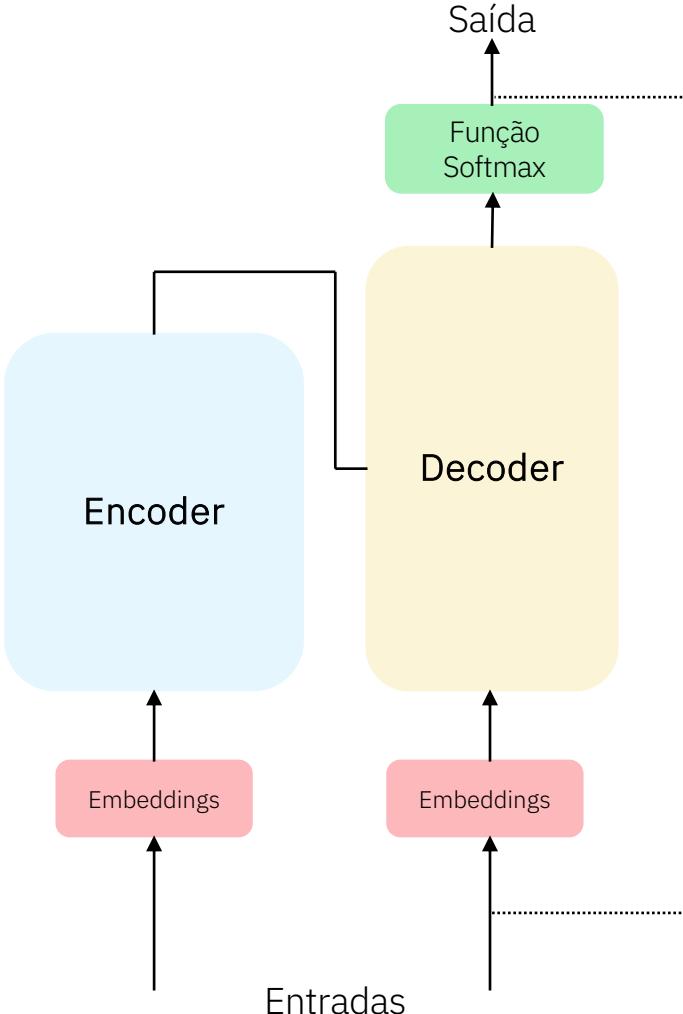
Função Softmax

Decoder

Embeddings

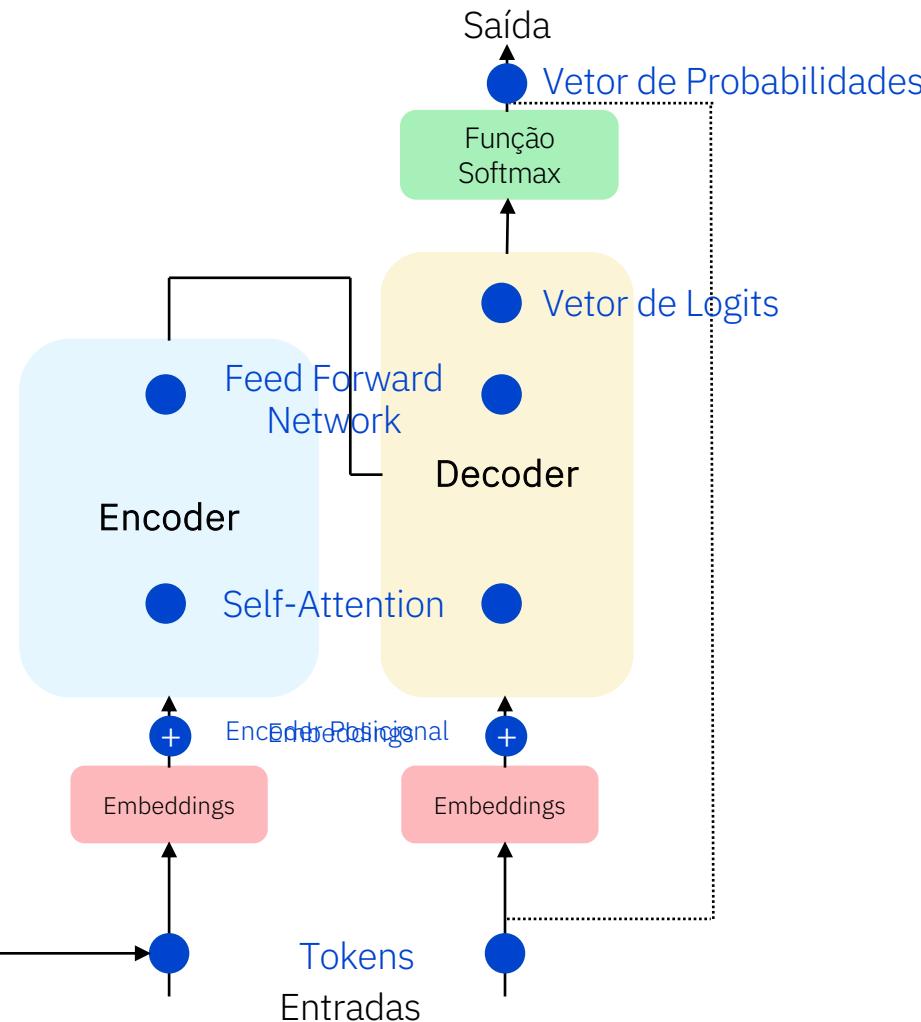
Decoder

Aceita tokens de entrada e gera novos tokens.



Transformers

Arquitetura simplificada



Parâmetros de Inferência

Definindo como os modelos irão responder

Min tokens: quantidade mínima de tokens a serem gerados.

Max tokens: quantidade máxima de tokens a serem gerados.

Stop sequences: caractere, ou sequência de caracteres, para delimitar o final da inferência.

Decoding: técnica de decodificação.

Decoding

Greedy Sampling [\(i\)](#)

Temperature

0 2 0,7

Top P (nucleus sampling)

0 1 1

Top K

1 100 50

Random seed

Repetition penalty

1 2 1

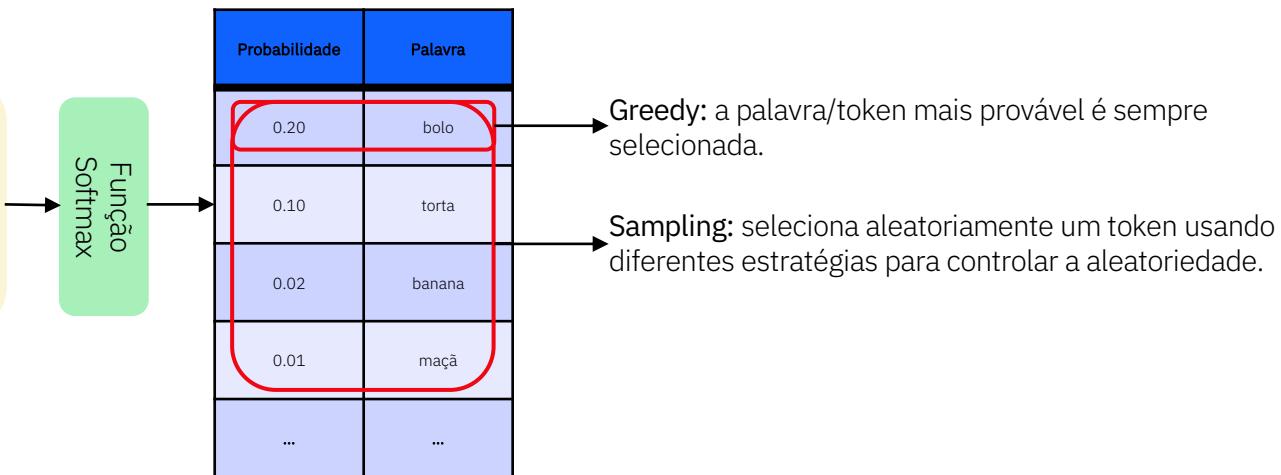
Stopping criteria [\(i\)](#)

Stop sequences

[+](#)

Min tokens 0

Max tokens 20



Parâmetros de Inferência

Definindo como os modelos irão responder

Top K: limita a aleatoriedade da inferência.

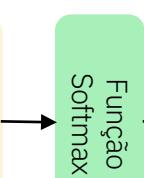
Permite que o modelo escolha entre os Top-K tokens para inferência.

Top P: limita a aleatoriedade da inferência.

Permite que o modelo escolha um token de uma distribuição que não ultrapasse a probabilidade P.

Top R = 0.31

Probabilidade	Palavra
0.20	bolo
0.10	torta
0.02	banana
0.01	maçã
...	...



Decoding

Greedy Sampling [\(i\)](#)

Temperature

0 2 0,7

Top P (nucleus sampling)

0 1 1

Top K

1 100 50

Random seed

Repetition penalty

1 2 1

Stopping criteria [\(i\)](#)

Stop sequences

Min tokens Max tokens

0 20

Parâmetros de Inferência

Definindo como os modelos irão responder

Temperatura: influencia a distribuição de probabilidades que o modelo calcula para o próximo token.

Quanto maior a temperatura, maior a aleatoriedade – ou a “criatividade” do modelo.

Random seed: semente de aleatoriedade, para garantir consistência nos resultados.

Repetition penalty: penalidade aplicada sobre as probabilidades de palavras repetidas.

Decoding

Greedy Sampling [\(i\)](#)



Top P (nucleus sampling)



Top K



Random seed

Repetition penalty



Stopping criteria [\(i\)](#)

Stop sequences

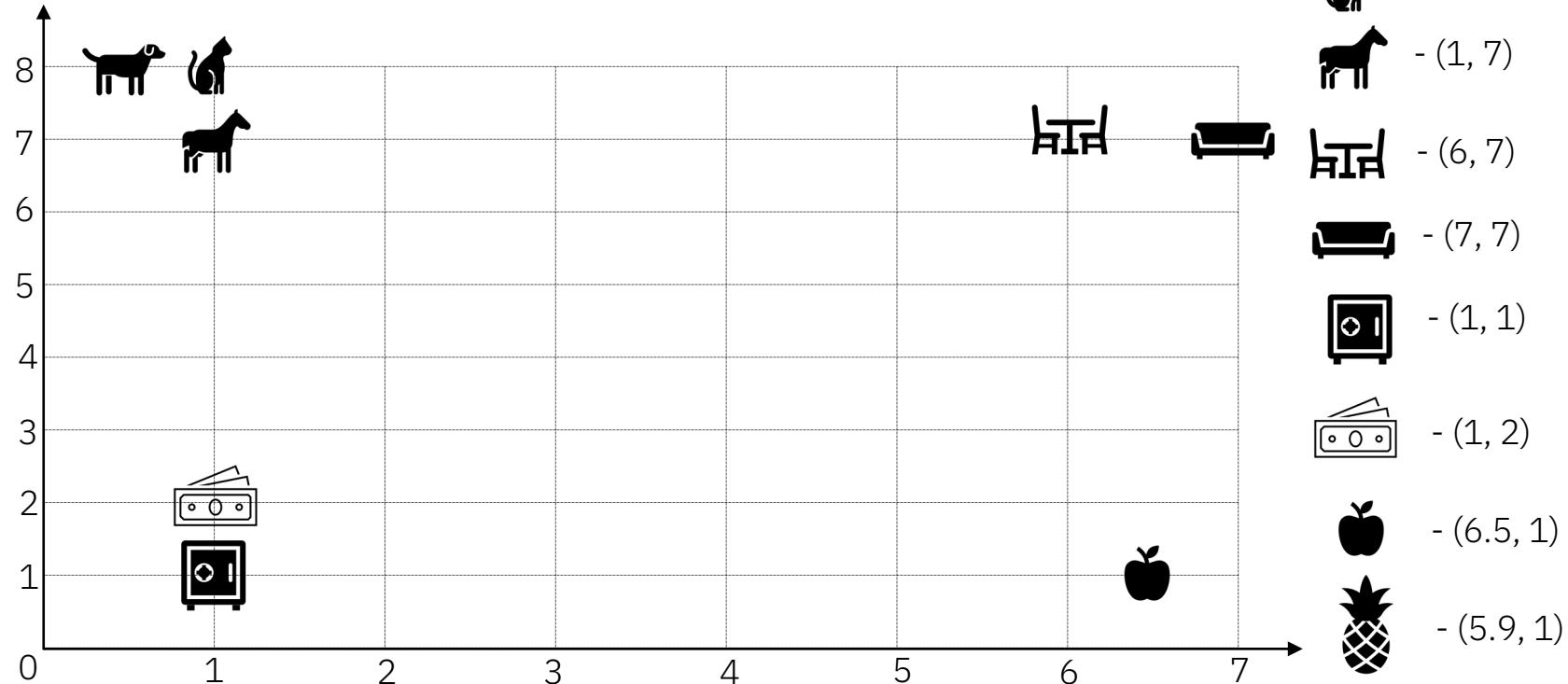
 +

Min tokens

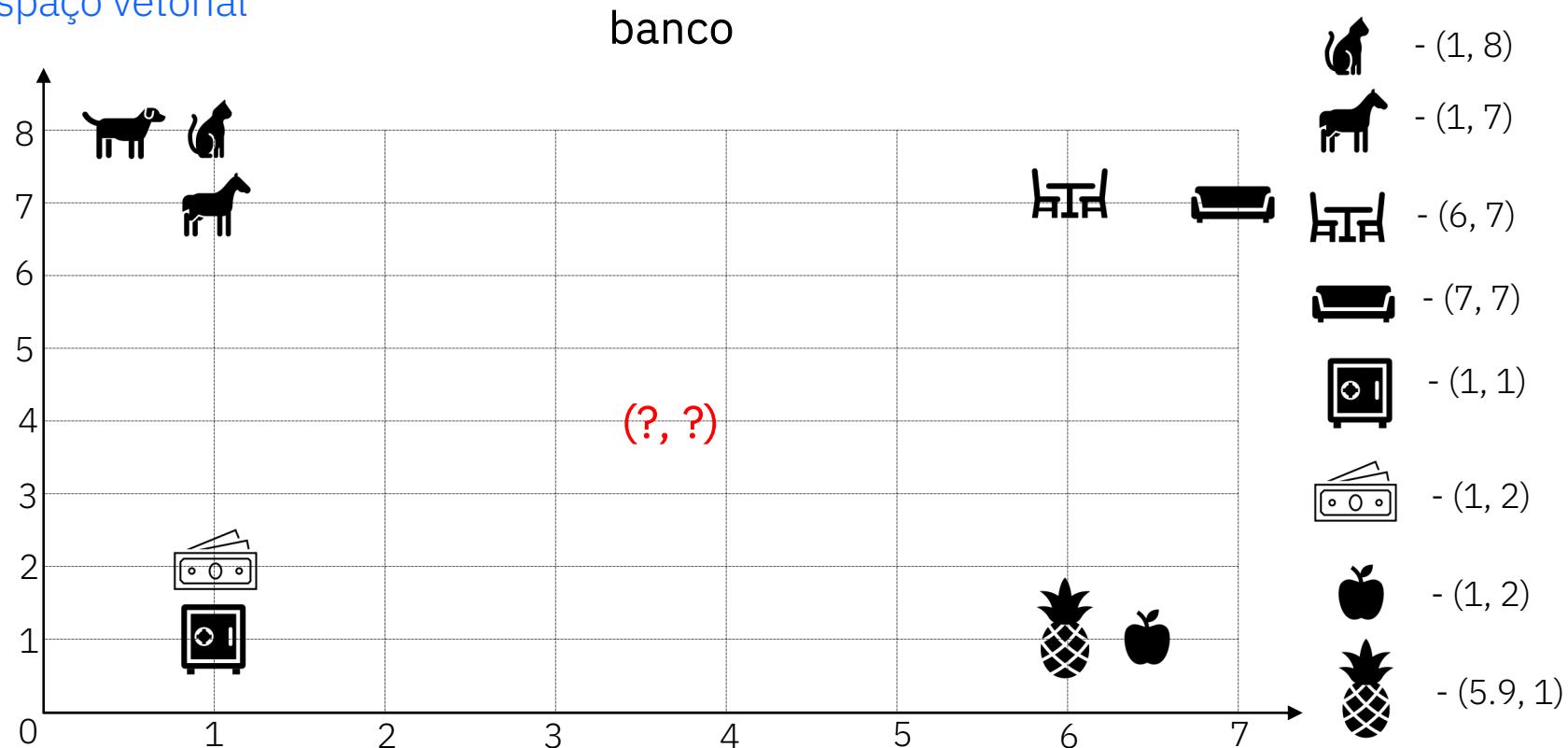


Max tokens

Embeddings e o espaço vetorial



Embeddings e o espaço vetorial



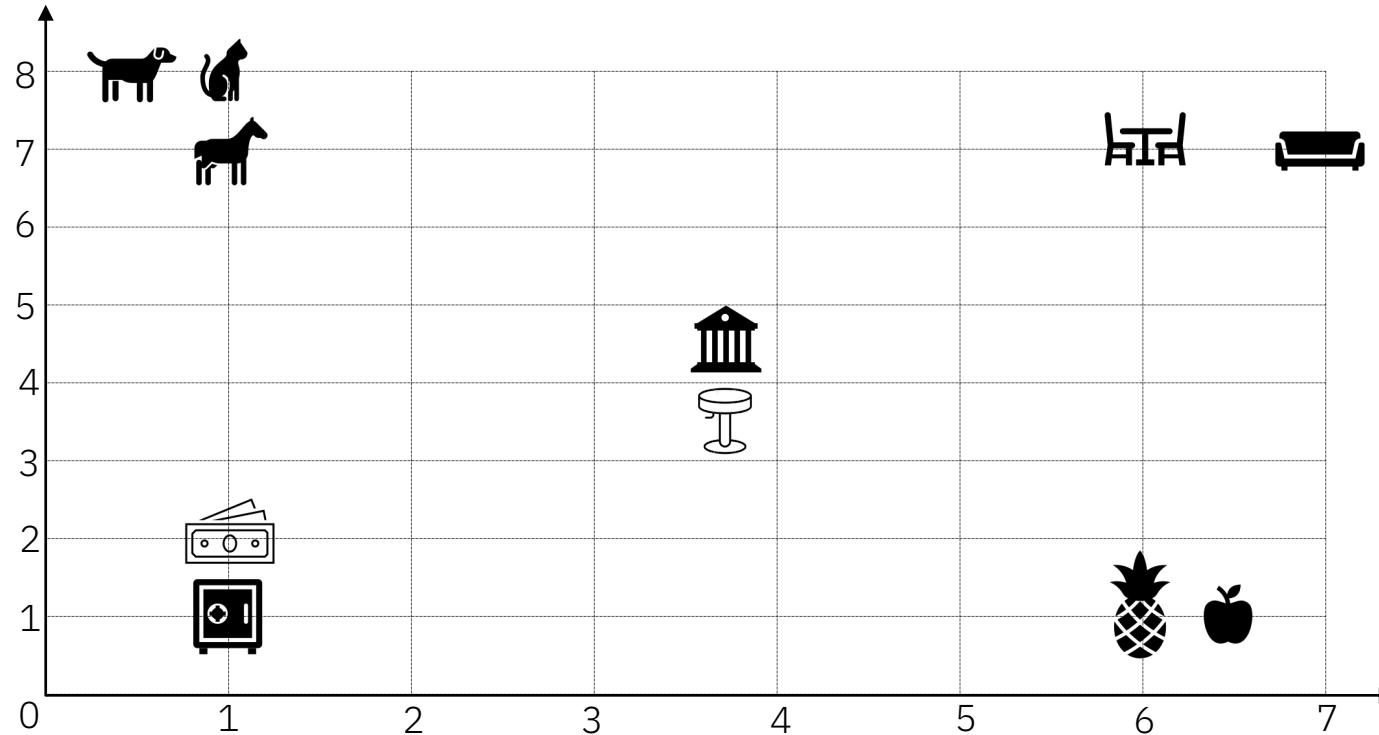
Embeddings

e o espaço vetorial

“Estava no banco



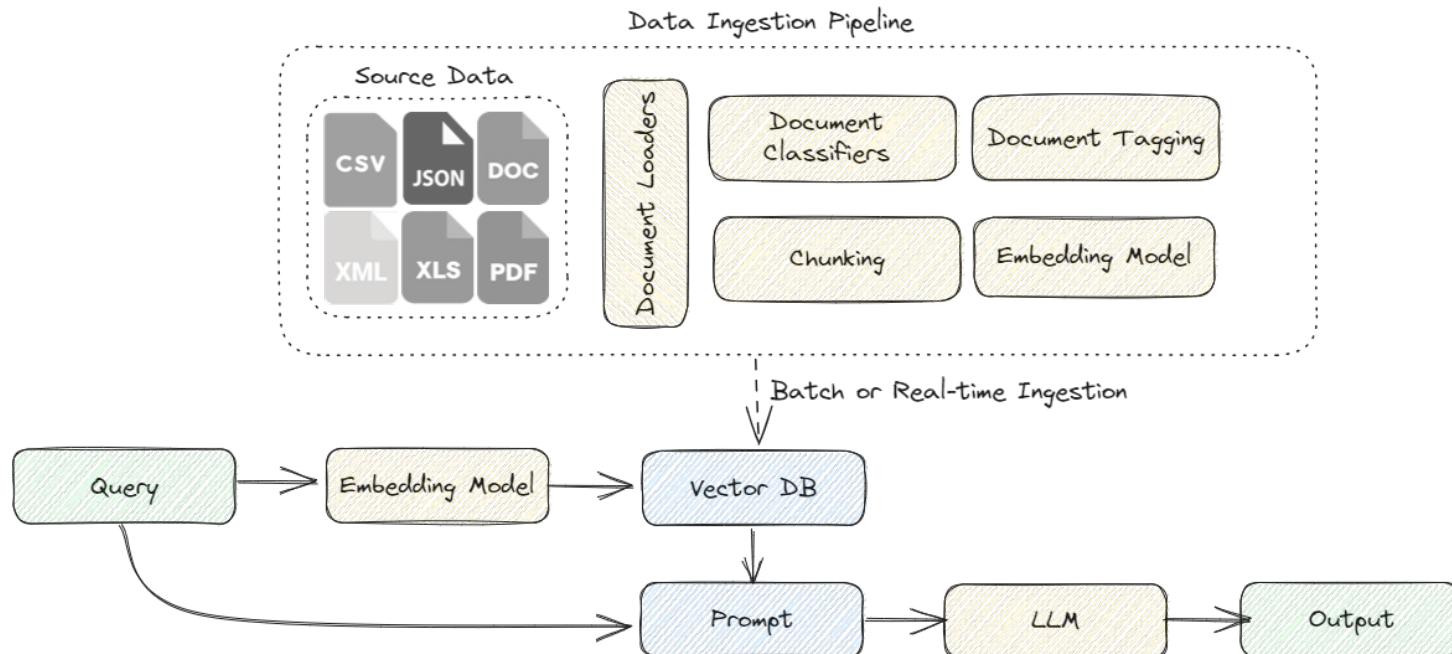
“Escolha um banco



O que é RAG?

Retrieval-Augmented Generation

RAG combina processos de recuperação e geração para aprimorar as capacidades dos LLMs



O que é RAG?

Exemplo muito simples...

1. Dada uma pergunta, pesquise documentos relevantes para obter resposta

Há estacionamento para funcionários?

Documentos da empresa



Benefícios



Política de
licença



Instalações



Folha de
pagamento

2. Incorpore o texto recuperado em um prompt atualizado

Use as seguintes partes do contexto para responder à pergunta no final:

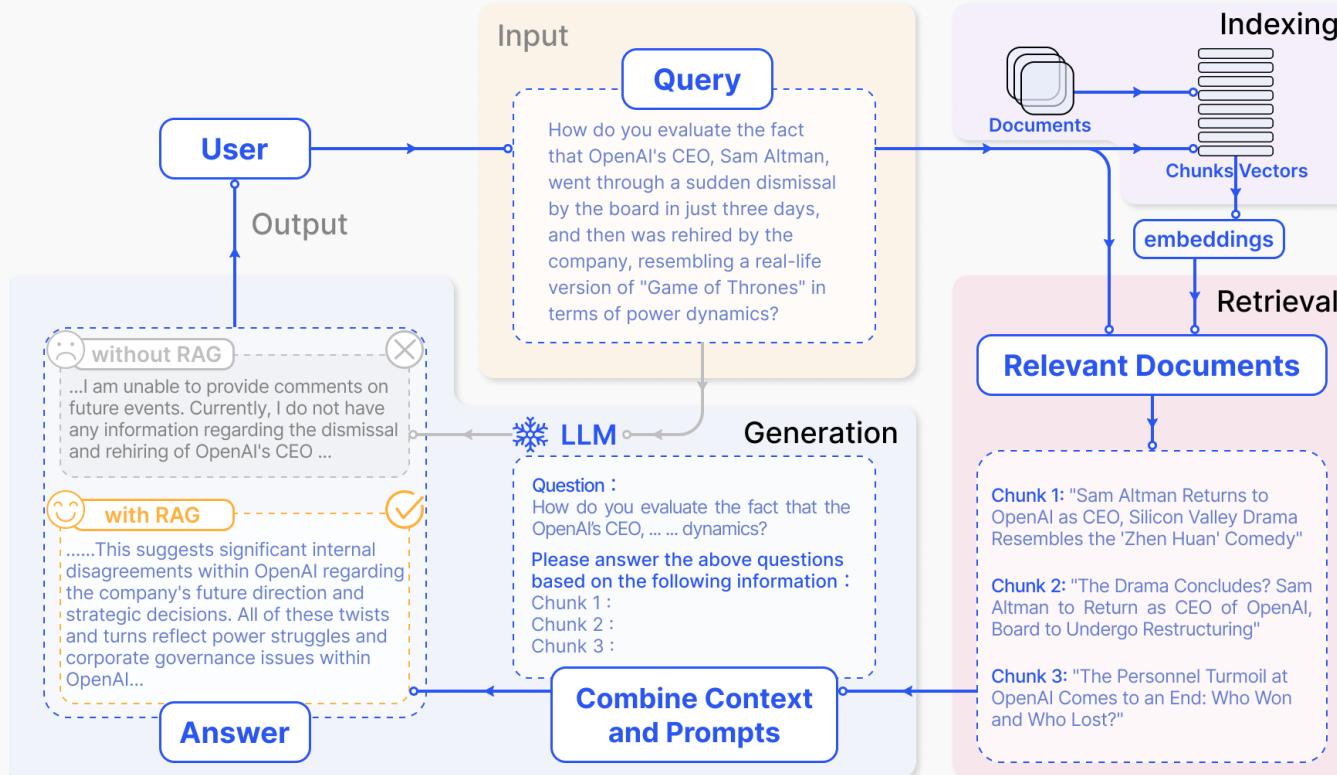
Política de Estacionamento: Todos os funcionários podem estacionar nos níveis 1 e 2 do escritório.

Use a entrada na Front St [...]

Há estacionamento para funcionários?

O que é RAG?

Exemplo de ponta a ponta...

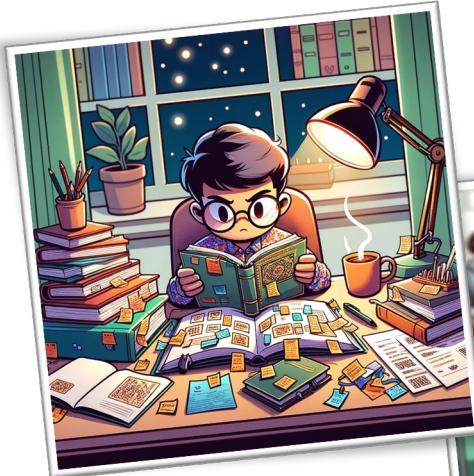


RAG

VS

Fine-Tuning

Treinamento



Inferência



Treinamento



IBM

Vamos colocar a Mão na Massa?



PERGUNTAS



Obrigado!

jorgedbchagas



jorgedbchagas@gmail.com



